IPRO 316 | Improving Audio CAPTCHAs

Objective:

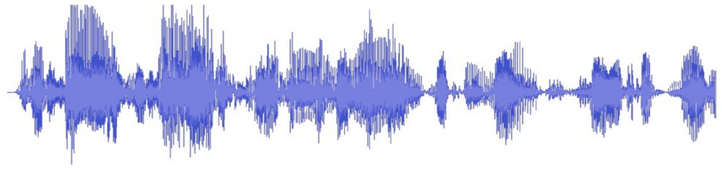
To find solutions that would allow humans to better understand audio CAPTCHAs (Completely Automated Public Turing test to tell Computers and Humans Apart) and to limit the ability for computers to interpret the same tests.

Background:

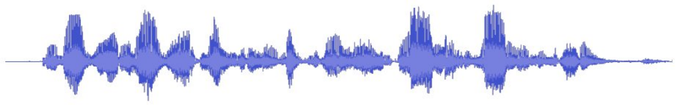
- CAPTCHAs are used by various sites like Google and Digg to prevent "bots" or computer programs from registering and getting the same rights as a user.
- Most CAPTCHAs come in visual form, able to keep most programs from entering, but limiting visually disabled people from entering as well.
- Audio CAPTCHAs are currently very hard to understand by humans, but easy to break by computer programs.

Methods:

We took sentences created by our team members and had one of our team members record each of the sentences as a base. Using those base audio files, we modified them using different techniques. We tested the files against an open-source speech recognition software, Sphinx, to determine how effective the computer could understand the files. Once completed, we created an experimental website and had people from all over the world listen to the sentences to see if they could fare better than Sphinx.

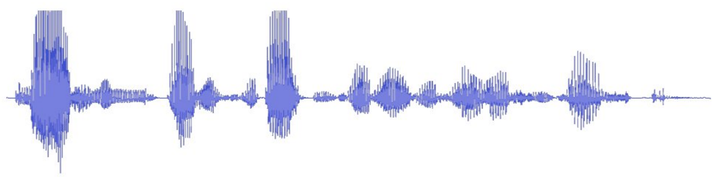


Control: The control sound file is a sound file that was taken straight from the recording software. There was nothing added nor subtracted from this sound file.



Echo: The echo sound file was created by repeating the original sound file with a slight delay added while the original was being run. This resulted in hearing the sentence on top of itself.

Dutch: The Dutch sound file was created by placing a Dutch-speaking audio file over the original audio file, resulting in hearing both English and Dutch speech at approximately the same time.

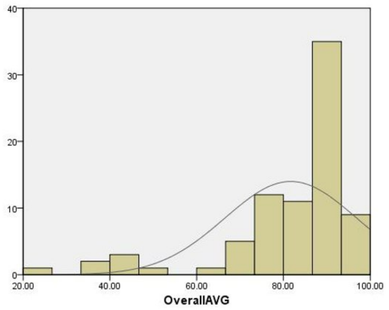


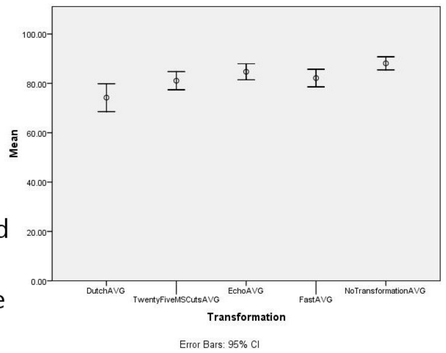
Fast: The fast sound wave was created by speeding up the rate at which the original file was played, then reducing the pitch of the resulting sound. This allowed the text to be sped up but still be in a range that would allow humans to hear the text in an unmodified form.

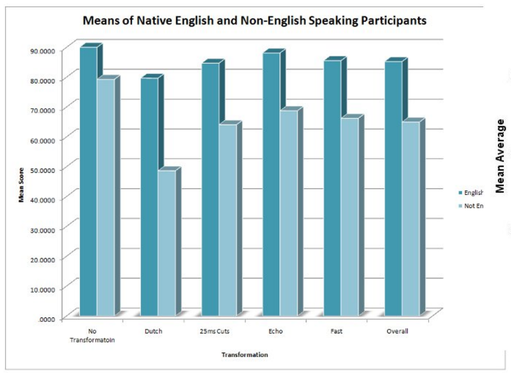
25ms: The 25ms sound file was created by alternating between 25 milliseconds of the sound file with 25 milliseconds of silence, creating a stopped sound effect for the file.

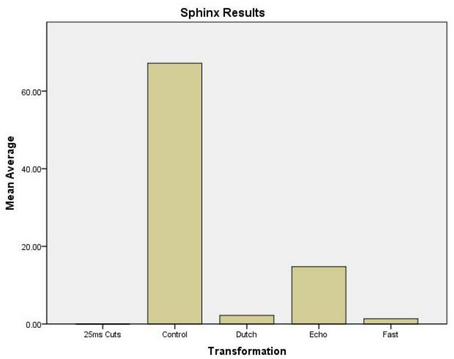
Results:

Over the course of our experimentation, we found that most of our participants handled the test well with only a few exceptions with people who could not comprehend the audio. The majority of the participants in the study scored higher than the mean of the total scores. The mean scores of all the human tests show that no matter the transformation, the ability to recognize the sentences were over 70%. However, with Sphinx's trials, we found that while it scored over 60% on the untransformed files, it was unable to understand 25ms, Dutch, and fast transformations at all. Because we were able to find participants from all over the world, we were able to also determine how well English speakers and Non-English speakers did against each other in each of the transformations. The Dutch transformation fared the wost, with non-English speakers only able to score 50% on the tests.









Conclusion:

We recommend that in further creations of audio CAPTCHAs, the developers attempt to use 25ms cuts and speeding up the audio files. Transformations with another language can be attempted, but only if being used in a primarily English speaking website.