International Journal of Mathematical, Engineering and Management Sciences
Vol. 9, No. 4, 801-820, 2024
https://doi.org/10.33889/IJMEMS.2024.9.4.041

Ram Arti Publishers

Enhancing the Experience and Accessibility of Users with Disability by Integrating Voice Navigation into a Telemedicine Website

Sucheta V. Kolekar
Department of Information and Communication Technology,
Manipal Institute of Technology, Manipal Academy of Higher Education, 576104, Manipal, Karnataka, India.
E-mail: sucheta.kolekar@manipal.edu

Shreevasta Agnihotri
Department of Information and Communication Technology,
Manipal Institute of Technology, Manipal Academy of Higher Education, 576104, Manipal, Karnataka, India.
E-mail: ratnam2001shree@gmail.com

Divya Rao
Department of Information and Communication Technology,
Manipal Institute of Technology, Manipal Academy of Higher Education, 576104, Manipal, Karnataka, India.
Corresponding author: divya.r@manipal.edu

(Received on January 5, 2024; Revised on April 1, 2024 & April 29, 2024 & May 10, 2024; Accepted on May 15, 2024)

Abstract
As per the universal principles of usability and sustainable development goal defined for reducing inequality, each individual will have special abilities and challenges. Individuals with visual challenges have trouble interacting with digital platforms. In order to achieve inclusivity, there is a need to integrate universal accessibility into the web portals which are used as the most popular digital platforms. This study mainly focuses on the requirements of visually impaired users. The research work discusses the proposed approach for a telemedicine web platform by integrating voice navigation system. With this system, users can orally interact with the platform by using defined commands. Users can also receive the audio feedback from the computer for the specific command. The proposed speech recognition engine is implemented using deep learning models and tested on various browsers. The engine captures the commands in the form of user inputs and generates the proper audio feedback after executing the commands. Users with visual impairment have been involved in the evaluation by allowing them to interact on the telemedicine platform with verbal commands. The evaluation questions have been asked after each interaction to capture the response time, accuracy, experience and satisfaction. The outcome of the evaluation shows that individuals are showing significant progress in accessing required information and navigating the web pages on their choice of browsers. The study also observed that speech recognition engine along with speech grammar, acoustic model and synthesis has improved the usage of the system for all types of users. Ultimately, integrating voice navigation into web platforms provides a satisfactory experience to the users and improves accessibility, inclusivity and reduces inequality.

Keywords- Speech recognition, Natural language processing, Acoustic modelling, Language model, Viterbi algorithm.

1. Introduction

Technology has tremendously impacted our lives in various sectors, especially healthcare. Telemedicine- is one of the most popular digital platforms that provides medical care remotely. Web portals offer services of telemedicine platforms such as scheduling appointments with doctors from anywhere in the world and getting assistance for any medical issues. Though technology makes such platforms easily accessible, designing these platforms by considering diversified users and catering to their demands is necessary (Iyer et al., 2020). Visual impairment is one of the demands which affects people worldwide with low vision to a high degree of blindness (Chen et al., 2021). Such an individual faces various challenges while accessing the services on platforms and accessing the required content/options (Xue et al., 2024). Accessing

information or services is challenging, as well as using additional accessories such as a keyboard or mouse (Iyer et al., 2020). Since telemedicine is the most popular, many constraints can be identified, and solutions can be provided for inclusiveness and universal accessibility. To identify solutions, speech recognition technology is the most promising technology. This technology enables visually disabled people to interact with the platform. They can access the interface of the web portal without using a keyboard and mouse through spoken commands and audio feedback (Wei, 2024). Using speech synthesis technology, computerized speech can be generated for commands to access contents (Liu et al., 2004).

Users can receive the required information in the form of audio content and enhance their interaction experience. Speech synthesis technology is useful to remove the noise from voice commands to improve the accuracy of the system. The research work described in this article shows the design and development of voice navigation, which is a more inclusive and convenient experience on the portal.

Universal users with special disabilities have certain limitations to interact with the digital world. Designing voice navigation features can provide an effective solution to eliminate these limitations. The solution can provide direct opportunities for the users to be comfortable with the telemedicine framework. Integrating speech synthesis and speech recognition along with voice navigation into digital platforms is an essential principle of universal design (Swetha and Swami, 2021). Such platforms will be helpful in reducing barriers and increasing comfort in accessing information. Eventually, advancements in technology-based platforms can significantly improve standards, which will, in turn, enhance the quality of living and engagement with the digital world.

Successful integration of voice navigation enhances the telemedicine website for all universal users. The website can create a more comprehensive experience by avoiding usage of traditional input/output devices. The universal speech grammar module is implemented with algorithms to accept the voice commands that the system accepts to execute. The module receives the commands into spoken commands and converts them into executable instructions for processing. The module is also supported by speech synthesis to produce computer-generated audio information for users (Ra et al., 2023).

The research mainly focuses on implementing telemedicine websites for all users, regardless of disabilities. Through this work, we are creating an awareness of inclusivity in digital healthcare and promoting enhancements in usability and accessibility. While developing the following objectives, emphasis is given to the website's accuracy, efficiency, and response time.

- Develop a speech grammar module to define the list of English commands for interpretation.
- Design and develop a speech recognition module that interprets the list of English commands and converts them into text instructions for computers to execute.
- Design and implement a speech synthesis module for producing computerized audio in response to the defined English commands.
- Evaluate the system's accuracy, efficiency, and response time for each functionality of the voice grammar, recognition, and synthesis modules.

The research work further progressed with the scope of voice navigation in digital healthcare; the research includes extensive literature on existing research and provides the significance of speech recognition technology for universal applications. The framework has been designed for the telemedicine website to integrate voice navigation. The readers of this paper can get mindful information on speech recognition and synthesis modules in the implementation section. The result section discusses the execution and analysis of all the interactions. Further, the system is validated and tested by defining user and expert participated evaluation. This evaluation provides insight related to user feedback and the scope of improvement. Finally,

the study features how the telemedicine website is enhanced by integrating a voice navigation system and how such platforms can improve universal accessibility across a range of industry applications.

2. Related Work

This section discusses the relevant published literature similar to the scope of our study.

Chen et al. (2021) discussed the progress and uses of attention-based models in speech recognition in their work. They talk about the shortcomings of old models and present attention mechanisms as a remedy. The researchers examined different attention-based models and their performance in diverse speech recognition tasks. This work (Xue et al., 2024) presents a new approach created to reduce the negative impacts of speaker voice variations using normalization methods. It uses artificial speech inputs produced by a Text-to-Speech system and then enhances translation quality with the help of alignment adapters and normalized speech knowledge distillation modules. Wei (2024) study analyzes text-to-speech use in math assessment among students with disabilities, English language learners, and general education peers. Text-to-speech is employed more for complex math items by all students. However, the benefits are limited for students with disabilities and English language learners without extended time accommodations. The study therefore suggests that time constraints hinder its effectiveness in math problem-solving.

