Introduction

Technological innovation has outpaced our privacy protection. As a result, companies and organizations have more ways than ever to track our data. While we enjoy the convenience of technology, we give up more of our personal information, such as our biometric identifiers, behavior patterns, and preferences. Organizations sometimes misplace or exploit collected users' information, rendering such users vulnerable. With unbounded data collections around us extracting our information faster than ever, we lose track of where our data end up. Subsequently, our data becomes more susceptible to unintended uses by those unknown to us.

My research aims to develop different mediators allowing users to control the amount of information they provide when using technology. The starting point of my research will be analyzing voice assistants such as smart speakers. A smart speaker is a loudspeaker connected to the internet with an integrated virtual assistant controlled by spoken commands, activated by a wake word. 94.9 million smart speakers, such as Amazon Alexa Echo and Google Home Assistant, are installed in the US [1]. This massive scale of smart speaker usage makes people's preferences, voices, and daily activities accessible to other companies and open to unwanted exploitation. To combat such an issue, I propose placing a mediator such as a chip with privacy preservation interventions inside devices to protect more of customers' data. I intend to apply my research in coordination with the National Institute of Standards and Technology (NIST) to motivate systemic changes in business behavior. The mediator can help NIST determine reasonable criteria to set a benchmark for their NIST Privacy Framework that "helps organizations answer fundamental questions: How are we considering the privacy impacts to individuals as we develop our systems, products, and services [2]." Products with mechanisms implemented to preserve privacy following the discovered benchmark should receive privacy labels issued by NIST. As a result, when customers purchase products with privacy labels from NIST in the future, they know that such products have approved methods to retain more of their data.

In terms of broader impact, there has not been an accepted mediator placed between users and voice-assistant devices, and this research aims to be the first one that sets a precedent. Suppose the mediator can carry out the discussed solutions in an efficient manner. In that case, NIST can apply our mediator to establish a benchmark for a privacy label that indicates the product conducts reasonable privacy preservation. Subsequently, customers can discern products

that protect their privacy better. As a result, incentivized to obtain the privacy label from NIST, companies are more likely to implement privacy-protection mechanisms similar to the mediator. The starting point of my research is analyzing smart speakers, as I aim to create a paradigm for establishing a benchmark for privacy preservation in the space of audio. However, I believe this paradigm of analyzing how devices collect users' information, finding ways to minimize information released to the device, and maintaining functionality can extend to other spaces. For example, devices that involve visual information, such as Ring Video Doorbell, can also improve how they protect users' privacy. Nutrition facts labels tell us what invisible ingredients we take in when consuming foods. Similarly, privacy labels intend to inform users how their information will be used and provide an essential guarantee of some data preservation. If we gradually extend this privacy label to more spaces, privacy will no longer be just a bonus but a requirement and a human right.

Outline of questions/problems addressed:

1. How to allow people to control when devices collect their information
2. Hide users' information but still be able to interact with smart speakers' virtual assistants
3. Mimic functionalities such as personalization in a safer way

Step 1: Allow people to control when devices collect their information

Evidence shows that a smart device's actions do not match the user's mental model. For example, most people believe that smart speakers such as Alexa Echo only listen after people activate them with a wake word [3]. However, there has been evidence indicating that Alexa Echo starts listening before being activated or always listens to users despite the mute button [4]. The mismatch between users' expectations and how the devices collect users' information revolves around when the device begins collecting users' data. Therefore, the first step in the mediator's development is creating a mechanism that lets users know precisely when the device is listening.

Step 1: Results

The approach I will investigate is loading an ML wake-word model on the mediator that controls when the smart speakers' microphones receive signals. Users can choose a unique wake

word that is unlikely to be mistaken for a word used in common speech, thus reducing the chances of the smart speaker being unintentionally activated. Also, unless our wake word is given, the smart speakers' microphones will be shut. As a result, users will know that when they do not intend to use the smart speaker, the smart speaker is not listening.

