Proceedings of the Fifth International Conference on Communication and Electronics Systems (ICCES 2020)
IEEE Conference Record # 48766; IEEE Xplore ISBN: 978-1-7281-5371-1

Virtual assistant for the visually impaired

Vinayak Iyer
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
vinayak.iyer@spit.ac.in

Kshitij Shah
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
kshitij.shah@spit.ac.in

Sahil Sheth
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
sahil.sheth@spit.ac.in

Kailas Devadkar
Department of Information Technology
Sardar Patel Institute of Technology
Mumbai, India
kailas_devadkar@spit.ac.in

***Abstract*—Research shows that people with visual impairments are 31% less likely to access the internet than individuals without disabilities. This paper illustrates the implementation of software that provides assistance to the visually impaired for accessing the internet. The software shall prove instrumental in the way the internet has accessed and will increase the ease of use drastically. Although technology has grown leaps and bounds, the internet - especially websites are still inaccessible by the visually impaired. The software provides a way to interact with these websites with much ease. With the use of voice commands instead of the traditional keyboard and mouse, our software provides a new dimension to access and provide commands to any website. The software will read out the content of the website and then using speech to text and text to speech modules along with selenium, our software can automate any website. The user is free from remembering complex braille keyboard commands or the hassle of typing, he/she can simply voice out his/her command and the software will execute it. The system also has the functionality of providing a summary of the content on the website and answering questions asked by the user with reference to the summary using a BERT model trained on the Stanford Question Answer Dataset. This software will revolutionize the internet and pave the way for Web3.0.**

Keywords— Visually impaired; Voice control; automate website; blind people

I. INTRODUCTION

Today there are nearly 285 million people in the world that are visually impaired [12]. Although technology has grown leaps and bounds, the accessibility, especially that of the internet for differently-abled people is still far-fetched. In this modern world, more and more things can be performed online. From shopping, ordering food, to booking train tickets everything can be done online. For almost all of these online facilities a person has to use a website. Using a website can be a trivial task for most people but it is very difficult for visually impaired people. The internet is a highly visual form of communication, different "accessibility blockers" can hinder different types of websites, unlike brick and mortar businesses where accessibility can be made by including a ramp for wheelchairs or braille interfaces. For example, researchers found that 80% of news sites “had significant accessibility issues," while 70% of respondents said they were “unable to access information and services through government websites.” Thus, wanted to come up with a unique way of allowing visually impaired people to access the internet. Although the W3C has a set of recommendations that stipulate the rules to be followed when designing a website for the visually impaired, not all websites necessarily stick to the high standards in terms of accessibility.

The major challenge in developing a stable software is to include as few keystrokes as possible and to provide an end to end experience with the help of voice alone. The inclusion of multiple languages and setting the right pace of the speech when played back to the user are important factors to consider. To support the widespread usage of the software, a crucial parameter is the dependency of the software on the local environment and operating systems. While the tech has evolved greatly, the accessibility, especially the internet for the differently abled is still stagnant.

Assistive technologies such as a screen reader or magnifiers can enable visually impaired individuals to access the internet. Unfortunately, these screen readers need to keep the functionality of the website in mind otherwise it becomes difficult to read data from the website. Some of the screen readers work only with a particular kind of browser and some require the user to remember complex commands thus screen readers are not an effective solution to the problem at hand and cannot be used to access the internet.

There are the following two common themes visible in most websites:

1. Web pages are partially accessible. Some parts are usable for the visually impaired, while others are not.
2. The accessibility of some web pages regressed due to updates on the web site.

Authorized licensed use limited to: Western Sydney University. Downloaded on July 26,2020 at 05:06:57 UTC from IEEE Xplore. Restrictions apply.

Both of the themes lead to an inconsistent state with regards to its accessibility. The American Foundation for the Blind [10] determined that people with visual impairments are more than 31% less likely to report to connect to the Internet and more than 35% less likely to use a desktop computer than people without disabilities.