Conducted at Stanford University, this research (Levine et al., 2023) explores how speech-to-text aids students in writing more and potentially better compositions. Some students found this helpful in easing the cognitive load of composition. There were others that felt constrained by the tool or encountered technical issues. Special education designation correlated with the students' choice to use speech-to-text technology. Despite mixed student experiences, teachers were consistently positive towards aids for speech-to-text and planned their continued use in classrooms. This paper introduces a novel approach to discretization (Kharsa et al., 2024) using Bidirectional Encoder Representations from Transformer (BERT) models. The introduced work was evaluated and demonstrated superior performance in discretization. It even surpassed previous systems with significant error reductions. Achieving state-of-the-art results with a real-time discretization system, it outperformed existing tools in prediction accuracy and input format preservation. Wang et al. (2022) presented a novel approach to end-to-end speech recognition and understanding by optimizing the alignment of speech and language latent spaces. The work introduced a latent space alignment module performance compared to existing models on benchmark datasets and deemed superior. The paper is included in our review as the findings from this study have significant implications for improving speech recognition systems. The research work by Rao et al. (2017) used a variation of the Recurrent Neural Network Transducer model and performed end-to-end speech recognition on streaming data. Different models of architecture and training datasets were investigated in their study. State-of-the-art results were reported on the Libri Speech dataset. A novel streaming speech recognition dataset was released as part of their work. Another work (Tang and Lin, 2018) proposed a keyword-spotting system. Using deep residual network architecture, deep models were trained. Vanishing gradients, a problem commonly found in deep learning architectures, were avoided by using skip connections. Results showed the that proposed deep residual learning architecture had high accuracy in keyword spotting. They attained this with a low parameter count, making it ideal for devices with access to low resources. In their paper, Subedar et al. (2019) have proposed a deep learning approach for audiovisual activity recognition. By incorporating model uncertainty and Bayesian variational inference to estimate parameter posterior distributions the authors achieved state-of-the-art performance on benchmark datasets. Padmanabhan and Premkumar (2015) summarized research that used machine learning for automatic speech recognition. Their work covered limitations and advancements of traditional systems through machine learning.

Rattan et al. (2021) have proposed a new approach in the form of voice-assisted development, which allows the users of a cloud-hosted service to speak in a natural language and express their needs to the system. The authors have explored various advanced options for the dynamic generation of webpages and done time analysis using traditional methods to analyze the webpages. Torad et al. (2022), in their paper, discussed implementing a home automation system using voice commands on Android or web-based applications or via a chat form. An NLP-based command interpretation system was implemented on the intelligent controller.

This paper helped explore the future scope as voice commands can be integrated with small devices and provided with a web application for accessibility. Kamra et al. (2023) implemented an online quizzing system for blind people using modern voice recognition techniques. As there is an increase in the usage of online platforms for assessment, there is a need for a system for executing speech commands that can be followed to ensure the smooth conduction of online quizzes. The authors have implemented the system using MERN STACK technology. This paper is helpful to explore voice recognition techniques that can be useful to incorporate in the research. Sharma et al. (2022) have implemented a new speech-based mailing system for visually impaired people. The authors implemented basic functionalities such as composing, sending, receiving, deleting, and searching E-mails in the user's pre-registered Gmail account and integrated them into the web browser. This web application accesses Gmail using text-to-speech and speech-to-text modules to make emailing feasible for visual-impaired users. The authors have proposed many additional features that can be integrated. However, the system cannot be generic for any web application.

Barnwal et al. (2023) have designed and developed an Artificial Intelligence Enabled Voice Assistance System using Natural Language Processing. This system takes input as a voice signal and provides output in numerous ways, like voice and visual display. Authors claim that the system is faster and more effective in space management; however, reliability is the most critical challenge. This challenge makes researchers explore different technologies and algorithms for voice assistance implementation. Sadi et al. (2022) implemented a voice control e-commerce application in their research paper. Authors have used IBM's STT model to implement speech-oriented ordered goods purchased online business functionality. However, the authors failed to create a pleasant experience for the visually impaired users and incorporate a feedback mechanism in the system. In their research, the AI-based email voice assistant system was implemented by Maheswari et al. (2022). The authors have implemented message transfer agents to validate and execute each voice command related to email management. Authors have failed to make it generic for any email application and web browser.

Our literature survey has pinpointed crucial research areas. These include the need for a generic voice integration system, faster response times through intelligent technology, effective user feedback mechanisms, and the pursuit of a reliable model for consistent performance. These gaps are the focal points guiding our research to contribute valuable insights to voice technology integration.

3. Methodology

The overall methodology framework is designed to process users' voice commands and convert them into actions to be executed on the website. **Figure 1** shows a detailed implementation framework introduced in this study.

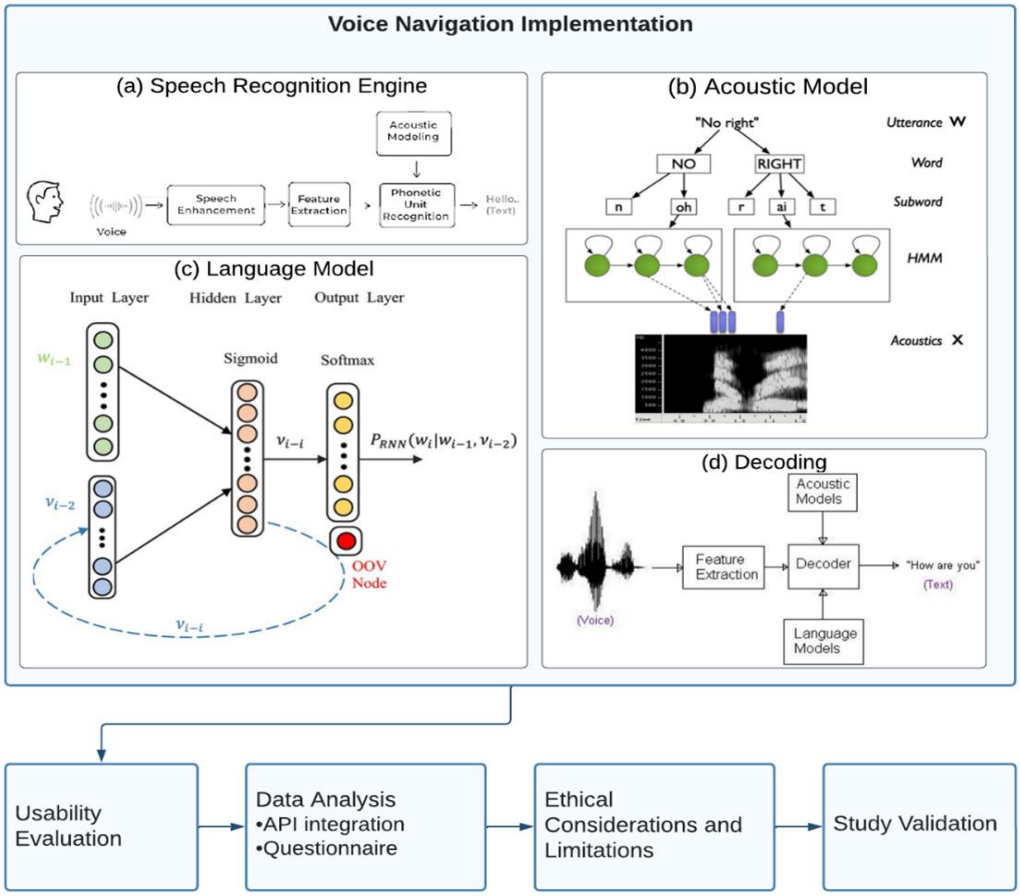

Figure 1. Methodology.

