AI- Enabled Real-Time Speech -To- Sign Language Conversion with An Interactive Chatbot and a Learning Module

A. Sathvika[1], J.Hemalatha[2] , Mrs. Chinchu Nair[3], Dr. R. Shoba Rani[4], Dr. F.Jerald[5]

[1,2,3,4,5]Department of Artificial Intelligence, Dr. M.G.R. Educational and Research Institute, Chennai 600095, India

ABSTRACT:
This project presents an innovative Sign Language Chatbot system designed to bridge communication gaps for the deaf and hard-of-hearing community. The system takes real-time sign language inputs, recognizes them using a Ridge Classifier-based model, and generates meaningful responses through an NLP and CNN-powered chatbot engine. Additionally, the chatbot produces a dynamic sign language avatar, visually delivering responses in sign form. To further support learning, the platform includes a training module where users can input text or voice commands and receives corresponding sign language outputs, enabling interactive and accessible sign language education. By integrating machine learning, computer vision, and natural language processing, this project creates a two-way communication and learning environment, promoting inclusivity and sign language literacy.

Keywords: Sign Language Recognition, Machine Learning, Convolutional Neural Networks (Cnn), Hand Gesture Recognition, Opencv, Ridge Classifier.

INTRODUCTION

Communication is a fundamental aspect of human interaction that allows individuals to exchange information, express emotions, and build social relationships. However, for individuals who are deaf or hard of hearing, communication with people who do not understand sign language remains a significant challenge. Sign language is a visual form of communication that uses hand gestures, body movements, facial expressions, and hand shapes to convey meaning. It is a complete language with its own grammar and syntax and is widely used by the hard-of-hearing community. Various sign languages exist across the world, including American Sign Language (ASL), Chinese Sign Language (CSL), and Arabic Sign Language (ArSL). Despite its importance, sign language is not widely understood by the general population, creating communication barriers between hearing and non-hearing individuals.

Traditional communication methods such as written text or human interpreters are often inefficient, limited, or unavailable in real-time situations. This limitation reduces accessibility and social inclusion for the deaf and hard-of-hearing community. Therefore, there is a strong need for intelligent technological solutions that can bridge the communication gap between sign language users and non-signers.

Recent advancements in machine learning, computer vision, and Natural Language Processing (NLP) have opened new opportunities for developing automated sign language recognition systems. These systems capture sign gestures using cameras and analyze them through image and video processing techniques. Static gestures, such as alphabet signs, can be recognized using images, while dynamic gestures that involve motion require video-based analysis. Key features such as hand shapes, finger positions, and hand movements play a crucial role in identifying sign language gestures.

Deep learning models, particularly Convolutional Neural Networks (CNN), have shown promising results in gesture recognition tasks. CNN models extract important visual features such as edges, shapes, and finger positions from gesture images through convolution and pooling layers and classify them into corresponding alphabets or words. Once gestures are recognized, Natural Language Processing techniques are used to interpret the generated text and convert it into meaningful language structures. In addition to gesture recognition, conversational agents such as chatbots have become widely used for automating communication in fields such as education, healthcare, and customer support. Chatbots use NLP techniques to understand user input and generate appropriate responses in a conversational manner.

Integrating chatbot technology with sign language recognition systems can significantly enhance interaction between sign language users and the general public.

Motivated by these advancements, this paper proposes a **Sign Language Chatbot System** that enables real-time bidirectional communication between deaf or hard-of-hearing individuals and people who are not familiar with sign language. The proposed system captures hand gestures using a camera and processes them through image preprocessing techniques. A **Ridge Classifier-based model** is used to recognize sign language gestures and convert them into textual representations. The recognized text is then processed by an **NLP-based chatbot**, which generates appropriate responses based on user input. To ensure accessibility for sign language users, the chatbot responses are converted into visual gestures using a **dynamic sign language avatar**.

Furthermore, the system includes a **training module** that allows users to learn and practice sign language using text or voice inputs. The proposed system aims to provide an accessible and interactive platform that improves communication between hearing and non-hearing individuals. By integrating gesture recognition, natural language processing, and chatbot interaction, the proposed system helps reduce communication barriers and promotes inclusivity for the deaf and hard-of-hearing community.

RELATED WORK

Sign language recognition and translation systems have gained significant research attention in recent years due to the increasing need to support communication for the deaf and hard-of-hearing community. Several studies have explored deep learning, computer vision, and natural language processing techniques to improve the accuracy and efficiency of sign language recognition systems.

Natarajan et al. [1] proposed an end-to-end deep learning framework for sign language recognition, translation, and video generation. Their approach integrates the MediaPipe library with a hybrid Convolutional Neural Network (CNN) and Bidirectional Long Short-Term Memory (Bi-LSTM) model to extract pose information and generate text from sign gestures. In addition, the system uses Neural Machine Translation (NMT) combined with Dynamic Generative Adversarial Networks (GAN) to produce sign gesture videos from spoken sentences. Experimental results demonstrated that the proposed framework achieved more than 95% classification accuracy and improved visual quality across multilingual sign language datasets.