Step 1: **Results**

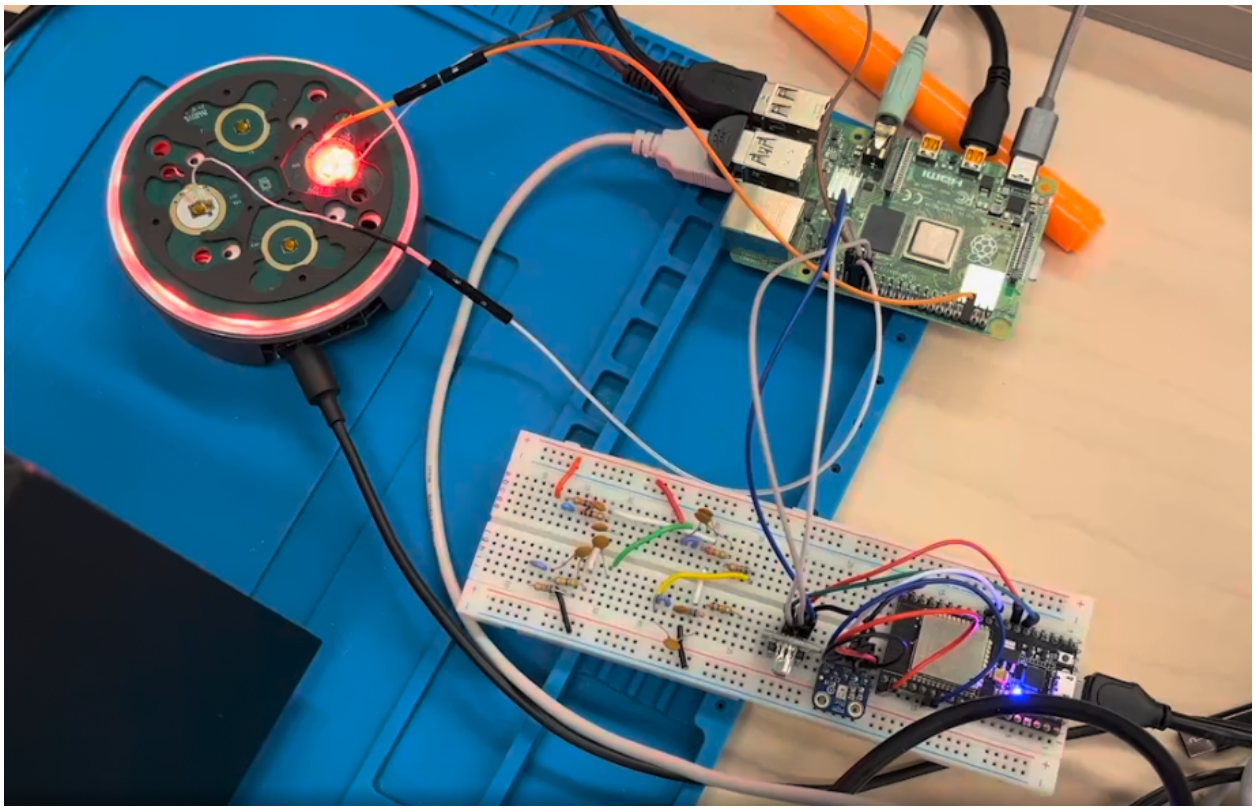

Figure 1. Illustration of implementation of an independent wake word detection implementation

An ESP can be utilized to conduct our own wake word detection. A wake word detection model is loaded onto the ESP, shown at the bottom right of the diagram. The two available wake words are "Sheila" and "Marvin" to activate the process of delivering a command to Alexa (shown on the left hand side). Otherwise, by default, all microphones on Alexa will be shut, so that Alexa will not be able to receive any audio information. Once the wake word is given, the ESP will use GPIO to send signals to the Raspberry Pi to record the users' command. If given the wake word "Marvin", no additional voice obfuscation will be conducted on the recorded audio. On the other hand, if the wake word Sheila is given, a simple voice obfuscation method, such as changing sampling rates, will be conducted. Once the users' command is recorded, the Raspberry Pi will send two signals to Alexa: one for unmuting an Alexa microphone, and the

other for activating Alexa, so that Alexa will be activated without giving the wake word "Alexa." Alexa will only be receiving audio information after the Raspberry Pi sends signals to Alexa. In the current example, the recorded/altered user's command is outputed outloud to Alexa. For future work, the recorded/altered user's command will be delivered through the microphone stream, so that they do not need to be outputed outloud. As a result, the process of using a smart speaker will be the same as before, except that users will use a different wakeword. This process allows users to control precisely when smart speakers are listening. For future work, I will try to find a microprocessor that is robust enough to conduct the wake word model, audio recording, and voice obfuscation all in one. Hence, as a proven concept, with reasonable computational power and affordable hardware, users will be able to know exactly when smart speakers are listening.