3.1 Voice Navigation Implementation

The universal design emphasis on accessibility principle in the user interaction and experience with any web portal. Most of the web portals are implemented with assumption of characteristics which are defined for normal users. In order to enhance the interaction and experience, there is a need to understand the characteristics of all the types of users. Implementation of voice navigation system is one of the solutions which is proposed and demonstrated in this research work. With the help of this solution, we can eliminate the traditional ways of interaction and create an alternate environment for especially abled or different types of users. This section mainly discusses phases such as speech recognition and synthesis engine, acoustic model, language model and decoding. The implementation is optimized by considering various parameters such as speech rate, ways of pronunciation, and variations in dialectal and computer literacy. Algorithms of each phase are fine-tuned with the parameters and experimented with many iterations to get better trained and tested.

3.1.1 Speech Recognition Engine

When a user speaks into a microphone, the computer's audio hardware con- verts the sound waves into electrical signals and digitizes them. These digitized audio data are then processed by the speech recognition engine in the browser. It can analyze the audio data and extract the features such as intensity, frequency and time taken to speak. Initially, the speech recognition engine is trained on large speech datasets to

identify the patterns between sounds and words. The training is done by implementing deep neural networks and hidden Markov models for multiple epochs. The identified patterns are used to extract the features which are important to map to the phonetic units (Palaz et al., 2019; Abdildayeva et al., 2023).

3.1.2 Acoustic Model

Voice navigation system is incomplete without implementation of acoustic model in speech recognition engine. The acoustic model has two important steps: feature extraction and phoneme sequence identification. Model captures spoken words as inputs and extracts the features such as frequency, duration, and intensity. Statistical models, such as Hidden Markov Models are implemented as a part of acoustic model to map the spoken words to closest phonetic units by referring to International Phonetic Alphabet database (Yuan et al., 2021). Initially, the model was trained using large datasets of speech for multiple iterations. HMM along with language models help to estimate the phonetic units with better accuracy (Cui et al., 2021). The detailed Acoustic modelling algorithm is explained in in Algorithm 1.

Algorithm 1. Acoustic model.

```
procedure PreprocessAudio(input_audio)
    Apply any necessary preprocessing steps to the audio
end procedure
procedure PreprocessText(transcript)
    Apply any necessary preprocessing steps to the transcript
end procedure
procedure ExtractFeatures(preprocessed_audio)
    features ← Extract acoustic features from the preprocessed audio
end procedure
procedure TrainAcousticModel(features, preprocessed_text)
    model ← Initialize the acoustic model
    Train the acoustic model using the extracted features and preprocessed
text
    return model
end procedure
procedure DecodeAudio(model, features)
    decoding_result ← Use the trained acoustic model to decode the audio
features
    return decoding_result
end procedure
procedure PostprocessTranscript(decoding_result)
    Apply any necessary postprocessing steps to the decoding result
end procedure
procedure EvaluateTranscript(transcript, ground_truth)
    Compare the transcript with the ground truth to compute accuracy or
other metrics
end procedure
Step 1: Preprocessing
PreprocessAudio(input_audio)
PreprocessText(transcript)
Step 2: Feature Extraction
features ← ExtractFeatures(preprocessed_audio)
Step 3: Training
model ← TrainAcousticModel(features, preprocessed_text)
Step 4: Decoding
decoding_result ← DecodeAudio(model, features)
Step 5: Postprocessing
transcript ← PostprocessTranscript(decoding_result)
Step 6: Evaluation
accuracy ← EvaluateTranscript(transcript, ground_truth)
Step 7: Output
print(transcript)
print("Accuracy:", accuracy)
```

3.1.3 Language Model

Probabilities must be assigned to the sequences of spoken words. This is useful for predicting context of words and their relationships. The language model is implemented to assign probabilities based on n-gram models. In this, probabilities are estimated based on previous n-1 words (Karita et al., 2019). Language model is supported by RNN or convolutional neural networks for more rational and smooth results (Li et al., 2019; Shewalkar et al., 2019). Language model is an essential step of natural language processing for capturing complex relationships between words in different contexts. The efficiency of the model can be increased by contextual perception (Torad et al., 2022). This contextual information is useful to assign probabilities to word sequences. It helps to solve errors related to homophone uncertainties as well. The Language model algorithm used in our work is shown in Algorithm 2.

Algorithm 2. Language model.

```
procedure PreprocessAudio(input_audio)
    Apply any necessary preprocessing steps to the audio
end procedure
procedure ExtractFeatures(preprocessed_audio)
    features ← Extract acoustic features from the preprocessed audio
end procedure
procedure RecognizeSpeech(features)
    acoustic_result ← Use the trained acoustic model to recognize speech from the features
    return acoustic_result
end procedure
procedure GenerateText(acoustic_result)
    language_result ← Use a language model to generate text based on the recognized speech
    return language_result
end procedure
procedure PostprocessTranscript(language_result)
    Apply any necessary postprocessing steps to the generated text
end procedure
procedure EvaluateTranscript(transcript, ground_truth)
    Compare the transcript with the ground truth to compute accuracy or other metrics
end procedure
Step 1: Audio Preprocessing
PreprocessAudio(input_audio)
Step 2: Feature Extraction
features ← ExtractFeatures(preprocessed_audio)
Step 3: Acoustic Modeling
acoustic_result ← RecognizeSpeech(features)
Step 4: Language Modeling
language_result ← GenerateText(acoustic_result)
Step 5: Postprocessing
transcript ← PostprocessTranscript(language_result)
Step 6: Evaluation
accuracy ← EvaluateTranscript(transcript, ground_truth)
Step 7: Output
print(transcript)
print("Accuracy:", accuracy)
```

3.1.4 Decoding

To produce the possible sequence of spoken words there is a need for a phase which will decode the words from audio input commands. The decoding phase is implemented by Viterbi Algorithm (Alhaidari and Zohdy, 2019). The main steps incorporated into the algorithm are acoustic modeling and language modeling. The algorithm has the structure of states which are divided into input, hidden and observed states. The sequence of words can be identified from hidden states by considering or feeding back the outcome of observed states. As we know the spoken words are getting converted into electrical signals, there is need to map each spoken word into phonetic units with the help of hidden states (Chavali et al., 2022).

Algorithm 3. Decoder using Viterbi algorithm.

Algorithm 4 Decoder using Viterbi Algorithm

procedure INITIALIZATION
    Initialize the decoder with any necessary data structures or parameters
end procedure
procedure VITERBIDECODE($features$)
    Initialize the trellis
    for each time step t **do**
        Calculate forward probabilities for each state at time step t
        Calculate backward pointers for each state at time step t
    end for
    Find the best path through the trellis based on forward probabilities and backward pointers
    return the best path
end procedure
procedure INITIALIZETRELLIS
    Initialize the trellis with the initial probabilities and transition probabilities
end procedure
procedure CALCULATEFORWARDPROBABILITIES(t)
    Calculate the forward probabilities for each state at time step t
end procedure
procedure CALCULATEBACKWARDPOINTERS(t)
    Calculate the backward pointers for each state at time step t
end procedure
procedure FINDBESTPATH
    Find the best path through the trellis based on the forward probabilities and backward pointers
end procedure
Main Code
Feature Extraction
$features \leftarrow extract\_features(input\_audio)$
Run the Decoder
$decoding\_result \leftarrow ViterbiDecode(features)$
Output the Decoding Result
print($decoding\_result$)
=0

In the Viterbi algorithm, each step estimated the probabilities of the current state by considering calculated probabilities of the previous states. The current state probability is calculated by considering the prior state information and current information. This is an iterative process which gets continued till the most probable

sequence of words not generated. The algorithm uses the mechanism to store maximum probability of each state in each iteration and allows to feed back the most likely sequence for optimization (Palaz et al., 2019). Viterbi algorithm is more popular and most likely getting used in the applications of natural language processing where the information needs to be determined from the hidden states using observed outcomes. Algorithm 3 discusses the steps of decoder and Viterbi in detail (Anza and Abdullahi, 2018).