Luqman [2] introduced an efficient two-stream deep learning architecture for isolated sign language recognition using accumulative video motion. The proposed model consists of three hierarchical networks: Dynamic Motion Network (DMN), Accumulative Motion Network (AMN), and Sign Recognition Network (SRN). The approach extracts key postures from gesture sequences and combines spatial and temporal information to enhance recognition performance. Experimental evaluation on Arabic sign language datasets showed that the model outperformed several existing techniques, particularly in signer-independent recognition scenarios.

Shanableh [3] proposed a two-stage deep learning framework for continuous Arabic sign language recognition. In the first stage, the system predicts the number of words present in a sign language sentence. In the second stage, the sentence is segmented and each segment is represented using motion images generated through frame difference estimation. These motion images are then processed using CNN-based feature extraction and Bidirectional LSTM models for sentence recognition. Experimental results demonstrated high recognition performance with word-level accuracy of 97.3% and sentence-level accuracy of 92.6%.

Kumar et al. [4] developed a sign language translation system that bridges the communication gap between American Sign Language (ASL) and Indian Sign Language (ISL). The proposed approach integrates a hybrid CNN and Random Forest Classifier (RFC) model for gesture recognition. The recognized gestures are converted into text and refined using Large Language Models (LLMs) to improve grammatical correctness and contextual understanding. Finally, the corrected text is converted into ISL gestures using the RIFE-Net model to generate smooth gesture videos. The system achieved 93% gesture recognition accuracy and demonstrated real-time translation capability.

Kuhail et al. [5] investigated the influence of chatbot-human personality congruence on user engagement in chatbot-based advising systems. The study analyzed how personality traits such as extraversion, agreeableness, and conscientiousness affect user trust and interaction with conversational agents. The experimental results showed that personality alignment between users and chatbots significantly improves engagement and user experience, highlighting the importance of conversational design in intelligent communication systems.

Özcan et al. [6] introduced a zero-shot sign language recognition approach that utilizes hand and body pose features for recognizing previously unseen sign gestures. The proposed method extracts landmark features from hand and body movements and employs a self-attention mechanism to generate visual embeddings. These embeddings are combined with textual descriptions of signs to support zero-shot learning. Experimental evaluation on benchmark datasets demonstrated the effectiveness of the approach in recognizing gestures without requiring extensive labeled training data.

Ranjbar and Taheri [7] presented ISLR101, a large-scale dataset designed for Iranian word-level sign language recognition. The dataset includes 4,614 video samples representing 101 unique signs collected from multiple participants. The authors also introduced baseline recognition frameworks using both visual appearance-based models and skeleton-based pose estimation techniques. Experimental results showed that the proposed baseline models achieved accuracies of 97.01% and 94.02%, respectively, demonstrating the effectiveness of the dataset for advancing sign language recognition research.

El Ghoul et al. [8] proposed JUMLA-QSL-22, a large-scale annotated dataset for continuous Qatari sign language recognition. The dataset consists of 6,300 recordings representing 900 sentences commonly used in healthcare environments.

Data were collected from diverse participants using advanced depth cameras to capture different signing styles and variations. The dataset aims to support research in continuous sign language processing and improve recognition systems for real-world applications.
Although previous studies have achieved considerable progress in sign language recognition, translation, and dataset development, most existing systems focus primarily on gesture recognition or translation tasks independently. Furthermore, limited research has integrated sign language recognition with intelligent conversational systems capable of supporting real-time bidirectional communication. To address these limitations, the proposed work aims to develop a **Sign Language Chatbot System** that integrates gesture recognition, natural language processing, and chatbot interaction to enable seamless communication between sign language users and non-signers.

PROPOSED SYSTEM

The proposed system combines advanced techniques from computer vision, machine learning, and natural language processing to create a fully interactive, real-time sign language chatbot. By using a Ridge Classifier-based model for sign recognition, the system is capable of interpreting various sign language gestures with high accuracy. It generates contextual text responses powered by an NLP engine, which are then visually conveyed through a dynamic sign language avatar. The platform includes an integrated training module that allows users to practice sign language by inputting text or voice commands and receiving corresponding sign outputs. This dual approach, where users interact both through speech/text and sign language, facilitates communication while enhancing sign language literacy. The system aims to bridge communication gaps in real time, enabling more inclusive interactions between individuals with hearing impairments and the broader population.

