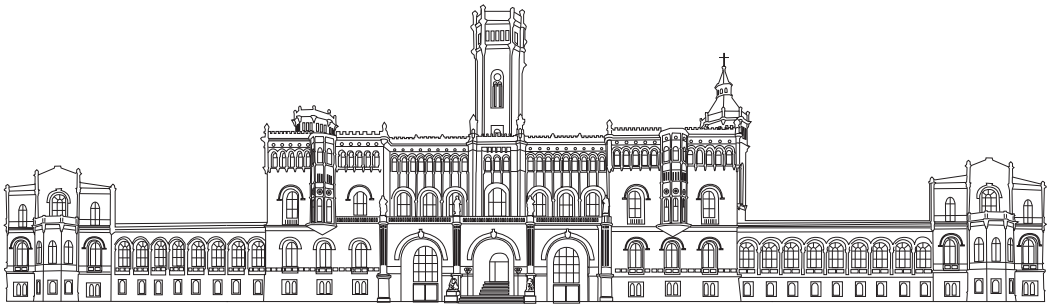


LEIBNIZ UNIVERSITÄT HANNOVER

FACULTY OF ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

HUMAN-COMPUTER INTERACTION GROUP

SIMPLIFYING THE WAKE WORD USAGE IN CONVERSATIONAL USER INTERFACE

A Thesis presented for the degree of Bachelor of Science

by

Amrit Gaire

03200280

August 2020

First Examiner :	Prof. Michael Rohs
Second Examiner :	Prof. Wolfgang Nejdl
Supervisor :	Shashank Ahire, M.Sc.

EIDESSTATTLICHE ERKLÄRUNG

Hiermit versichere ich, die vorliegende Arbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die wörtlich oder inhaltlich aus den Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Hannover, August 2020

Amrit Gaire

ABSTRACT

The use of voice assistants and smart speakers are dramatically increasing every year. The inbuilt voice assistant apps on smartphones make it further accessible. Whoever owns a smartphone could easily interact with voice assistants by allowing the wake word detection or pressing home buttons on their phone. The user could then perform tasks like asking for weather updates, playing music via an integrated app, performing web searches, and many more efficiently with a voice command.

The wake word is used to let such Intelligent Personal Assistants (IPAs) know that the command is directed towards them. On the one hand, the use of wake words helps the IPAs as it signifies the conversation is going to happen. On the other hand, it reduces the unnecessary involvement of IPAs by ignoring everything that does not start with wake words. Moreover, personalized wake words could even be significant to minimize the risk of speech-based attacks and other privacy problems.

However, the easiness turns out to be monotonous and a little awkward if someone has to repeat the wake word again and again within a short period. Thus, the question arises, "Can't we just avoid the repetition of wake words?" "Does the avoidance of wake words make the interaction easier?" If "Yes" then how and in what context?

The project finds out the most used context via a survey where users were asked various questions like the task they most performed, wake-word problems, opinions on continuous listening features. The result of the survey, along with results from other sources, were then analyzed to find out that the user performed web search and listened to music via voice assistant quite often.

An android app was developed, which could allow the user to remain in the contexts (Web search and Spotify playback control) chosen for the desired time and without any wake word repetition. A user study is then performed where user experience is explored using qualitative methods. Users having prior experience with voice assistants were part of the study. All of them agreed on the ease in interaction with no wake word repetition. However, the duration of interaction also made a significant difference.

Furthermore, the user liked Spotify's playback control with the freedom to place the commands anywhere in their phrase while completely avoiding wake words. Thus, the study finds out that the avoidance of wake words in specific contexts is possible, and the resulting experience is more efficient than with the repetition of wake words.

ZUSAMMENFASSUNG

Der Einsatz von Sprachassistenten und Smart Speakers nimmt von Jahr zu Jahr dramatisch zu. Die eingebauten Sprachassistenten-Apps auf Smartphones machen es weiter zugänglich. Wer ein Smartphone besitzt, kann problemlos mit Sprachassistenten interagieren, indem er die Erkennung von Schlüsselwörtern ermöglicht oder die Home-Tasten seines Telefons drückt. Der Benutzer kann dann Aufgaben wie das Anfordern von Wetteraktualisierungen, das Abspielen von Musik über eine integrierte App, das Durchführen von Web Suchen und vieles effizienter mit einem Sprachbefehl ausführen.