3.2 Usability Evaluation

A rigorous usability evaluation was conducted to comprehensively assess the efficacy of the voice navigation system in enhancing website accessibility. The primary focus was to gauge visually impaired users' experience, efficiency, and satisfaction while navigating the telemedicine website through voice commands. An assumption underlying the system's scope is that it focuses on website navigation rather than information generation.

The data that is to be converted from text to speech is provided by applications accessed through the website. The evaluation process went through stages of participant selection, assessment question formulation, usability test setup, data collection, analysis, and interpretation of results. A group of visually impaired participants were selected. They were explained about the study and consent was sought for performing usability evaluation. Familiarity with technology, age group, and diversity of backgrounds were factors in selecting the user evaluation group. The reason for doing this was to ensure a sample that had opinions and experiences spanning different user perspectives. The participants' visual impairments ranged from partial to complete blindness, aligning with the intended target audience.

There are various ways of conducting the usability tests for the system. The most effective way is to conduct the test in a controlled environment as a simulation which will help participants to experience the functionalities in real-time. Participants are equipped with headphone devices and microphones of the computers to interact with the telemedicine platform. Participants were told to use the specific commands in order to test the voice navigation system. Once the set of interaction rounds are completed to test all commands, the assessment questions are provided in the audio format to capture the real-time feedback.

The parameters considered for the feedback are usability, efficiency and satisfaction by rating from 1 to 10 scale where 1 specified as low score and 10 as a high score. In this scenario, simulation is supported with actual functionalities of the portal.

Assessment Questions

1) Rate the ease of using voice commands to navigate through different sections and pages of the website. (Rate on a scale of 1-10).
2) Rate the accuracy of the voice navigation system in understanding and interpreting your commands. (Rate on a scale of 1-10).
3) Rate the effectiveness of the voice navigation system in assisting you with scrolling, zooming, and button navigation. (Rate on a scale of 1-10).
4) Rate how independent and empowered you felt while using the voice navigation system to browse the website. (Rate on a scale of 1-10).
5) Rate the clarity and helpfulness of the audio feedback or assistance provided by the voice navigation system. (Rate on a scale of 1-10).
6) Rate your satisfaction with the voice navigation integration in terms of accessibility, usability, and user experience. (Rate on a scale of 1-10).

3.3 Data Analysis

Usability evaluation step has generated lot of data which was collected through quantitative and qualitative feedback. Participants were given the option to answer each question by providing their honest feedback in using voice navigation system on the portal. The questions were prepared for capturing participants' experience, suggestions and requirements for improvement in the form open-ended questions. The responses are captured in an iterative manner by allowing participants to complete each functional task. The responses to questions have been captured and analyzed using thematic analysis. The thematic analysis suggests defining themes as per responses to the questions. The themes can be generated by considering initial codes assigned to each question. The themes have been defined, reviewed and refined for better results. Thematic analysis expects various levels of stages such as familiarization with the functionalities, questions and assigning initial codes. The comprehensions generated from qualitative feedback have been taken into consideration for enhancing the performance of the system. It provides detailed context of the participant's experience by analyzing the ratings and demonstrations.

The performed qualitative analysis has been validated through statistical methods for quantitative analysis. Mean and standard deviation formulas are used for all the responses given to the specific questions. Statistical analysis provides quantitative impression of responses and manages deviation in responses. Apart from responses to qualitative questions, the data is collected for time to complete tasks and frequency of errors. These two additional parameters are useful to understand the real performance of voice navigation system and efficiency of participants. To improve the structure of each functionality, the descriptive statistics have been defined and common usage of patterns have been identified for improvement. The statistical analysis provides a detailed description of participants' experiences for various usability scenarios. The results of qualitative and quantitative have been combined through triangulation approach. With this, the themes of qualitative analysis have been refined and overall experience has been captured. Further, the performance of the platform has been analyzed by understanding consistent patterns and inconsistencies.

The choice of browser is the main impacting factor in identifying the performance of voice navigation system. The speech recognition engines are designed by taking into consideration the settings of different browsers. The integration of such engines is creating strange effects on participant's real time experience by affecting accuracy, response time and satisfaction. In order to handle this variation in browsers, the qualitative analysis has been performed and analyzed association between participants' insights and speech recognition engine for different settings of browsers.

The qualitative analysis gives clarity on the performance of recognition engine as a browser APIs. It also shows how the interaction experience is enhanced during execution of functionalities and retrieving information. By combining qualitative and quantitative analysis, clear understanding has been developed on how to choose a better browser, how to upgrade the engine functionality and how to enhance user experience.

The analysis of feedback offers a comprehensive view of the voice navigation system's role in improving accessibility of a website. By integrating insights from various perspectives, the findings were validated. Different browsers use various speech recognition engines and APIs to enable speech recognition. Google Chrome, for instance, uses Google's Web Speech API and a speech recognition engine based on deep neural networks. Microsoft Edge utilizes the Windows Speech Recognition engine, an integral part of Windows 10. Mozilla Firefox employs the Web Speech API and the Sphinx speech recognition engine, an open-source toolkit. Like Google Chrome, Opera leverages Google's Web Speech API and speech recognition

engine. Internet Explorer lacks built-in speech recognition capabilities. Brave adopts the speech recognition engine and API of the user's default browser, Google Chrome or Firefox.

3.4 Ethical Considerations and Limitations

There was a stringent and strong ethical framework designed for this study. This formed the basis of the usability evaluation process. It was important to ensure the well-being and rights of the participants involved. The researchers adhered rigorously to established ethical guidelines. Approval was sought to ensure compliance with ethical standards. All participants were informed about the nature of the study, the tasks involved, the potential risks, and their rights to withdraw at any point without repercussions. Consent forms were designed in accessible formats. This was done to accommodate participants with varying levels of visual impairment. They provided informed consent before participating in the usability evaluation. were explicitly informed. Participants were given time to clarify their concerns before providing their consent.

The data collected during the usability evaluation were treated with respect for participants' privacy. Confidentiality measures were in place to protect participant identities. Participant identifiers were anonymized during data analysis. Data was stored securely in compliance with data protection regulations.

Screen readers, voice-over software, and other assistive technologies facilitated seamless interaction with the voice navigation system. Participants were offered technical support. This was done to ensure they could engage with the study completely. Though a lot of effort was put to conduct a robust usability evaluation, certain limitations should be acknowledged. Integrating voice-based authentication in scenarios involving sensitive medical information can be tricky.