A. REAL-TIME SIGN LANGUAGE RECOGNITION

The heart of this project lies in its ability to recognize real-time sign language inputs through a Ridge Classifier-based model. This technology allows the system to accurately interpret gestures made by users, translating them into recognizable commands. By utilizing computer vision techniques, the system captures and analyzes hand shapes, positions, and movements to identify specific signs. The Ridge Classifier is chosen due to its efficiency in handling complex, high-dimensional data, which is crucial for real-time applications. This component forms the first step in the communication process, converting visual sign language inputs into digital formats that the system can understand. The system's performance is enhanced by continuous learning, improving its accuracy over time as it interacts with more users. Furthermore, the real-time nature of the recognition ensures that the conversation flow remains natural, eliminating significant delays. The system's ability to process input quickly is vital for fostering meaningful interactions between deaf individuals and others. This feature represents a major technological advancement in facilitating communication within the deaf and hard-of-hearing community.

FIGURE 1: SIGN LANGUAGE TRAINING

B. NATURAL LANGUAGE PROCESSING AND CHATBOT ENGINE

Once the sign language is recognized, the next crucial step involves processing it to generate meaningful responses. This is achieved through a sophisticated Natural Language Processing (NLP) and chatbot engine. The NLP algorithms enable the system to understand the context and intent behind the sign language input, allowing it to produce appropriate, coherent replies. This module ensures that the conversation remains fluid, with the chatbot not only responding based on predefined answers but also generating dynamic responses based on user inputs. The system is designed to handle a wide variety of queries, ranging from simple questions to more complex statements, making it adaptable for both casual and educational conversations.

By combining NLP with Convolutional Neural Networks (CNN), the system can refine its understanding of human language and sign language over time, providing a continuously improving user experience. This interaction mimics natural communication, providing a more intuitive and accessible method for those who rely on sign language. Ultimately, this integration of NLP with the chatbot engine creates an interactive, conversational interface that bridges language gaps for the deaf community.

FIGURE 2: NLP SENTENCE FORMATION

C. SIGN LANGUAGE AVATAR FOR VISUAL COMMUNICATION

A standout feature of this project is the use of a dynamic sign language avatar that visually delivers the chatbot's responses in sign language. This avatar serves as a crucial component, ensuring that individuals who use sign language can receive information in their preferred communication format.

Instead of relying on text or speech, the avatar translates the system's textual responses into sign language, enabling a more inclusive interaction. The avatar is designed to be expressive and accurate in its gestures, mimicking the fluidity and intricacy of human sign language. This visual aspect not only makes the system more accessible but also helps users better understand the context of the conversation. Additionally, the avatar provides a personalized experience by adjusting its hand shapes and movements based on the specific sign language being used, offering a customized learning and communication platform.

For individuals learning sign language, the avatar serves as both an educational and communicative tool, helping them to grasp proper hand movements and expressions. The incorporation of such a feature allows for an innovative, user-friendly approach to sign language communication.

FIGURE 3: AVATAR TRAINING

D. INTERACTIVE TRAINING MODULE FOR SIGN LANGUAGE EDUCATION:

To further enhance inclusivity, the project includes an interactive training module that enables users to learn sign language. This module allows users to input text or voice commands and receive corresponding sign language outputs, thus facilitating an engaging and hands-on learning experience. The training system uses machine learning to adapt to individual users, personalizing lessons to suit their pace and learning style. By offering real-time feedback and visual sign language demonstrations, it ensures that learners can practice and improve their skills in a supportive environment.

The module includes a variety of lessons, from basic sign language vocabulary to more advanced phrases, catering to users at different stages of learning. This feature not only promotes sign language literacy but also supports the creation of a more inclusive society where communication barriers are minimized. As the user progresses, the system can provide additional challenges and resources, making learning dynamic and interactive. Ultimately, this training module empowers individuals to both communicate in sign language and gain a deeper understanding of the culture and nuances behind it.

FIGURE 4: TRAINING MODULE

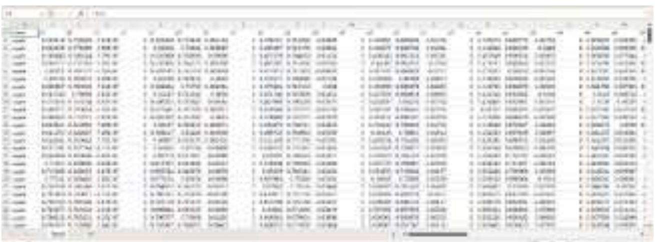

FIGURE 5: SIGN TRAINING

SYSTEM ARCHITECTURE

At its core, the system is organized using a layered and modular architecture in which each component performs a specific function while remaining interconnected with other modules. Tasks are distributed across dedicated layers responsible for voice processing, language understanding, gesture generation, and learning support. This structured design ensures low latency, scalability, and efficient real-time performance. When a user provides input through speech or text, the system processes it immediately[4]. Spoken input is captured via a microphone and converted into a digital format, while typed input is received directly through the user interface. In some cases, hand gestures captured through a camera interface may also serve as [4]. These multiple input modalities enable flexible and intuitive interaction while minimizing unnecessary steps in the communication process.

