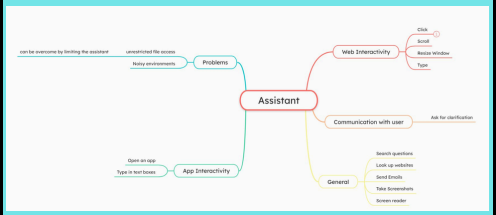
Utilzing Automatic Speech Recognition to Create a Desktop Assistant

Adel Benchemam
Advisor: Dr. Kevin Crowthers, Ph. D.

Background

We already rely on virtual assistants to turn on and off the lights in a room, stream music, or as search engines (Subhash et al., 2020).
Previous assistants such as Microsoft's Cortana, Google Assistant, and Apple's Siri. While these assistants are very powerful and can act as a search engine and retrieve information such as weather (Jain et al., 2021). However, they lack
This technology relies on a process named Automatic Speech Recognition, which transcribes what the user says.
Automatic Speech Recognition systems initially record speech and save the speech as a file, the file will then be cleaned of any background noise and analyze what is stated sequentially. Probability tests are applied to recognize all the words that complete the input. Finally, it will produce an output in the form of text content (Subhash et al., 2020).



How ASR Works

Output: "I read a book"

Figure 1: The architecture behind an assistant's ASR model inspired by (Appalaraju et al., 2021)

Engineering Need

Human-computer Interface are hardware-based and usually very hard to access for those who are disabled. These disabilities can be a physical challenge when using technology. Voice assistants are the solution to this as they can perform tasks such as setting timers, sending text messages, and look up questions. However, these assistants can't perform specific tasks such as clicking on buttons in websites.

Engineering Objective

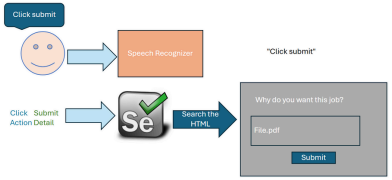
The goal of this project is to program a cross-platform assistant that can interact with website elements and interact with files safely. The audience for this can be general, however, it's aim is to help those with disabilities interact and navigate around their computer.

Analysis

No ASR model is 100% perfect. However, this data shows that the model would recognize speech in environments with some background noise. This also allows us to run a 1-proportion Z-interval. The confidence intervals tell us that at low background noise levels, the model recognizes speech well.

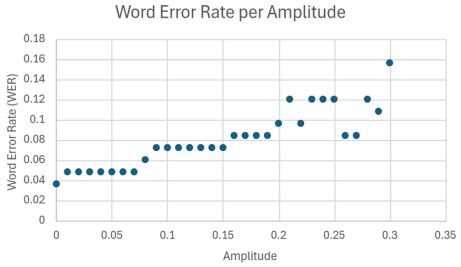
The Word Error Rate graph also shows that under low levels of background noise there is a low error rate.

Design



Methodology

To test the accuracy of the model, I decided to use a website named NaturalReaders, and Audacity. The website provides voices that I used to test if the model would recognize them. The use of Audacity ensured that there was background noise to disrupt the voice. There were 33 voices that were used, and they all said the same sentence: "How many feet are in a mile". The accuracy of each background noise was taken and used to make a projection.
Word Error Rate (WER), a common metric for speech recognition accuracy was also collected using a news article audio and white noise.

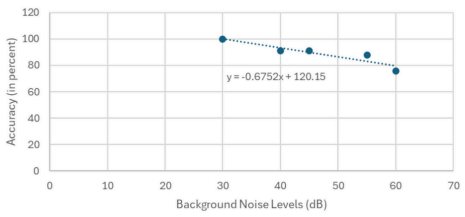


Conclusion

I created an application that builds upon past voice-controlled assistants. This application allows for hand free navigation of a computer and allows the user to select website elements that appear on the screen. I've also added a feature to track the head of the user and navigate with the cursor. Word Error Rate and proportions were both used in order to determine which noise levels was the assistant usable.

Materials

Accuracy With Different Background Noise Levels



Confidence Interval of Different Background Noise Levels

Background Noise(dB)	Confidence Interval
40	81.1% - 100%
45	81.1% - 100%
55	76.7% -99.0%
60	61.1% - 90.4%

Future Steps

In the future, I would like to:

- Create a model with better noise reduction techniques such as using Spectrum Matched Training (Prodeus & Kukharicheva, 2016)
- Work with motion detection
- Receive Feedback from people who may use this assistant to increase convenience
- Add an eye detection feature