Keeping all the above factors in mind came up with the solution of virtual assistant. The primary objective is to bridge the accessibility gap between the average user and the visually impaired individuals with regards to the internet. The internet is blind to the visually impaired, but to not make the converse the truth, in this paper present an end-to-end voice-based software for the visually impaired to enable them to access the internet with minimal to no keystrokes required. The user will provide the commands he wants to execute as a voice input instead of using a keyboard. The software then uses a speech to text module to convert the input speech to text which will be the command to be executed. The command is executed using selenium web driver. Once executed the user will have three options: - either to read the entire content of the website, read a summary or ask a question. The second and third options are implemented using machine learning. Once the voice input is taken and the command is executed the output is said to the user using the text to speech module. Thus, the software manages to make the internet more accessible easily, quickly and more effectively for the visually impaired. Figure 1 gives a gist of the overall solution and how the software works. Input speech is recognized using speech to text, the commands are then recognized and executed using a selenium web driver used to automate systems, the resulting output is played back to the user using text to speech.

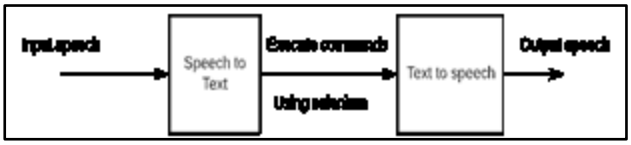

Figure 1: Flow Diagram of the proposed solution

This paper describes the implementation of the software modules automating the three most frequently used websites that are Google, Wikipedia and Gmail by users, so as to fulfil the needs of the visually impaired as much as possible.

II. LITERATURE SURVEY

Pilling et al. [1] conducted a study to determine whether the internet provides opportunities for disabled people to carry out activities which they were previously unable to do or whether it leads to greater social exclusion. Sinks and kings et al. [6] states that there is no known research to determine the reasons people with disabilities can't access the internet. Muller et al [7] on the other hand states that the primary barrier in the accessibility is that of economic and technical capabilities. This thought is seconded by Kirsty et al. [8] who states that bad html code and use pdf causes an hindrance in accessing the internet for the visually impaired Although the W3C mentions a list of guidelines for maintaining a high level of accessibility for the visually impaired, Power et al. [5] states that only 50.4% of the problems encountered by users were covered by Success Criteria in the Web Content Accessibility Guidelines 2.0 (WCAG 2.0) and 16.7% of websites implemented techniques recommended in WCAG 2.0 but the techniques did not solve the problems.

Porter [2] points out that a lot of editing has already been done if newspapers are produced on Braille for visually impaired individuals, but when they are available on the Internet the individual has the choice of what to read, which increases the accessibility. A respondent from the study conducted by [1] stated that "without the software there is no access for blind people. JAWS [3] is a specialized software which needs a knowledgeable person to provide support." Thus, with regard to these studies, it can be inferred that there is a need for a software to access the internet that is much easier for the user to operate than existing solutions like screen readers.

For developing a software to enhance web accessibility for the visually impaired, Ferati et al. [4] mentions that a "one solution for all model" is inadequate without considering the levels of visual impairment when providing customized web experience. A modular, plug-in based solution seems the most ideal for providing accessibility support for the visually impaired, that does not heavily rely on keyboard inputs (either QWERTY or Braille).

III. SYSTEM OVERVIEW AND DESIGN

The system comprises a modular client server distributed architecture. The system consists of the main menu which first runs on the startup of the software and the website modules. The client communicates with the server and back with the use of REST APIs, thus the website modules are not local to the client. Throughout the system, the user communicates with the software via speech-to-text interface. The Google library of speech-to-text (Speech Recognition) for Python is used for this purpose. For communicating the system's output to the user as well as for confirming the user input, the recognized input is played back to the user using the Python text-to-speech library (pytssx3). The modules are written in Python and make use of Selenium for automation of the respective module and Beautiful Soup for scraping the contents of the web page. The "Script" component of each module consists of the customized code that entails the features of the website contained in the module. For instance, the Wikipedia module consists of a Question and Answer and Summary feature along with the traditional feature of reading out the entire article. The former is implemented by training a BERT model on the Stanford Question Answering Dataset (SQuaD). The APIs that hold the system together are written in Flask. The software is operating system independent to support hassle free application and usage of the system.