The processing layer integrates speech recognition, machine learning, and natural language understanding. [4]. Speech signals are converted into text using neural network–based automatic speech recognition models [1].

The transcribed text undergoes semantic and contextual analysis to determine intent and meaning. Based on this analysis, the chatbot module generates contextually appropriate responses. Simultaneously, text intended for sign conversion is mapped to learned gesture representations derived from motion-based pattern recognition models.

The output layer presents information through synchronized visual and textual formats. A digital avatar renders sign language gestures in real time, ensuring natural and fluid motion. Textual output is displayed alongside the animation to enhance clarity and accessibility.

This multimodal presentation improves comprehension for users with varying communication preferences. The system also incorporates a learning component that supports sign language practice. Users can observe animated gestures corresponding to spoken or written input and receive guided feedback to reinforce learning. Repeated exposure to synchronized text and motion enhances retention and fluency.

Overall, the coordinated interaction between input, processing, and output layers ensures efficient communication. The system enables seamless interaction between deaf signers and non-signers while promoting accessibility, inclusivity, and practical real-world deployment.

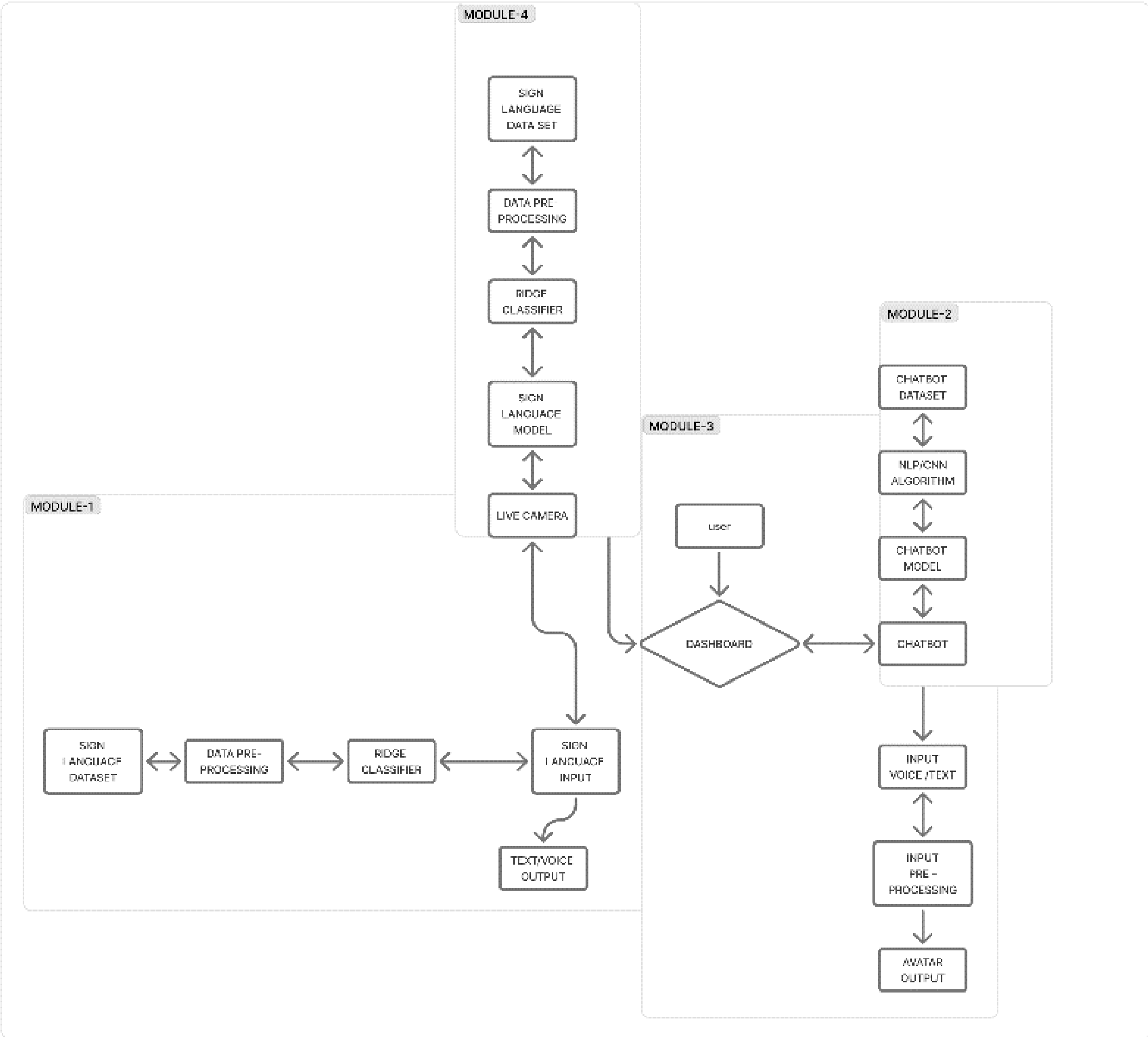

FIGURE 6 : SYSTEM ARCHITECTURE

ALGORITHMS USED

A. Ridge Classifier

The **Ridge Classifier** is a linear model used for classification tasks. It is derived from **Ridge Regression**, which is primarily used for regression problems. In the Ridge Classifier, the regression output is converted into class labels based on the sign of the predicted value. Unlike **Logistic Regression**, which uses a probability-based approach and optimizes log-loss, the Ridge Classifier minimizes a **least-squares loss function** with **L2 regularization**. This regularization helps prevent overfitting by penalizing large coefficient values. The Ridge Classifier is particularly useful when dealing with **high-dimensional datasets**, such as text classification problems where the number of features may be very large.

Basic Principle

The Ridge Classifier treats classification as a regression task by minimizing the squared error loss with an L2 penalty. The optimization function can be written as:

L(w)=∑i=1n(yi−wTxi)2+α∥w∥22L(w) = \sum_{i=1}^{n} (y_i - w^T x_i)^2 + \alpha \|w\|_2^2L(w)=i=1∑n(yi−wTxi)2+α∥w∥22

Where:

- xix_ixi – input feature vector
- yiy_iyi – actual class label
- www – weight vector
- nnn – number of training samples
- α\alphaα – regularization parameter controlling the penalty strength

The model predicts a continuous value, and the **sign of the prediction determines the class label**.

B. Convolutional Neural Networks

A **Convolutional Neural Network (CNN)** is a deep learning architecture specifically designed for processing grid-structured data such as images. CNNs are widely used in applications like **image classification, object detection, and image segmentation**.

CNNs consist of several layers that automatically learn hierarchical features from input images.

Key Components of CNN

1. Convolution Layer

The convolution layer applies small filters (kernels) to the input image. These filters slide across the image to detect features such as edges, textures, and patterns.

2. Activation Function