Das Schlüsselwort wird verwendet, um solche intelligenten persönlichen Assistenten (IPAs) wissen zu lassen, dass der Befehl an sie gerichtet ist. Einerseits hilft die Verwendung von Schlüsselwörtern den IPAs, da dies bedeutet, dass die Interaktion stattfinden wird. Auf der anderen Seite wird die unnötige Beteiligung von IPAs reduziert, indem alles ignoriert wird, was nicht mit Schlüsselwort beginnt. Darüber hinaus können personalisierte Schlüsselwörter sogar von Bedeutung sein, um das Risiko sprachbasierter Angriffe und anderer Datenschutzprobleme zu minimieren.

Die Leichtigkeit erweist sich jedoch als eintönig und etwas umständlich, wenn jemand das Schlüsselwort innerhalb kürzester Zeit immer wieder wiederholen muss. Daher stellt sich die Frage: "Können wir nicht einfach die Wiederholung von Schlüsselwörtern vermeiden?" "Erleichtert die Vermeidung von Schlüsselwörtern die Interaktion?" Wenn ja, wie und in welchem Kontext?

Das Projekt ermittelt den am häufigsten verwendeten Kontext anhand einer Umfrage, in der den Benutzern verschiedene Fragen gestellt wurden, z.B. die von ihnen am häufigsten ausgeführte Aufgabe, Probleme mit Schlüsselwörtern und Meinungen zu Funktionen für kontinuierliches Hören. Das Ergebnis der Umfrage wurde dann zusammen mit Ergebnissen aus anderen Quellen analysiert, um herauszufinden, dass der Benutzer häufig eine Websuche durchführte und Musik über den Sprachassistenten hörte.

Es wurde eine Android-App entwickelt, mit der der Benutzer in den für die gewünschte Zeit und ohne Wiederholung von Schlüsselwörtern ausgewählten Kontexten (Websuche und Spotify-Wiedergabesteuerung) bleiben kann. Anschließend wird eine Benutzerstudie durchgeführt, in der die Benutzererfahrung mithilfe qualitativer Methoden untersucht wird. Benutzer, die bereits Erfahrung mit Sprachassistenten hatten, waren Teil der Studie. Alle waren sich einig, dass die Interaktion ohne die Wiederholung von Schlüsselwörtern einfach war. Die Dauer der Interaktion machte jedoch auch einen signifikanten Unterschied.

Darüber hinaus, findet der Benutzer die Wiedergabesteuerung von Spotify effizient, in dem er die Freiheit hatte, die Befehle an einer beliebigen Stelle in seiner Phrase zu platzieren und dabei Schlüsselwörter vollständig zu vermeiden. Die Studie stellt somit fest, dass die Vermeidung von Schlüsselwörtern in bestimmten Kontexten möglich ist und die daraus resultierende Erfahrung effizienter ist.

We have seen that computer programming is an art,
because it applies accumulated knowledge to the world,
because it requires skill and ingenuity, and especially
because it produces objects of beauty.

— Donald E. Knuth [Knuth 1974]

ACKNOWLEDGMENTS

Foremost, I would like to thank Shashank Ahire and Professor Michael Rohs, and HCI team for allowing me to work on this topic. Shashank has been a great supervisor and guided me well throughout the project. He was always available whenever I needed some advice.

I am also very grateful to all 11 participants who gave their time and effort, along with great feedback during user study. I would like to thank them all for sharing their user experience.

Additionally, I would like to thank my parents and brother Prabin for supporting me throughout this journey. I would also like to thank my friends Natalia, Bishal, Deniz, Didier, and Anish for being there by my side.