Step 2: Hide users' information but still be able to interact with smart speakers' virtual assistants

Smart speakers send users' spoken commands to the cloud without informing users explicitly. Also, additional functionalities developed by third-party platforms, such as Alexa Skills, could exploit users' command messages and voices for their purposes. In step 2, I will analyze interventions to minimize information released to smart speakers while interacting with their virtual assistants.

Step 2: Results

One approach is loading an ML speech-recognition model onto the mediator to transcribe users' spoken commands, and then output transcribed messages in a different synthetic voice to the smart speaker. This first method preserves the most privacy: it hides the users' original voice (timbre), speech intonation, and emotions. I will analyze the computational power and accuracy of varying speech-recognition models since they will decide if virtual assistants can understand the given command properly. The process is shown in the diagram below.

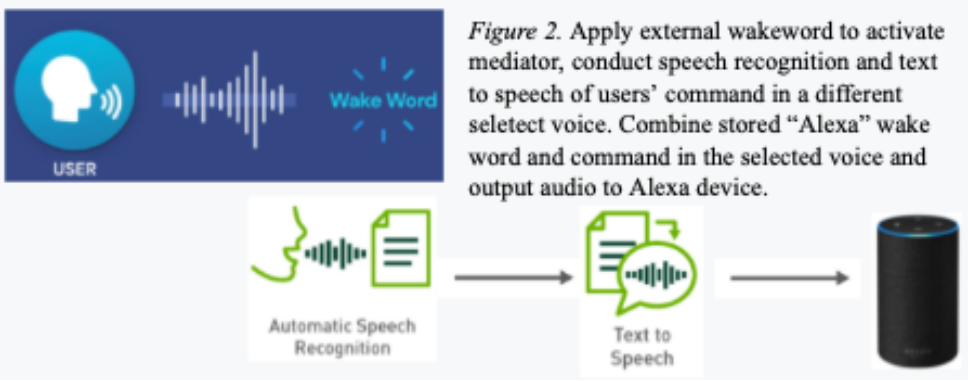

Figure 2. Apply external wakeword to activate mediator, conduct speech recognition and text to speech of users' command in a different seletect voice. Combine stored "Alexa" wake word and command in the selected voice and output audio to Alexa device.

Another way will be using a voice conversion model, such as Gaussian-Mixture Model (GMM), to output the users' commands in a different voice directly. This second method retains users' speech patterns. However, direct voice conversions can more likely allow people with different accents to interact with virtual assistants. That is because virtual assistants' speech-recognition model is more robust than a lite version of a speech-recognition model likely placed on the mediator. I will apply a speech-recognition model to evaluate the intelligibility of the commands in the converted voices. Furthermore, I will use speaker recognition models on the original and converted voice commands to check how similar they are. Hence, I can choose an optimal voice conversion model by comparing the intelligibility and how different the converted voices are from the original voices. To test the feasibility of the GMM voice conversion model, we applied two existing audio datasets, M1 and F1. M1 includes voice samples from a typical male, while F1 includes voice samples from a typical female.

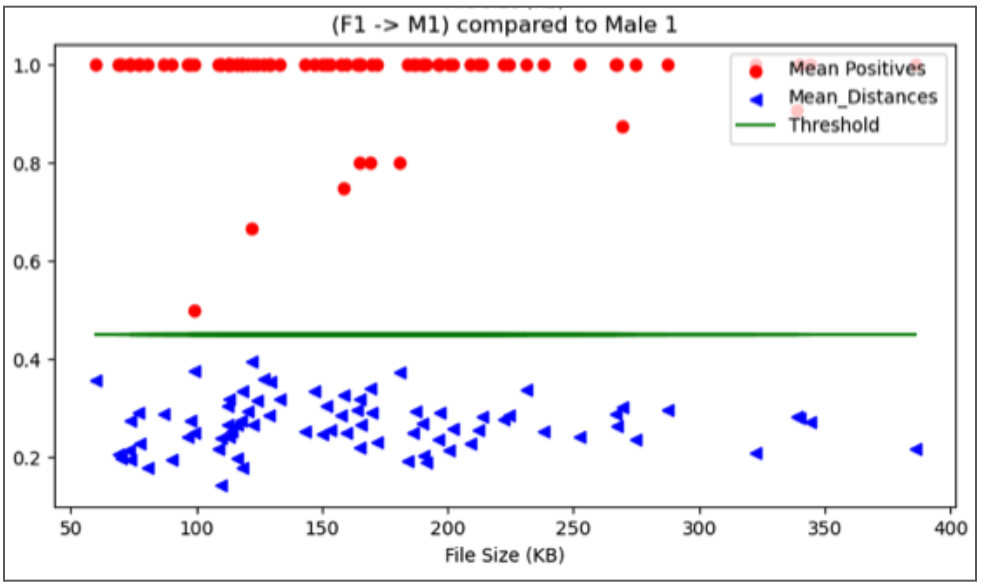

Figure 3. The differences between M1 voice converted from F1 compared to the original M1 voice.