After convolution, an activation function such as **Rectified Linear Unit (ReLU)** is applied. ReLU sets negative values to zero and keeps positive values unchanged, helping the network learn complex patterns.

3. Pooling Layer

Pooling layers reduce the spatial size of the feature maps, which helps decrease computational complexity. Two common pooling techniques are:

- **Max Pooling** – selects the maximum value in a region
- **Average Pooling** – calculates the average value of the region

4. Fully Connected Layer

The final layers in a CNN are fully connected layers that combine all the extracted features and perform classification. These layers produce the final prediction based on the learned features. CNNs are powerful because they automatically learn relevant features from raw images without requiring manual feature extraction.

C. Natural Language Processing

Natural Language Processing (NLP) is a branch of artificial intelligence that enables computers to understand, interpret, and generate human language. NLP combines techniques from **linguistics, computer science, and machine learning** to analyze textual data. NLP is used in applications such as **sentiment analysis, text classification, machine translation, chatbots, and question-answering systems**.

Traditional NLP Techniques

Earlier NLP approaches represented text using numerical features.

TF-IDF

TF-IDF (Term Frequency–Inverse Document Frequency) measures the importance of a word in a document relative to a collection of documents. Words that appear frequently in a document but rarely in other documents receive higher scores.

Word Embedding Methods

Word embedding techniques convert words into dense numerical vectors that capture semantic relationships.

- **Word2Vec** learns word relationships by predicting surrounding words in sentences.
- **GloVe** generates word vectors using global word co-occurrence statistics.
- **FastText** improves word embeddings by representing words as character n-grams, allowing the model to understand unseen words.

RESULTS

A. System Implementation

The proposed system was successfully implemented to provide an interactive platform that supports communication between deaf and non-deaf individuals. The system integrates multiple modules, including sign language recognition, natural language processing, chatbot interaction, sign language avatar generation, and an interactive training module. These modules work together to process user input, interpret gestures or speech, and generate meaningful responses in both textual and visual sign language formats. The system was developed and tested on a standard computing environment and demonstrated efficient real-time performance. The modular architecture enables smooth communication between components while maintaining system stability and scalability.

FIGURE 7: SYSTEM IMPLEMENTATION

B. Sign Language Recognition Performance

The sign language recognition module was evaluated based on its ability to accurately detect and classify hand gestures using the Ridge Classifier model. The system captures hand landmarks and gesture movements using computer vision techniques and processes them to identify the corresponding sign language gestures.

The model performed effectively in recognizing hand shapes, orientations, and movement patterns under different lighting conditions and user variations. The experimental results show that the Ridge Classifier provides reliable gesture classification with fast processing speed, making it suitable for real-time applications. The recognition module successfully converts hand gestures into machine-readable commands, which are then used for further processing within the system.