Finally, I would like to appreciate Andre Miede and each of them involved in creating this beautiful *ClassicThesis.tex* template, which helped a lot. Last but not least, my humble gratitude goes to the StackOverflow community. Although I did not ask myself any question, their older post helped me whenever I was stuck.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACRONYMS

- **API** Application Programming Interface
- **CUI** Conversational User Interface
- **DPA** Digital Personal Assistant
- **GNN** Graph Neural Networks
- **IPA** Intelligent Personal Assistant
- **SDK** Software Development Kit
- **UI** User Interface
- **VPA** Virtual Personal Assistant
- **VUI** Voice User Interface

1

INTRODUCTION

1.1 CONVERSATIONAL USER INTERFACE

Talking to a virtual assistant is no longer a science fiction one used to watch in the early Star Wars movie series. Today, a smartphone or a smart device is all one needs to interact with voice assistants. The term conversational Interface refers to the ability of such a natural way of interaction with smart devices using spoken language [McTear et al. 2016]. Terms like Digital Personal Assistant (DPA), Intelligent Personal Assistant (IPA), voice assistants used in this literature refer to the voice-based interaction with any of those applications or devices. These terms are used quite interchangeably as it has gone mainstream. The intelligence to understand the natural language and higher integration features have increased their potential and popularity. Thus, the use of voice assistants is on the rise as most companies have developed their versions.

"OK Google, ...", "Hey Siri, ...", "Alexa, ..." are some of the most uttered words followed by a query or a request while interacting with renowned voice assistants in 2020. Apple's (Siri), Microsoft's (Cortana), Google Assistant, and Amazon's (Alexa) dominate globally, with the market on course to exceed 2.5 billion shipments by 2023 [Futuresource 2019].

After the introduction of Apple's (Siri) in 2011, the emergence of voice assistants led the tech innovation to a different height. The competition between the brands for a more significant share impacts the exponential growth of this technology. May it progresses around the text-to-speech synthesis or continuous improvement to make the voice assistant less robotic, there has been much investment for better use. Natural Language Understanding (NLU) or Automatic Speech recognizer (ASR) is changing the way of Human-Computer-Interaction. The implementation of AI has improved the ability to understand the context and make the interaction more human-like. The introduction of emotions into Amazons Alexa [Furey and Blue 2018] in conveying empathy is not science fiction anymore.

No wonder voice assistants are now demanded and implemented in every industry ranging from healthcare to banking and entertainment. The voice recognition ability has also attracted students and researchers. The automobile industry is also eager to include voice assistant features adapting to their needs [Braun et al. 2019].

As CUI's popularity is increasing, the voice user interface is being embedded both into everyday life mobility and into the life of home via an assistant device like Google Home and Alexa [Porcheron et al. 2018]. One has to agree on specific terms and conditions like sharing contacts and location and allowing audio records to use the inbuilt voice assistants with almost all smartphones and tablets. As Siri and Google assistants are capable of performing basic tasks like web search, setting reminders and alarms, sending messages, and a phone call, just to name a few, it does require specific structures or commands to follow to get these things done efficiently. However, basic patterns of interaction needed in almost all voice assistants to date are the same.

1.2 CUI BRANDS AND THEIR STRUCTURE OF INTERACTION

One could argue the whole day, which is the best among tech superpower (Google, Apple, Amazon) but could not omit any of them from the top three. Though Amazon came late on the market with its Alexa Echo smart speakers, it has competed highly against all other competitors. Google launching its Google Home as smart speakers and voice assistant application on smartphones has advantages over others because of android phones having Google Assistant as an inbuilt application. At the same time, Apple seems to be satisfied with Siri being the charm of its different device models.

As these firms are trying to be the prize amongst the users introducing new features and improvements now and then, they do share the same patterns and structure of commands one needs to follow through to perform a particular task. Siri needs to be activated either by uttering "Hey Siri" or pressing the home button in order to perform any task. While Google seems to love its name, it wants to listen to "Hey Google" or "OK Google" every single time before a user pleads another query. The same goes for Amazon's Alexa as they all require a wake word activation followed by a query or request.