I utilized a one-hot encoded speaker recognition model. In essence, the model uses cosine similarities of the audio samples to check if audio samples are from the same person. From the above diagram, I illustrated that the converted M1 audio samples and the original M1 audio samples have mean distances below the threshold. Hence, the machine considers that converted M1 audio samples and original M1 audio samples are from the same person. This provides evidence that GMM voice conversion can conduct personalization, where users can consider registering a converted voice. Subsequently, each time the user interacts with Alexa, their command will be converted to the registered voice and can hide their original voice.

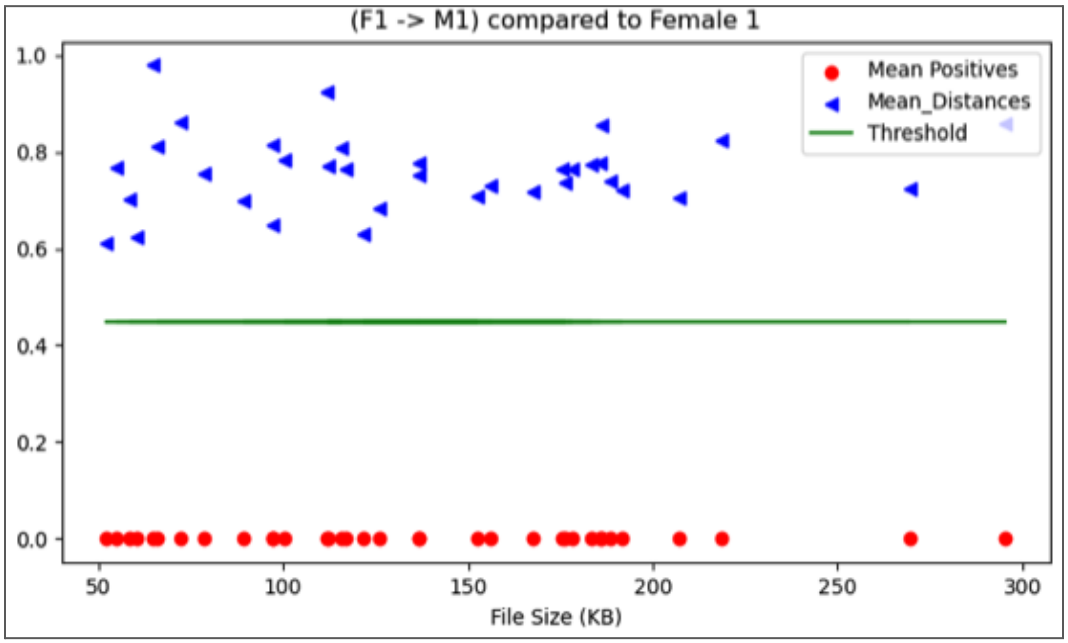

Figure 4. The differences between M1 voice converted from F1 compared to the original F1 voice.

The above diagram showcases that the converted audio samples are very different from their original audio samples. The machine is not able to tell where the converted audio samples came from. Once F1 audio samples are converted to M1's voice, the machine considers that the converted M1 voice drastically different from the original F1's voice. This illustrates that machine learning voice conversion algorithms can reasonably hide users' original voices.

The third approach is conducting randomized pitch shifts on the users' voice command, using methods such as Wavelength Similarity Overlap-Add (WSOLA). WSOLA requires much less computational power and maintains great intelligibility. From my testing, Alexa could not tell the correct speaker from the pitch-shifted voices, indicating some potential to conceal users'

identities through WSOLA. However, it might be relatively easy for companies to reverse the pitch-shifted voices back to the original voices, thus exposing users' original voices. Hence, I will use speaker recognition models on the original and converted voice commands to check how similar they are.