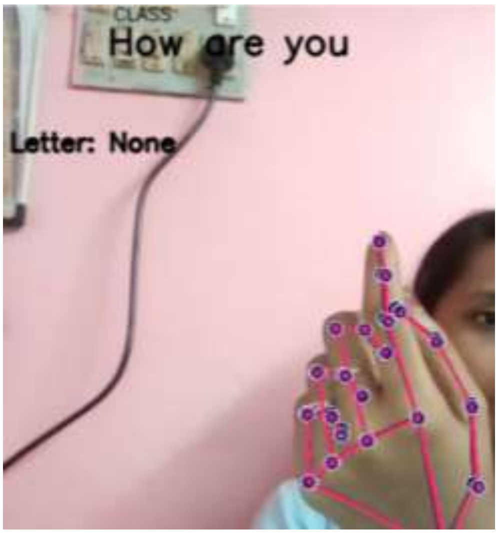

FIGURE 8: SIGN LANGUAGE RECOGNITION

C. Natural Language Processing and Chatbot Interaction

The Natural Language Processing (NLP) module was tested to evaluate its ability to understand user inputs and generate appropriate responses. The chatbot engine analyzes the recognized text and identifies the intent and context of the input. Based on this analysis, the system produces relevant responses that maintain a natural conversational flow. The chatbot supports both simple queries and more complex conversational interactions. During testing, the system demonstrated effective response generation and improved interaction quality. The integration of machine learning techniques further enhances the chatbot's ability to adapt to different user inputs and improve communication efficiency.

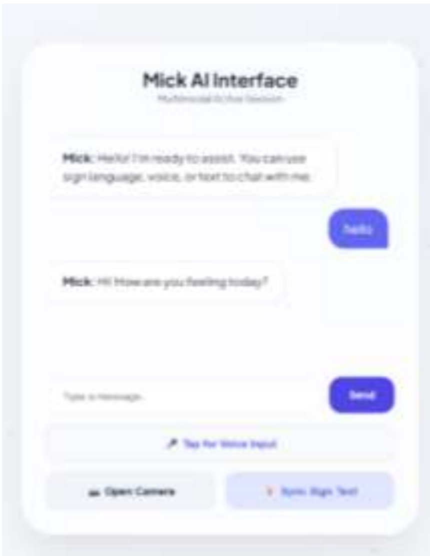

FIGURE 9 : CHATBOT INTERACTION

D. Sign Language Avatar Output

The sign language avatar module was evaluated based on its ability to convert chatbot responses into visual sign language gestures. The avatar generates animated hand movements that represent the

corresponding sign language output. The gestures produced by the avatar were designed to resemble natural sign language expressions, enabling deaf users to easily understand the system's responses.
The animation module ensures smooth transitions between gestures and maintains synchronization with the generated text output. The results show that the avatar effectively enhances communication accessibility by providing a clear and visual representation of the responses.

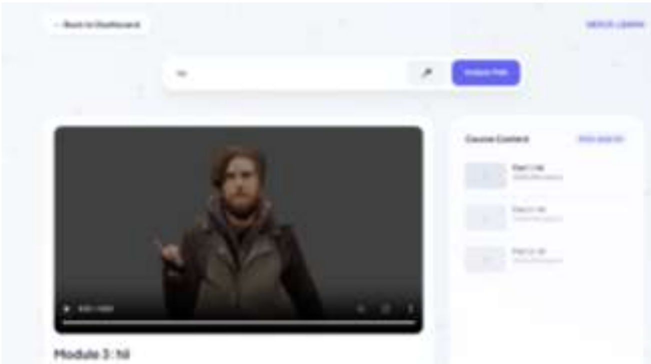

FIGURE 10 : AVATAR INTERACTION

E. Interactive Training Module Evaluation

The interactive training module was developed to help users learn and practice sign language through visual demonstrations and guided interaction. The module allows users to input text or speech and observe the corresponding sign language gestures generated by the system. During testing, the training module successfully demonstrated various sign language gestures and provided users with a structured learning experience. The system also offers feedback mechanisms that help users correct their hand gestures and improve their signing accuracy. This feature supports both beginners and advanced learners, making the system useful as both a communication tool and an educational platform.

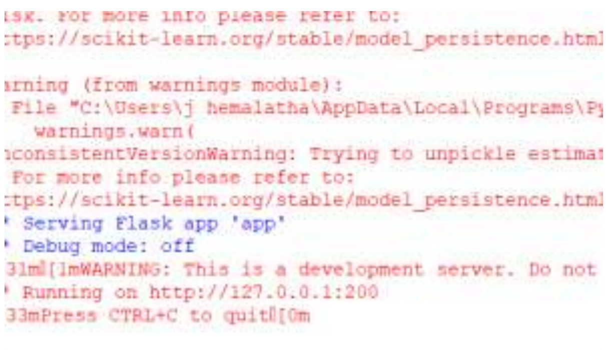

FIGURE 11: TRAINING MODULE

CONCLUSION

A new chatbot for sign language tackles the challenge of helping deaf and hard-of-hearing people communicate more fully.[3] [5]. Instead of relying on old methods, it uses smart software that sees and understands hand movements as they happen. Because it responds right away, conversations feel smoother and closer to natural talk. Alongside the main tool lives an animated signer that moves in real time, while a built-in practice section helps users learn by doing. Far from being only about messaging, it doubles as a way to grow skills in signing. Progress like this doesn't just connect people today - it opens doors to better understanding tomorrow. One step at a time, it could weave into different tools people already use. This opens doors - helping those who hear and those who do not connect more easily. Picture clearer chats shaping a world where fewer feel left out. Down the road, adding new tongues and deeper ways to interact might just widen that door even more.