Authorized licensed use limited to: Western Sydney University. Downloaded on July 26,2020 at 05:06:57 UTC from IEEE Xplore. Restrictions apply.

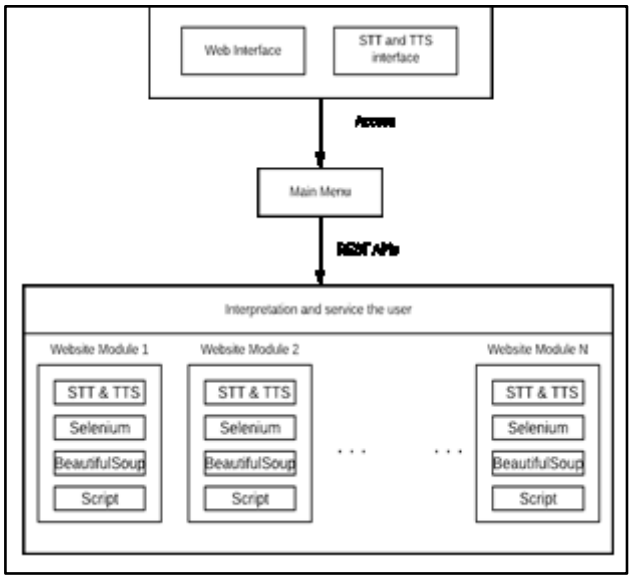

Figure 2: System Architecture

Figure 2 is a representation of the system architecture of our software. The user accesses the software using the web interface where the speech to text (STT) module converts the voice input to text. The user is then presented with the main menu where they have three options to choose from and decide which website they want to browse. Accordingly, the module is invoked with its corresponding speech to text modules, web driver and machine learning module. The output is played to the user using text to speech (TTS) module. This is the overview of the software.

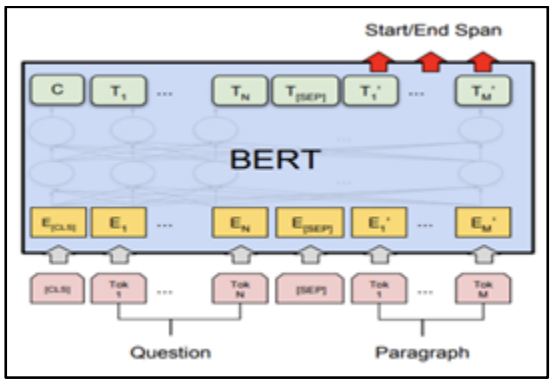

Figure 3: BERT model on SQuAD Dataset architecture [11]

Dataset - The Standard Question Answering Dataset (SQUAD) available to pre train the machine learning model for the question and answers component of the module. The dataset has questions posed by people on Wikipedia where the answer to the question is from within the given excerpt of text on Wikipedia or it may be unanswered [9].

IV. METHODOLOGY

The user first interacts with the main menu of the software once the desktop computer or laptop has been switched on. The main menu of the software can be invoked by either the integrated voice assistant, for example Siri, or by a predefined keyboard shortcut, being the only keyboard interaction required. The main menu interface provides the available options to the user viz. Installed website modules, pace of the audio, accent of the audio. Each of the website modules contains a speech-to-text and text-to-speech bundle, a python script that automates the website and the features specific to the website. For efficient speech recognition, the user is provided with a beep at all stages after which he is free to speak. The input received and recognized by the system from the user is also played back to the user so that the user can confirm his intended input, to reduce any errors right at that particular stage, thus, enabling a sense of editing. The methodology followed to implement three modules - Google, Gmail, Wikipedia - and the main menu is described below.