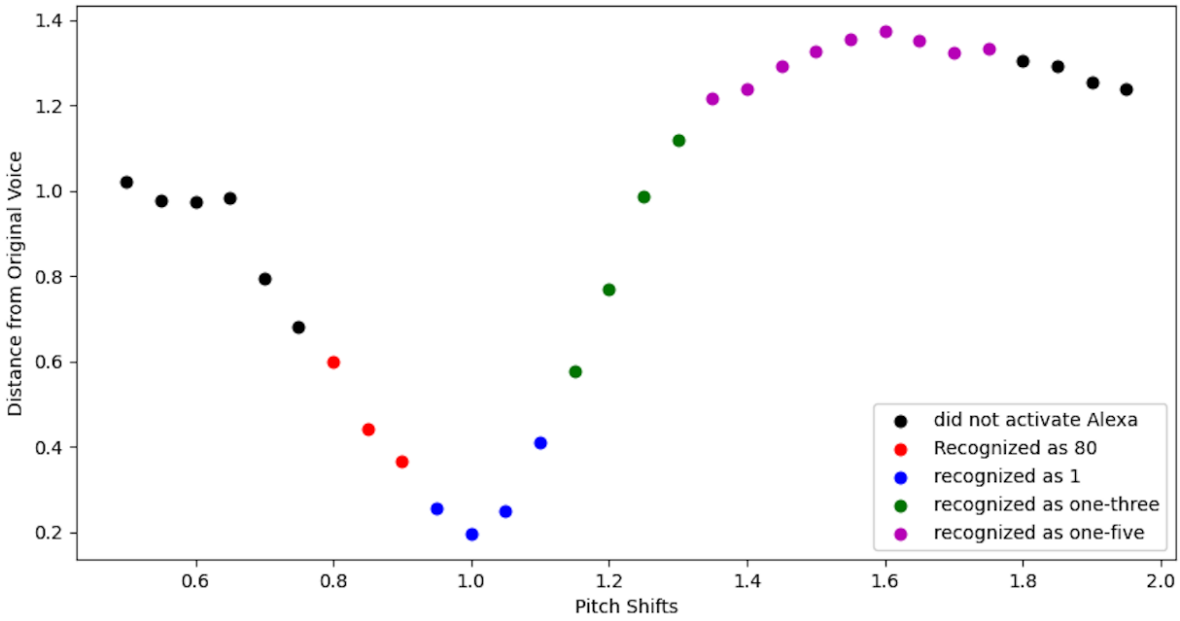

Figure 5. The differences between pitch shifted voices and original voices.

I recorded the audio samples of answers needed to register a user's voice to Alexa. Subsequently, I pitch shifted the original voice sample at different magnitudes, between 0.5 and 2.0.

The lower the magnitude, the lower the pitch, and vice versa. 1.0 represents the original voice and original pitch. Similarly, I applied YOSO, the one-hot encoded speaker recognition model to check how different the pitch shifted voices are from the original voices. Below a certain pitch shift magnitude, 0.75, Alexa will not recognize human voice due to lack of intelligibility. Gradually, as intelligibility increases, mean differences decrease. Once the pitch shift magnitude increases beyond 1.0, the mean differences increases again. However, once the pitch shift magnitude surpasses 1.6, the increase in pitch shift magnitude does not make increase mean difference. Above the pitch shift magnitude of 1.8, Alexa considers the audio sample as inaudible. This approach shows that if we can pitch shift at random magnitudes within reasonable ranges, Alexa might not be able to tell the speakers' original pitch.

Step 3: Mimic functionalities such as personalization in a safer way

The mediator should still be able to process personalized requests. For example, users can provide commands such as "play my song playlist," and Google Home Assistant will know to play the playlist from the correct speaker's account, given the speaker's voice. One way to set up

personalization and preserve privacy is to have the mediator create a one-to-one mapping between a user's voice and another voice. As a result, whenever someone talks to the smart speaker, the mediator will automatically map that person's voice to a different voice. This requires the mediator to contain an ML speaker recognition model to identify the speaker and correctly output the corresponding voice. Another approach is to have different wake words, where each wake word matches a different voice. For example, "Alexa-one" is designated for the first family member and matches a voice. Whenever "Alexa-one" is used to activate Alexa Echo, it will receive the voice for the first family member and provide personalization.