REFERENCES

1. J. J. Godfrey, E. C. Holliman, and J. McDaniel, “SWITCHBOARD telephone speech corpus,” 1992.
2. T. Westeyn, H. Brashear, A. Atrash, and T. Starner, “Georgia Tech gesture toolkit: Supporting experiments in gesture recognition,” in *Proc. 5th Int. Conf. Multimodal Interfaces*, 2003, pp. 85–92.
3. T. Shanableh, K. Assaleh, and M. Al-Rousan, “Spatio-temporal feature extraction techniques for isolated gesture recognition in Arabic sign language,” *IEEE Trans. Syst., Man, Cybern.*, vol. 37, no. 3, pp. 641–650, Jun. 2007.
4. V. Athitsos et al., “The American Sign Language Lexicon Video Dataset,” in *Proc. IEEE Conf. Computer Vision and Pattern Recognition Workshops*, 2008, pp. 1–8.
5. K. Assaleh, T. Shanableh, M. Fanaswala, F. Amin, and H. Bajaj, “Continuous Arabic sign language recognition in user-dependent mode,” *J. Intell. Learn. Syst. Appl.*, vol. 2, no. 1, pp. 19–27, 2010.
6. T. Shanableh and K. Assaleh, “User-independent recognition of Arabic sign language for facilitating communication with the deaf community,” *Digital Signal Process.*, vol. 21, no. 4, pp. 535–542, 2011.
7. L. Bradeško and D. Mladenić, “A survey of chatbot systems through a Loebner prize competition,” in *Proc. Slovenian Language Technologies Society Conf.*, 2012.
8. M. F. Tolba, A. Samir, and M. Abul-Ela, “A proposed graph matching technique for Arabic sign language continuous sentences recognition,” in *Proc. 8th Int. Conf. Informatics Systems*, 2012, pp. 14–20.
9. B. ChandraKarmokar, K. M. Rokibul Alam, and M. Kibria Siddiquee, “Bangladeshi sign language recognition employing neural network ensemble,” *Int. J. Comput. Appl.*, vol. 58, no. 16, pp. 43–46, Nov. 2012.
10. P. Baldi and P. J. Sadowski, “Understanding dropout,” in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 26, 2013.
11. A. V. Nair and V. Bindu, “A review on Indian sign language recognition,” *Int. J. Comput. Appl.*, vol. 73, no. 22, pp. 33–38, 2013.
12. I. Sutskever, O. Vinyals, and Q. V. Le, “Sequence-to-sequence learning with neural networks,” in *Proc. Adv. Neural Inf. Process. Syst.*, 2014.
13. M. Mohandes, S. Aliyu, and M. Deriche, “Arabic sign language recognition using the Leap Motion controller,” in *Proc. IEEE Int. Symp. Industrial Electronics*, 2014, pp. 960–965.
14. S. Wiesler et al., “RASR/NN: The RWTH neural network toolkit for speech recognition,” in *Proc. IEEE Int. Conf. Acoustics, Speech and Signal Processing*, 2014.
15. K. Tripathi and N. B. G. C. Nandi, “Continuous Indian sign language gesture recognition and sentence formation,” *Procedia Computer Science*, vol. 54, pp. 523–531, 2015.
16. N. Tubaiz, T. Shanableh, and K. Assaleh, “Glove-based continuous Arabic sign language recognition,” *IEEE Trans. Human-Machine Systems*, vol. 45, no. 4, pp. 526–533, Aug. 2015.
17. L. Pigou, S. Dieleman, P. Kindermans, and B. Schrauwen, “Sign language recognition using convolutional neural networks,” in *Proc. European Conf. Computer Vision Workshops*, 2015.