A. *Main Menu*

The main menu runs when the software is first opened. Using the pytts (Python text-to-speech) module, the initial set of instructions illustrating the options provided to the user. The system takes the user input after the beep using Google speech-to-text python module. The keywords from the voice are then extracted and appropriate response is executed. The user is also free to change the voice tempo and accent that suits him/her the best.

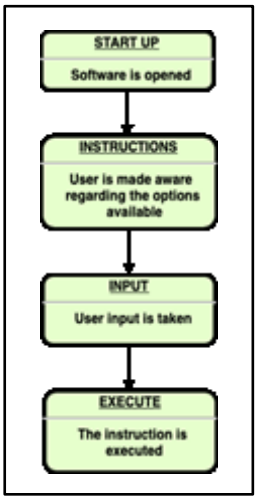

Figure 4: Flow diagram for main menu

Authorized licensed use limited to: Western Sydney University. Downloaded on July 26,2020 at 05:06:57 UTC from IEEE Xplore. Restrictions apply.

B. *Google Module*

This module consists of a python script that automates the website using Selenium and BeautifulSoup. The user can search for any query through the speech-to-text and text-to-speech interfaces and the recognized query is searched with the help of

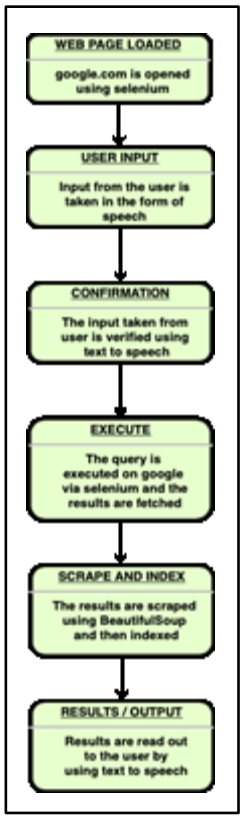

Figure 5: Flow diagram for Google

Selenium. The resulting search results are read back to the user by scraping the web page contents using the Beautiful Soup module of python. The search results are indexed which enables quick accessing of the web page according to the user's desire, thus saving time, as opposed to the user reading out the whole search result that he wishes to select.

C. *Gmail module*

This module consists of a python script that starts up Gmail, logs the user into his/her mailbox and provides the support for the user to send or read mails. For sending a new mail, the system prompts the user to provide relevant details and after filtering out noise, through selenium the input fields are filled respectively, and then sent with the user's confirmation. At each stage, the user is free to edit and undo any of his inputs. The system repeats the recognized user input and the input is finalized only if it's confirmed by the user.

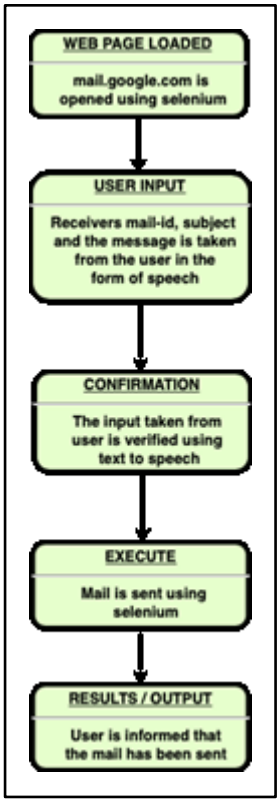

Figure 6: Flow diagram for Gmail

D. *Wikipedia module*

The Wikipedia module presents the user novel options such as summarizing and reading out the article, and provides intelligent answers to queries using NLP and Machine Reading Comprehension. Once the web page is loaded, the user enters the search query, followed by the confirmation, after which the user is provided with 3 options- reading out the entire article, reading out the summary of the article, a question and answer session. The entire article is read by scraping the web page, cleaning the text, and using the text-to-speech module. Summarization of the text is performed using the summary method provided by the Wikipedia python library. For the question answer session, a BERT model on Stanford Question Answering Dataset (SQuaD) is used. It consists of 100,000 questions with over 50,000 unanswerable questions. BERT is used for Question Answering on SQuAD dataset by:

- applying two linear transformations to BERT outputs for each sub token.