In the methodology section, implementation of voice biometrics would involve several key steps. Capturing and storing voice samples securely would be crucial. We have to ensure they are encrypted both during transmission and storage. Voice biometric algorithms then analyze unique vocal characteristics such as pitch, tone, and rhythm to create a digital voiceprint for each user. These voiceprints were securely stored and compared during authentication, utilizing advanced encryption techniques to protect against unauthorized access. Multi-factor authentication could be employed, combining voice biometrics with other factors such as passwords or biometric identifiers for added security. Regular audits and updates to the voice biometric system would be necessary to address evolving security threats and ensure compliance with privacy regulations. By implementing these measures, the browser extension can provide a robust layer of security and privacy protection, especially in contexts involving sensitive medical information.

Though the participants of the study were diverse in age and technology access, the small sample size used in the study is not comprehensive enough to generalize the findings. This is an initial pilot study that can be reproduced across a broader population of visually impaired users. There is a possibility in the study design that participants' experiences could have been influenced by the specific browser and device they used. This is because the interaction of the user is not only with the extension but also with the parent browser. For instance, a person who found difficulty in certain features of the extension may be using a particular browser they are not used to. With reduction in familiarity comes the reduction in perceived performance. We acknowledge the potential confounding variables which exist that could impact the consistency of results. Efforts have been made to recreate real-world scenarios. However, the controlled environment of usability testing might not fully mirror the nuances of users' daily interactions with telemedicine websites. Participants' performance and perceptions during testing may differ from their experiences in a natural context. The usability evaluation captured participants' interactions with the voice navigation system over a relatively short duration. Long-term use patterns and potential adaptation effects might not have been fully explored within the study's timeframe. The reliance on self-reported ratings and

qualitative feedback introduces the potential for bias. Participants' responses might be influenced by their expectations, prior experiences, or personal preferences. We also have to keep into account that participants may have been cautious about offering candid feedback due to their awareness of the study's objectives.

3.5 Study Validation

The usability evaluation through validation of the study provided a comprehensive assessment of work. Impact of website accessibility was assessed via a combination of methods described below. Participants provided ratings for each assessment question. This was the quantitative data collected in order to quantify subjective experience with the voice navigation system. For quantitative data, mean ratings for each assessment question were computed. This helped to get an overview of participant satisfaction across usability dimensions. Standard deviations were computed on the quantitative questionnaires to determine the variation in responses. Quantitative data provides an overview of participant satisfaction.

Qualitative feedback was collected via open-ended questions for participants to share thoughts, suggestions, and areas for improvement. Qualitative data was systematically reviewed and categorized to identify common themes and concerns. Constructive feedback offered was captured. Qualitative data provides insights into interaction patterns and challenges faced by them.

Usability evaluation provides the opportunity to derive inferences based on the generated results. The first inference is related to scoring the performance of the telemedicine website by emphasizing accessibility parameters. Second is capturing the suggestions and recommendations from the participants to enhance the accuracy, clarity and user experience. The defined inferences are important to incorporate into the implementation of websites and make them available for more universal users.

While targeting universal accessibility, there is a chance of increasing vulnerabilities in the system due to external integration. However, voice navigation systems can be used to enhance security in terms of authentication and privacy, especially when accessing sensitive information such as medical records. While capturing voice commands, the sound of users can be captured and trained as a voice biometric. The voice biometric technique considers various parameters such as unique tones, pitch sound quality, rhythm, pauses etc. to create voice profile for each user. The data of these parameters can securely store and be used for authorization purposes. The voice profile can be used to enhance authentication by integrating in multi-factor authentication. In this scenario, regular auditing of voice quality needs to be captured regularly. By implementing this security mechanism, the browser can have inbuilt security and privacy protection.

Privacy, security, and anonymity in speech data is critical for building user trust. This is ensured by first minimizing the collection of speech data. Only data essential for functionality is collected. This data is anonymized and stripped of personal identifiable information using hashing or tokenization. All communication between the extension and servers is encrypted. Users must consent to data collection and have the option of opting out or deleting their data. Secure storage of data is essential, along with regular audits for compliance with privacy regulations. Users must have control over access and deletion of their data and a response plan for data breaches must be in place. Through these measures, developers can uphold user privacy while delivering the intended functionality of the browser extension.

4. Implementation

The JavaScript-based work aims to facilitate the integration of speech recognition and synthesis into web applications. It provides a user-friendly interface for web developers to access speech recognition and synthesis engines that are integrated into web browsers. Using this research work, developers will be able to define speech grammar and speech vocabulary. Doing this will positively improve speed and accuracy

of speech recognition tasks. The interface allows control over audio input and output. The interface also provides callbacks for recognition events. The work simplifies the process of incorporating speech recognition into web applications by offering an intuitive interface. Developers can customize settings and respond efficiently to speech recognition events.

4.1 Speech Grammar

The WebSpeech API has a 'SpeechGrammar' interface that allows in t allows us to define a set of words or word patterns we want the speech recognition service to recognize. A grammar is created using the SpeechGrammar interface. This is then formatted with J Speech Grammar Format (JSGF). Once the format is set, the speech recognition system listens to these rules. Grammar consists of rules that specify what a user can say to trigger a specific response from the system. For example, if we want the system to respond to the phrase "scroll up," we will define a rule that triggers required action on utterance of the command. Words and phrases that make up the grammar can be customized or also used in combination to form more complex utterances. Words represent individual terms like "scroll" or "up," while phrases are combinations of words such as "scroll up" or "go to the top of the page." Special symbols and operators are used to create rules. The "—" symbol indicates that the user can say any of the listed words or phrases, like "scroll up," "move up," or "go up." The "*" symbol denotes that a word or phrase is optional. Once the grammar is defined in JSGF, it is used by the speech recognition system. The system compares the user's speech to the defined rules in the grammar. If a match is found, the associated action is executed.

4.2 Speech Recognition

The sequence of steps of creation and composition of Speech Recognition objects are described here. In the first step, a new instance of the 'Speech Recognition' object is created. Properties, such as the language and maximum number of alternatives are then set. Following this, the application starts the recognition process by calling 'start()'. During the recognition process, several events can be triggered. The 'audio start' event is fired when the user agent begins capturing audio. The 'sound start' event is fired when the user starts speaking and sound is detected. When speech is detected, the 'speech start' event is fired. The 'speech end' event is fired when the user stops speaking. The 'sound end' event is fired when no more sound is detected. The 'audio end' event is fired when the user agent finishes capturing audio. If the recognition process is successful, the 'result' event is fired. The 'Result' object contains a list of 'Speech Recognition Result' objects. Each 'Speech Recognition Result' object contains a list of 'Speech Recognition Alternative' objects, representing the different possible transcriptions of the speech input. If there is an error during the recognition process, the 'error 'event is fired and contains a 'Speech Recognition Error' object, which provides information about the error. The 'Speech Recognition' object also provides several methods that can be used to control the recognition process. The 'stop()' method stops the recognition process. The 'abort()' method cancels the recognition process. The 'listening' property returns a Boolean value indicating whether the user agent is listening for speech.