18. O. Koller, H. Ney, and R. Bowden, "Deep learning of mouth shapes for sign language," in *Proc. IEEE Int. Conf. Computer Vision Workshops (ICCVW)*, 2015, pp. 85–91.
19. M. T. Ribeiro, S. Singh, and C. Guestrin, "'Why should I trust you?' Explaining the predictions of any classifier," in *Proc. ACM SIGKDD Int. Conf. Knowledge Discovery Data Mining*, 2016, pp. 1135–1144.
20. Y. Li et al., "DailyDialog: A manually labeled multi-turn dialogue dataset," 2017.
21. K.-J. Oh, D. Lee, B. Ko, and H.-J. Choi, "A chatbot for psychiatric counseling in mental healthcare service," in *Proc. IEEE Int. Conf. Mobile Data Management*, 2017, pp. 371–375.
22. A. Xu, Z. Liu, Y. Guo, V. Sinha, and R. Akkiraju, "A new chatbot for customer service on social media," in *Proc. CHI Conf. Human Factors in Computing Systems*, 2017.
23. K. Shenoy, P. V. R. Rao, and S. N. Kumar, "Real-time Indian sign language recognition using computer vision techniques," *Procedia Computer Science*, vol. 125, pp. 681–688, 2018.
24. J. Huang, W. Zhou, Q. Zhang, H. Li, and W. Li, "Video-based sign language recognition without temporal segmentation," in *Proc. AAAI Conf. Artificial Intelligence*, 2018.
25. D. Guo, W. Zhou, H. Li, and M. Wang, "Hierarchical LSTM for sign language translation," in *Proc. AAAI Conf. Artificial Intelligence*, 2018.
26. N. C. Camgoz, S. Hadfield, O. Koller, H. Ney, and R. Bowden, "Neural sign language translation," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2018, pp. 7784–7793.
27. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using Siamese BERT networks," 2019.
28. Q. Chen, Z. Zhuo, and W. Wang, "BERT for joint intent classification and slot filling," 2019.
29. T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating text generation with BERT," 2019.
30. H. V. Joze and O. Koller, "MS-ASL: A large-scale dataset and benchmark for understanding American sign language," 2019.
31. J. Pu, W. Zhou, and H. Li, "Iterative alignment network for continuous sign language recognition," in *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, 2019, pp. 4160–4169.
32. H. Zhou, W. Zhou, and H. Li, "Dynamic pseudo label decoding for continuous sign language recognition," in *Proc. IEEE Int. Conf. Multimedia and Expo*, 2019, pp. 1282–1287.
33. R. Rastgoo, K. Kiani, and S. Escalera, "Hand sign language recognition using multi-view hand skeleton," *Expert Systems with Applications*, vol. 150, 2020.
34. S. Aly and W. Aly, "DeepArSLR: A novel signer-independent deep learning framework for isolated Arabic sign language gestures recognition," *IEEE Access*, vol. 8, pp. 83199–83212, 2020.
35. Q. Xiao, X. Chang, X. Zhang, and X. Liu, "Multi-information spatial–temporal LSTM fusion continuous sign language neural machine translation," *IEEE Access*, vol. 8, pp. 216718–216728, 2020.
36. I. Papastratis, K. Dimitropoulos, D. Konstantinidis, and P. Daras, "Continuous sign language recognition through cross-modal alignment of video and text embeddings in a joint-latent space," *IEEE Access*, vol. 8, pp. 91170–91180, 2020.
37. B. Barbhuiya, R. Karsh, and R. Jain, "CNN based feature extraction and classification for sign language," *Multimedia Tools and Applications*, vol. 80, no. 2, pp. 3051–3069, 2021.
38. C. K. Lee et al., "American sign language recognition and training method with recurrent neural network," *Expert Systems with Applications*, vol. 167, 2021.

39. R. Elakkiya, P. Vijayakumar, and N. Kumar, "An optimized generative adversarial network based continuous sign language classification," *Expert Systems with Applications*, vol. 182, 2021.
40. A. Halder and A. Tayade, "Real-time hand gesture recognition using MediaPipe and machine learning," 2021.
41. M. Al-Qurishi, T. Khalid, and R. Souissi, "Deep learning techniques for sign language recognition: A survey," *IEEE Access*, vol. 9, 2021.
42. B. Natarajan et al., "Development of an end-to-end deep learning framework for sign language recognition, translation, and video generation," *IEEE Access*, vol. 10, 2022.
43. H. Luqman, "An efficient two-stream network for isolated sign language recognition using accumulative video motion," *IEEE Access*, vol. 10, 2022.
44. S. Wang, J. Li, and X. Zhang, "Sign language recognition using improved 3D-ResNet with enhanced hand features," *IEEE Access*, vol. 10, 2022.
45. T. Shanableh, "Two-stage deep learning solution for continuous Arabic sign language recognition," *IEEE Access*, vol. 11, 2023.
46. M. I. Saleem et al., "Machine learning based system supporting multiple sign languages for deaf and mute communication," 2023.
47. J. Shin et al., "Korean sign language recognition using transformer-based deep neural network," *Applied Sciences*, vol. 13, no. 5, 2023.
48. M. Amin Kuhail et al., "Assessing the impact of chatbot-human personality congruence on user behavior," *IEEE Access*, vol. 12, 2024.
49. Z. Chen et al., "LLM-assisted gloss-free sign language translation," 2024.