Authorized licensed use limited to: Western Sydney University. Downloaded on July 26,2020 at 05:06:57 UTC from IEEE Xplore. Restrictions apply.

- First/second linear transformation for prediction of probability that current sub token is start/end position of an answer

The user can then ask any question relevant to the topic of the article searched for, and the model returns the most suitable answer to the user through text-to-speech.

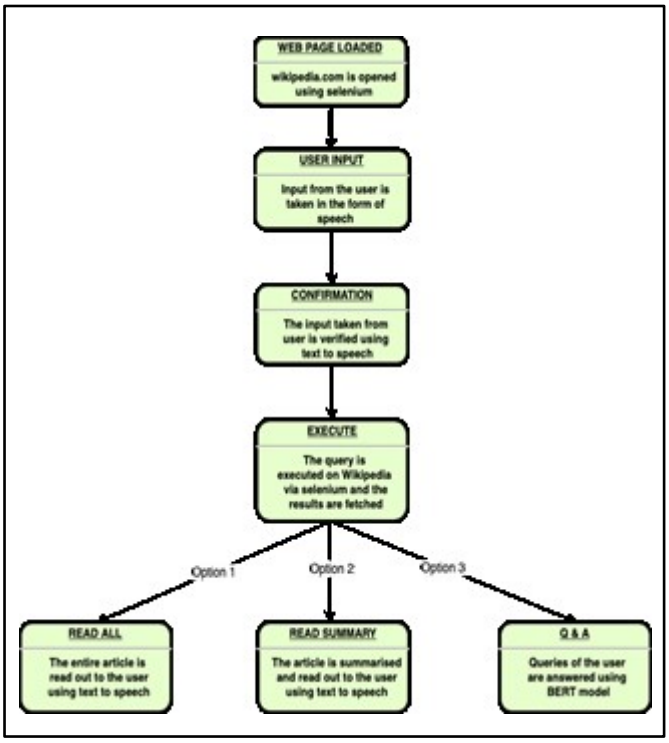

Figure 7: Flow diagram for Wikipedia

V. RESULTS

The built-in modules of text to speech (pyttsx3) and speech to text (speech recognition library by Google) in python provide a good accuracy and also provide an easy and quick way to convert the text. The speech-to-text recognized the words with 96.25% accuracy with 4 different voice samples each containing 20 different inputs in a moderate to quiet environment.

The BERT model on SQuAD dataset for the question answering feature in the Wikipedia module showed an Exact Match accuracy of 80.88% which is the percentage of predictions that match any one of the ground truths answers exactly, and the F1 score was found to be 88.49%.

Results showed that were able to run our software on the three most popular sites: Google, Gmail and Wikipedia. The software was run on each of them separately. The software could send an email effectively using the commands from the user. The software also provided an accurate answer to the question the user asked on Wikipedia. The software managed to summarize the text in Wikipedia accurately and thus were able to test and build a software that will make the website easily, quickly and efficiently accessible for the visually impaired.

VI. APPLICATION

Virtual Assistant for the visually impaired acts as a great support to the visually disabled people to access the internet on any browser as our software is browser independent. They can access the internet using their speech and then can navigate the website using voice commands. The software will read out the content of the website to the user thus making the website more accessible. This feature will not only help the visually impaired but also allow other people to access the internet with ease and eliminate the use of hardware devices like the keyboard.

Virtual Assistant also provides the feature of providing answers to a particular question from a given text of data, thus now the user does not have to read the entire text to figure out the answer, he/she has to simply input the question, the software will find out the answer from the text data on itself using machine learning. The software also provides a summary of the text using machine learning, so the user doesn't have to read the entire thing and thus making it easy to access the website. Thus, using machine learning and speech to text techniques make the task of accessing the website, which was earlier difficult now super easy, quick and efficient. Thus, believe that virtual assistants for the visually impaired are the beginning of Web 3.0.