4.3 Speech Synthesis

'Speech Synthesis()' API is developed as part of this work. This API enables web developers to generate text-to-speech output. The API supports diverse voices and speech in various languages. This feature allows a wide range of users with diverse backgrounds to use the app. It also enables developers to incorporate customized speech output into their web applications. Spoken instructions or responses to user input can be interpreted and manipulated. As part of the Speech Synthesis() API usage, Speech Synthesis Utterance is created. This object holds the text to be spoken, the desired voice, and relevant parameters. Once the utterance object is established, it is added to the speech queue using the Speech Synthesis. Speak() method. Several events throughout the speech synthesis process track its progress. The events include 'start', 'end', 'pause', 'resume', and 'boundary'. These events are used by developers to either trigger actions or update

the user interface based on the current state of speech synthesis. The Speech Synthesis() API also provides access to information about available voices. Properties such as name, language, and gender are parsed from the input voice. This information allows users to select their preferred voice for speech output.

4.4 Command Categorization

This work has integrated voice navigation into various browsers. This is performed with the help of a categorization system. The system categorizes by word length (from 2-4 length words). The accuracy is measured based on word length. The accuracy of 2 words length commands is computed differently from 3-word length commands etc. With this approach, the assessment of the accuracy of task completion based on command syntax is measured. Valuable insights are gained to improve the performance and reliability of the voice navigation system.

1) ***2-word length commands*** - Scroll Up, Scroll Down, Go Up, Go Down, Read Content, Zoom In, Zoom Out, etc.
2) ***3-word length commands*** -Increase Font Size, Decrease Font Size, Default Window Size.
3) ***4-word commands*** - Go to the Video Consult page, etc.
4) ***Miscellaneous*** - Go to the Sign-Up page.

5. Results and Analysis

Significant effort was put in to carry out this research. This section describes the implementation of key functionalities using voice commands elaborated in the methodology. Users with visual disability can utilize voice commands to perform various essential actions on websites. The implementation process involved designing and integrating three modules: speech grammar, speech recognition, and speech synthesis. The actions, the commands, and the respective results are described below. Users will be equipped to listen to the website content and fill out forms using voice input. These features greatly enhance accessibility, make users navigate websites better, and access information more effectively. Throughout the process, accuracy and responsiveness were evaluation and performance metrics.

5.1 Scrolling

The scrolling feature utilizes two voice commands: "Scroll Up" and "Scroll Down." These commands are used to move the page either up or down and are intuitive to use. The results of these commands can be observed at different stages: before and after they are executed.

5.2 Changing Window Size

Commands have been created to alter the window size. These commands allow users to adjust the zoom level and reset the window to its default size. You can see the effect on the web screen by using the phrases "Zoom In" and "Zoom Out." For the sake of completeness, two additional commands, "increase font size" and "decrease font size," have been designed. This increases text readability by making the font larger or smaller. This works differently from zoom as the window size remains constant.

5.3 Tab Navigation

Navigation between tabs is facilitated by commands like “Go to page_name.” These commands enable users to move between various pages on the website. The effect of these commands can be experienced without the need for figures.

5.4 Read Content

Executing the “Read Content” command on each web page initiates the system to read the contents aloud, benefiting particular users. This feature is designed to provide an auditory experience without including figures.

5.5 Form Filling

Form filling is made accessible for especially abled users, particularly on the Sign Up page, through voice commands. The system validates and provides feedback during the form-filling process without including figures. While demonstrated on the sign-up page, this feature is extendable to other pages requiring user input.

5.6 Find Text, Page Navigation and Search

The user can find specific text on the screen using the "Find" command, followed by the desired text. Commands like "Go Up, Go Down, Go Right, Go Left, and Select" allow general navigation through the web page. The "Search" command can locate doctors or specific items. These functionalities are explained without figures, although they were demonstrated on our demo portal for symptoms or medicines.

5.7 Validation of Voice Navigation System

The integration of voice navigation components improves the user-friendly experience for visually impaired people. The impact of these components is evaluated using some essential metrics related to the execution of every voice command. These evaluation metrics are quantitative measures used to assess the performance or effectiveness of a system. These metrics provide objective and standardized ways to evaluate various aspects of a system's performance, allowing for comparisons, improvements, and decision-making based on the results. We defined some standard evaluation metrics to validate the performance of all voice commands. Here, each voice command has been executed by 15 different visually impaired subjects, and the values have been measured to measure.

6. Evaluation Metrics to Perform Validation

The validation is performed using User Testing with 15 visually impaired users to understand the system's ease of use. The metrics are defined, and values are obtained by observing the user's performance for each command.

- ***Accuracy/Task completion rate*:** The ratio of correct voice command execution to total voice commands given.
- ***Error rate*:** The ratio of incorrect voice command execution to total voice commands given.
- ***Time to complete task*:** Average time all users take to complete each command.
- ***Cross browser compatibility*:** The project is compatible across different browsers.

6.1 Evaluation Result of Voice Commands

The evaluation categorizes voice commands based on word length: 2-word, 3-word, and 4 to 5-word. Each command is executed multiple times by 15 users, and **Tables 1**, **2**, and **3** present accuracy, error rates, and average execution times across different browsers. For 2-word commands, which include scrolling, zooming, and reading content, **Table 1** details accuracy, error rates, and average times. **Figure 2** displays a bar graph summarizing accuracy (81.9%) and error rates (18.1%) for all 2-word commands. 3-word commands, addressing window size, font size, and homepage navigation, are outlined in **Table 2**, showcasing accuracy, error rates, and average times. A corresponding bar graph is presented in **Figure 2**. Commands with four and 5-word lengths involving page switching are documented in **Table 3**, presenting accuracy, error rates, and average times. **Figure 2** also visually represents the data, showing an accuracy of 61.3% and an error rate of 39.7%.

The importance of accuracy is emphasized through **Figure 3**, a pie chart comparing accuracy rates across all categories based on user testing results."

Table 1. Accuracy and error rate for 2-word length commands.

Commands	Scroll up	Scroll down	Zoom in	Zoom out	Go up	Go down	Read content
Accuracy	11/15	9/15	13/15	12/15	14/15	14/15	13/15
Error Rate	4/15	6/15	2/15	3/15	1/15	1/15	2/15
Avg. Time	45 sec	55 sec	43 sec	54 sec	67 sec	75 sec	142 sec

Table 2. Accuracy and error rate for 3-word length commands.

Commands	Default window size	Increase font size	Decrease font size	Go to homepage
Accuracy	12/15	11/15	9/15	10/15
Error Rate	3/15	4/15	6/15	5/15
Avg. Time	65 sec	75 sec	53 sec	77 sec

Table 3. Accuracy and error rate for four and 5-word length commands.

Commands	Go to the video consult page	Go to the find doctors page	Go to medicine page	Go to the lab test page	Go to surgeries page
Accuracy	10/15	8/15	9/15	10/15	9/15
Error Rate	5/15	7/15	6/15	5/15	6/15
Avg. Time	75 sec	95 sec	87 sec	94 sec	95 sec

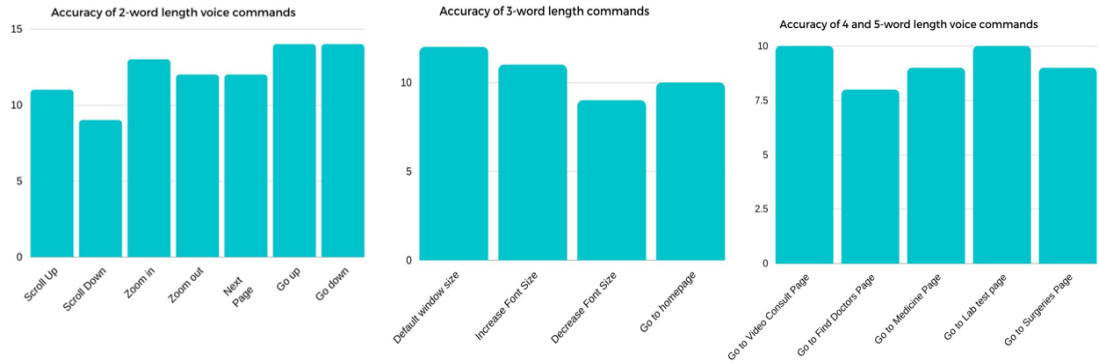

Figure 2. Bar graph for accuracies of word commands.