Step 3: Results

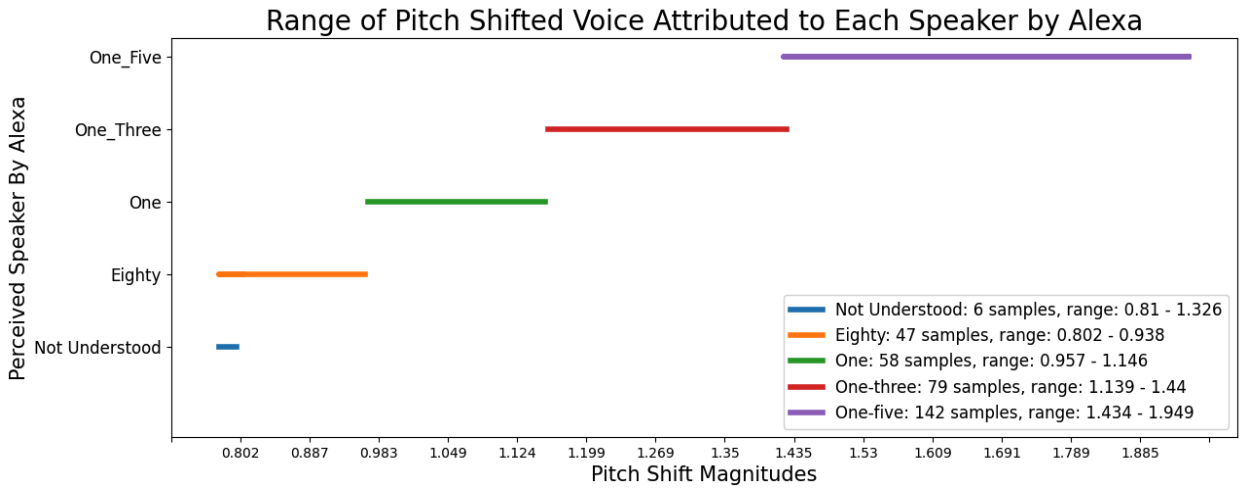

Figure 6. Range of pitch shifts Alexa associates with each registered user

In the beginning, I registered my original voice to Alexa. Subsequently, I registered my voice at different pitch shifts, 0.8, 1.3, and 1.5 to Alexa as well.

Using Alexa Developer Console and auto-clicker, I was able to automate the process of testing pitch shift's personalization capabilities. Using the Alexa Developer Console, all the Alexa responses will be recorded on the web page. Applying the auto-clicker will open the microphone for recording by clicking the microphone button on the Alexa Developer Console web page. There were 332 randomly pitch shifted audio samples: randomly pitch shifted the audio between magnitude of 0.8 and 1.75. The original audio sample includes the text, "Alexa, who am I," which triggers Alexa to respond with the registered users that Alexa thinks said the message. Syncing the Alexa Developer Console recording time and response with the auto-clicker allows for generation of responses on the web page that can be easily analyzed. Subsequently, the analysis indicates that there is very overlap between the pitch shift range associated

with each pitch-shifted registered user. Therefore, registering pitch shift voices can be a reasonable method of concealing users' original voices and maintaining personalized functionalities. As a reminder, giving the wake word "Sheila" conducts voice obfuscation. A user applying the registered pitch shifted voices can still conduct personalized requests, as each time they talk to the smart speaker, the mediator will conduct pitch shift automatically to a designated registered voice, if given the wakeword "Sheila."

Conclusion

The goal is to have a mediator in the form of a chip or embedded device placed within smart speakers that preserve privacy in some form, as discussed. To make the mediator compact, the wake word, speech recognition, speaker recognition, and voice conversion models will have to be efficient for them to work together. Therefore, analyzing the different ML models will be extensive, and field-programmable gate array (FPGA) will be used to speed up the ML models. The results discussed above show promising solutions to allow users control when smart speakers collect their information, how to conceal their original voice, and still maintaining existing functionalities that depend on them registering their voices. Subsequently, we will try to discover more powerful microprocessors that can do all three things together. In addition, I will conduct user testing for each discussed method. The final mediator will find the balance between privacy preservation, computational power, and usability. As a result, when manufacturers create such a mediator on a large scale, because of economies of scale, the cost of developing such a mediator will be very low. Hence, we will be able to make a convincing argument that it is reasonable for companies to create and implement such mediators in their devices to better protect users' information. As a result, such devices will receive privacy labels, and consumers can better discern products that protect their privacy.

Citations

[1] US: Smart speaker installed base 2018-2022, *Statista. 2022.*

[2] NIST PRIVACY FRAMEWORK, *Nist.gov, 2020.*

[3] J. Lau, B. Zimmerman, F. Schaub, *ACM on Human-Computer Interaction, 2018.*

[4] U. Iqbal. et al. *Arxiv, 2022.*