VII. CONCLUSION

In this paper, a modular solution is presented to improve web-based accessibility for the visually impaired. The virtual assistant is operating system independent and does not rely on keyboard inputs from the user to maximize ease of use and aims to provide a hassle-free experience for the user. Through speech to text and text to speech interfaces, the user can communicate with and customize the system. The system design is presented and methodology of the three modules that is currently implemented. The Wikipedia module uses a BERT model on the SQuAD dataset to answer user queries quickly and accurately. The Exact Match was found to be 80.88%.

The virtual assistant provides an easy way to access any website for the visually impaired. It eliminates the need to remember complex keyboard commands or the use of screen readers. The assistant is not only a great way to interact with the websites but also an effective way to do so. The software works as a steppingstone towards Web 3.0 where everything will work on voice commands.

Authorized licensed use limited to: Western Sydney University. Downloaded on July 26,2020 at 05:06:57 UTC from IEEE Xplore. Restrictions apply.

VIII. FUTURE ENHANCEMENT

At present the application supports only commands given in the English language and plan to expand that and make it available in most of the daily used languages thus people from all parts of the world can access the web without any issue

Also like to create a uniform framework that can be plugged to any website and create a browser extension thus making it possible to toggle between the two modes easily, especially for educational websites to enable visually impaired individuals to access online courses just like the average individual.

REFERENCES

[1] Pilling, D., Barrett, P. and Floyd, M. (2004). Disabled people and the Internet: experiences, barriers and opportunities. York, UK: Joseph Rowntree Foundation, unpublished.

[2] Porter, P. (1997) 'The reading washing machine', Vine, Vol. 106, pp. 34–7

[3] JAWS - https://www.freedomscientific.com/products/software/jaws/ accessed in April 2020

[4] Ferati, Mexhid & Vogel, Bahtijar & Kurti, Arianit & Raufi, Bujar & Astals, David. (2016). Web accessibility for visually impaired people: requirements and design issues. 9312. 79-96. 10.1007/978-3-319-45916-5_6.

[5] Power, C., Freire, A.P., Petrie, H., Swallow, D.: Guidelines are only half of the story:accessibility problems encountered by blind users on the web. In: CHI 2012, Austin, Texas USA, 5–10 May 2012, pp. 1–10 (2012)

[6] Sinks, S., & King, J. (1998). Adults with disabilities: Perceived barriers that prevent Internet access. Paper presented at the CSUN 1998 Conference, Los Angeles, March. Retrieved January 24, 2000 from the World Wide Web

[7] Muller, M. J., Wharton, C., McIver, W. J. (Jr.), & Laux, L. (1997). Toward an HCI research and practice agenda based on human needs and social responsibility. Conference on Human Actors in Computing Systems. Atlanta, Georgia, 22–27 March.

[8] Kirsty Williamson, Steve Wright, Don Schauder, Amanda Bow, The internet for the blind and visually impaired, Journal of Computer-Mediated Communication, Volume 7, Issue 1, 1 October 2001, JCMC712

[9] Deeppavlov documentation http://docs.deeppavlov.ai/en/master/features/models/squad.html accessed in April 2020

[10] The website for American foundation for the blind https://www.afb.org/about-afb/what-we-do/afb-consulting/afb-accessibility-resources/challenges-web-accessibility accessed in April 2020

[11] Ryle Zhou, Question answering models for SQuAD 2.0, Stanford University, unpublished.

[12] Global data on visual impairments 2010 by World Health Organisation (WHO)https://www.who.int/blindness/GLOBALDATAFINALforweb.pdf?ua=1

Authorized licensed use limited to: Western Sydney University. Downloaded on July 26,2020 at 05:06:57 UTC from IEEE Xplore. Restrictions apply.