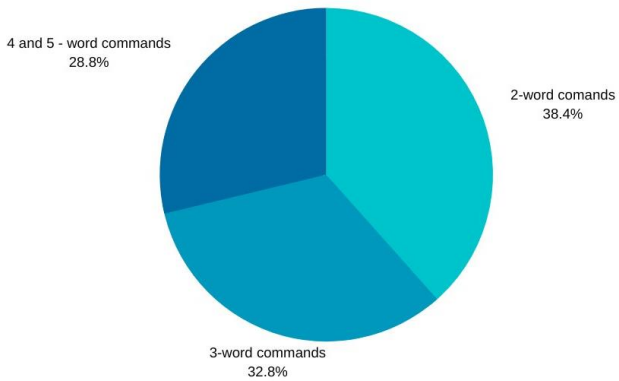

Figure 3. Pie Chart comparing the accuracy rates among all categories.

6.2 Expert and Participation-based Usability Testing

It evaluates the interface's usability and user experience by observing and collecting feedback from representative users and experts. The users selected here are a group of participants who represent the target user population and experts identified as the observers who will strictly monitor the actions of target users and the outcomes of the system. As the participating users are visually impaired, the outcomes of each question cannot be evaluated; the need for expert observers was essential for usability testing. With this blended approach, we could gather qualitative and quantitative data, including participants' feedback on a set of questions related to user interface, task completion rates, time taken to complete tasks, and observations of usability issues.

6.3 Rating and Assessment

By analyzing the results of commands executed by users and ratings given by experts, we have established three distinct categories to assess the quality of the interface: excellent, sound, and needs improvement. Questions with an average score above nine are categorized as excellent, signifying a remarkably positive response from the participants and experts. These high marks demonstrate a strong endorsement and indicate high satisfaction with the evaluated aspects. Furthermore, questions falling between 6 and 9 are deemed suitable. This category suggests a generally positive sentiment expressed by the respondents, indicating their overall satisfaction but with some room for enhancement or fine-tuning. However, ratings below 6 falls into the need's improvement category. Such ratings imply that some specific areas or aspects require attention and refinement. These lower scores highlight potential shortcomings or areas that need further development to meet the respondents' expectations. All the ratings collected from all the experts and average ratings are shown in **Table 4**.

Table 4. Average of expert's ratings.

Ease of use	Accuracy	Efficiency	Sense of independence	User satisfaction	Response time
6.5	7.9	8.4	7.5	8.5	5.7

The response time of each voice command can be improved by the processing of voice commands locally by integrating client and server-side caching mechanisms. It can also be enhanced by optimizing algorithms for voice recognition and execution. We can also implement voice command shortcuts for the same. Since the implemented system is in the prototype stage and plugged into all the browsers, the system's performance varies even though the compatibility is 100%. Since the participating users first performed each command on the browser after training, there were some disturbances in the sequence of execution. Hence, the rating of the easy-to-use parameter was low. This will subsequently be improved when users get more acquainted with using the voice commands fluently with patience. However, participating users and experts found the feature of button navigation using audio as the most impactful for visual disability due to the screen's layout.

7. Conclusion and Future Scope

This research aims to enhance the accessibility and user experience of a telemedicine website by integrating speech recognition and synthesis technology. Our primary mission revolved around catering to the needs of visually impaired individuals and those with disabilities by providing them with a more intuitive and natural means of interacting with websites. Implementing voice commands within a telemedicine website has unlocked essential functionalities, from simple actions like scrolling and adjusting font sizes to seamless page navigation. Through these innovations, we have not only improved accessibility but also elevated the overall user-friendliness of the telemedicine platform.

The impact of our work extends beyond the realms of telemedicine. The goals of the research included the development of robust speech recognition, the design and implementation of specialized speech modules, and the construction of responsive voice synthesis capabilities. The work done lays the groundwork for future applications. Thes developed modules have the potential to transform a number of industries, such as communication, entertainment, and education, by making them more inclusive and accessible to people with a range of requirements. The knowledge gathered from this study advances voice synthesis and recognition technologies, advancing the field of human-computer interaction and strengthening our dedication to inclusion and accessibility for all.

Future work will prioritize enhancing the system based on captured feedback and assessment from usability evaluations. Improvements will target functionalities, scalability, and other efficacy parameters. While the system implementation already considers browser compatibility, a more detailed performance analysis across browsers will be conducted.

Conflict of Interest

This research article was conducted independently, without the support of external funding. The authors declare no conflict of interest.

Acknowledgments

The authors would like to express their gratitude to Manipal Academy of Higher Education, Manipal, for their invaluable assistance in enabling the publication of this research. The authors would like to acknowledge Mr. Sanjeev Kushal Pendekanti for his contribution.

References

Abdildayeva, A., Zhyilyssova, D., & Nazar, G. (2023). Voice recognition methods and modules for developing an intelligent virtual consultant integrated with WEB-ERP. In *2023 IEEE International Conference on Smart Information Systems and Technologies* (pp. 468-473). IEEE. Astana, Kazakhstan. https://doi.org/10.1109/sist58284.2023.10223552.

Alhaidari, S., & Zohdy, M. (2019). Network anomaly detection using two-dimensional hidden Markov model based Viterbi algorithm. In *2019 IEEE International Conference on Artificial Intelligence Testing* (pp. 17-18). Newark, CA, USA. https://doi.org/10.1109/aitest.2019.00-14.

Anza, P.S., & Abdullahi, M.B. (2018). A framework for multiple choice multilingual translation system using hidden Markov model and Viterbi algorithm. In *2018 Proceedings of the 1st National Communication Engineering Conference.* Zaria, Nigeria.

Barnwal, V.K., Shaw, A., Sarkar, K., Chakraborty, S., & Mukhopadhyay, A.K. (2023). Ariva: Artificial intelligence enabled voice assistance system using natural language processing. In *2023 8th International Conference on Communication and Electronics Systems* (pp. 769-776). IEEE. Coimbatore, India. https://doi.org/10.1109/icces57224.2023.10192640.

Chavali, S.T., Kandavalli, C.T., Sugash, T.M., & Subramani, R. (2022). Grammar detection for sentiment analysis through improved Viterbi algorithm. In *2022 International Conference on Advances in Computing, Communication and Applied Informatics* (pp. 1-6). IEEE. Chennai, India. https://doi.org/10.1109/accai53970.2022.9752551.

Chen, X., Wu, Y., Wang, Z., Liu, S., & Li, J. (2021). Developing real-time streaming transformer transducer for speech recognition on large-scale dataset. Retrieved from https://arxiv.org/abs/2010.11395.

Cui, X., Lu, S., & Kingsbury, B. (2021). Federated acoustic modeling for automatic speech recognition. In *2021 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 6748-6752). IEEE. Toronto, ON, Canada. https://doi.org/10.1109/icassp39728.2021.9414305.

Iyer, V., Shah, K., Sheth, S., & Devadkar, K. (2020). Virtual assistant for the visually impaired. In *2020 5th International Conference on Communication and Electronics Systems* (pp. 1057-1062). IEEE. Coimbatore, India. https://doi.org/10.1109/icces48766.2020.9137874.

Kamra, V., Singh, A., Sharma, P., & Yadav, R. (2023). A novel online quizzing system for blind people by implementing modern voice recognition techniques. In *2023 2nd Edition of IEEE Delhi Section Flagship Conference* (pp. 1-4). IEEE. Rajpura, India. https://doi.org/10.1109/delcon57910.2023.10127578.

Karita, S., Soplin, N.E.Y., Watanabe, S., Delcroix, M., Ogawa, A., & Nakatani, T. (2019). Improving transformer-based end-to-end speech recognition with connectionist temporal classification and language model integration. *Proceedings of the Annual Conference of the International Speech Communication Association* (pp. 1408-1412). http://dx.doi.org/10.21437/Interspeech.2019-1938.

Kharsa, R., Elnagar, A., & Yagi, S. (2024). BERT-based arabic diacritization: A state-of-the-art approach for improving text accuracy and pronunciation. *Expert Systems with Applications, 248*, 123416. https://doi.org/10.1016/j.eswa.2024.123416.

Levine, S., Hsieh, H., Southerton, E., & Silverman, R. (2023). How high school students used speech-to-text as a composition tool. *Computers and Composition, 68*, 102775. https://doi.org/10.1016/j.compcom.2023.102775.

Li, J., Zhao, R., Hu, H., & Gong, Y. (2019). Improving RNN transducer modeling for end-to-end speech recognition. In *2019 Automatic Speech Recognition and Understanding Workshop* (pp. 114-121). IEEE. Singapore. https://doi.org/10.1109/asru46091.2019.9003906.

Liu, S., Ma, W., Schalow, D., & Spruill, K. (2004). Improving web access for visually impaired users. *IT Professional, 6*(4), 28-33. https://doi.org/10.1109/mitp.2004.36.

Maheswari, K.G., Meenakshi, R., NaliniPriya, G., Anandasayanam, K., Hariram, B., & Pandian, G.M. (2022). Dynamic AI-based email voice assistant for web services. In *2022 International Conference on Smart Technologies and Systems for Next Generation Computing* (pp. 1-4). IEEE. Villupuram, India. https://doi.org/10.1109/icstsn53084.2022.9761287.

Padmanabhan, J., & Premkumar, M.J.J. (2015). Machine learning in automatic speech recognition: A survey. *IETE Technical Review, 32*(4), 240-251. https://doi.org/10.1080/02564602.2015.1010611.

Palaz, D., Magimai-Doss, M., & Collobert, R. (2019). End-to-end acoustic modeling using convolutional neural networks for HMM-based automatic speech recognition. *Speech Communication, 108*, 15-32. https://doi.org/10.1016/j.specom.2019.01.004.

Ra, M.K., Jonathan, L.Y., Kesuma, D.C., Edbert, I.S., Hasana, S., & Suhartono, D. (2023). Smart glasses live transcription and audio classification model to assist hearing impairment. *ICIC Express Letters, 17*(11), 1287-1294. https://doi.org/10.24507/icicel.17.11.1287.

Rao, K., Sak, H., & Prabhavalkar, R. (2017). Exploring architectures, data and units for streaming end-to-end speech recognition with RNN-transducer. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop* (pp. 193-199). IEEE. Okinawa, Japan. https://doi.org/10.1109/asru.2017.8268935.

Rattan, H., Agarwal, S., & Poovammal, E. (2021). Acceleration of web interface generation through voice commands. *Journal of Physics: Conference Series, 1911*(1), 012009. https://doi.org/10.1088/1742-6596/1911/1/012009.

Sadi, M.T.A.H., Kadir, M.I., Rahman, M.S., & Khan, M.M. (2022). Development of a voice-controlled web-based e-commerce. In *2022 6th International Conference on Computing Methodologies and Communication* (pp. 1-8). IEEE. Erode, India. https://doi.org/10.1109/iccmc53470.2022.9753691.

Sharma, A., Ahmed, V., Sharma, S., Jana, B., & Rani, K. (2022). An effective approach to speech-based email assistance for visually impaired people. In *2022 8th International Conference on Signal Processing and Communication* (pp. 32-35). IEEE. Noida, India. https://doi.org/10.1109/ICSC56524.2022.10009538.

Shewalkar, A., Nyavanandi, D., & Ludwig, S.A. (2019). Performance evaluation of deep neural networks applied to speech recognition: RNN, LSTM and GRU. *Journal of Artificial Intelligence and Soft Computing Research, 9*(4), 235-245. https://doi.org/10.2478/jaiscr-2019-0006.

Subedar, M., Krishnan, R., Meyer, P.L., Tickoo, O., & Huang, J. (2019). Uncertainty-aware audiovisual activity recognition using deep Bayesian variational inference. In *IEEE/CVF International Conference on Computer Vision* (pp. 6300-6309). IEEE. Seoul, Korea (South). https://doi.org/10.1109/iccv.2019.00640.

Swetha, P., & Swami, D.T.G. (2021). AI based assistance for visually impaired people using TTS [text to speech). *International Journal of Innovative Research in Science and Technology, 01*(01), 8-14.

Tang, R., & Lin, J. (2018). Deep residual learning for small-footprint keyword spotting. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 5484-5488). IEEE. Calgary, AB, Canada. https://doi.org/10.1109/icassp.2018.8462688.

Torad, M.A., Bouallegue, B., & Ahmed, A.M. (2022). A voice-controlled smart home automation system using artificial intelligent and internet of things. *TELKOMNIKA (Telecommunication Computing Electronics and Control), 20*(4), 808-816. https://doi.org/10.12928/telkomnika.v20i4.23763.

Wang, W., Ren, S., Qian, Y., Liu, S., Shi, Y., Qian, Y., & Zeng, M. (2022). Optimizing alignment of speech and language latent spaces for end-to-end speech recognition and understanding. In *2022 IEEE International Conference on Acoustics, Speech and Signal Processing* (pp. 7802-7806). IEEE. Singapore. https://doi.org/10.1109/icassp43922.2022.9747760.

Wei, X. (2024). Text-to-speech technology and math performance: A comparative study of students with disabilities, English language learners, and their general education peers. *Educational Researcher*. https://doi.org/10.3102/0013189x241232995. (In press).

Xue, Z., Shi, T., Zhang, X., & Xiong, D. (2024). Speaker voice normalization for end-to-end speech translation. *Expert Systems with Applications, 248*, 123317. https://doi.org/10.1016/j.eswa.2024.123317.

Yuan, J., Cai, X., Zheng, R., Huang, L., & Church, K. (2021). The role of phonetic units in speech emotion recognition. *Computation and Language*. https://doi.org/10.48550/arXiv.2108.01132.