When Google launched its Google Home device with features like broadcasting something to every other Google Home device embedded in the house, consumers were rather demanding custom wake word features instead of "OK Google". People even find alternative phrases that sound like "OK Google" and activates the device [1]. Samsung's Bixby allows it to be woken by "Hey Galaxy" or "Hey Bixby", while Amazon Alexa also provides other wake words like "Amazon", "Echo", or "Computer". However, there is no sign of changing the pattern of interaction. Although Google's assistant can now have some 8 seconds of follow up time compared to Alexa's 5 seconds for a subsequent query, it is still not convincing in some contexts. While Alexa can whisper now [Raeesy et al. 2018], it has shown some signs that they would like to give the

1 Alternative Phrases for OK Google that works: https://9to5google.com/2020/01/14/weird-hey-google-alternatives/

freedom of putting wake word anywhere in the query by registering a dynamic wake word patent [Amazon Technologies 2017].

However, one could always raise the privacy regarded questions, which could be the limiting factor for not simplifying the wake words usage. But someone who often needs to interact with a voice assistant for a web search or controlling basic playback of music for a dedicated time would want to avoid the wake word repetition.

1.3 MOTIVATION

After scratching the surface of the conversational user Interface and being introduced with the vision of a seamless conversation with intelligent voice assistants [Ahire and Rohs 2020], I could not help but dive deep into the topic. The idea of not having to utter wake words again and again for dedicated time is quite impressive. Besides, as mentioned in the same paper, the freedom to choose some context and to be able to follow through requests without repeating wake words could make a conversation with a voice assistant human-like.

Other than that, very few research on wake words also intensified my interest in the field. While all the big tech firms are concerned about implementing machine learning and AI, collecting all-dimensional data, introducing emotions into the smart speaker and making them able to whisper back just to make the interaction as that of human to human, they seem to forget the first step towards seamless conversation; the freedom and flexibility to use the wake word.

Furthermore, it was a pleasure to be able to do my research on the ardent topic. To enjoy working on the problem to simplify the wake word usage and build something for a specific context was itself a huge motivation.Although there has been much work going around to customize the wake words, I am amazed to see very few being concerned about its repetition. Besides, the current voice assistants do not have a dedicated mode where one could activate with wake word once and keep on engaging without uttering it anymore till the desired time.

So, I decided to choose two of the contexts, as described in the earlier paper [Ahire and Rohs 2020]. The first context will allow users to perform a web search and the second to control the basic playback functions while listening to music. The users could then remain in a context for the desired time without necessarily having to repeat the wake words.

1.4 OBJECTIVE AND TASKS

The objective of this project is to study whether wake words are the barrier to seamless conversation. It is an attempt to see if it helps to have a dedicated

context where a voice assistant is enabled for continuous recognition avoiding wake words to its minimum. Moreover, it is an effort to test the effectiveness of avoiding wake words while interacting with the voice assistant within a particular context. To get started, I need to find what contexts users usually prefer, and if the repetition of wake words in that context could be minimized. Comprehending that the development of an android app is followed by a user study to get better insights into the user experience. So, to be able to perceive the context, there are some research questions to be answered:

- Do IPAs always need a wake word?
- Can we avoid the utterance of wake words in some context?
- What context could the wake words be helpful?
- Does the avoidance of wake words ease the interaction with IPA?

Foremost, the task is to dive deep in these research questions to understand the wake words problem. Besides, it is always essential to have a broad view of the topic. So, understanding the user's opinion and their problems regarding wake words on voice assistants through a survey is an imperative factor to roll the stone in the right direction.

2

RESEARCH REVIEW

2.1 WAKE WORDS, CONTINUOUS RECOGNITION AND CHALLENGES

Eye contact and a smile are all it takes to signify the start of a conversation within humans. There are body language and certain expressions one can convey to let the other person know that the conversation is going to happen [Seo and Koshik 2010]. Sometimes the context alone dictates one person is talking with the other. However, when it comes to having conversations with IPAs, there are limitations. Just like in human conversation, there is a need to provide a cue that signifies the start of an interaction. While there are different ways like approaching, making voice adjustments and looking to grab someone's attention to initiate human interaction [Seo and Koshik 2010], human-computer interaction has an initiation problem [Shi et al. 2011].

The complications to determine whether the utterance is directed towards the IPAs further enhance the dependency on wake words. The use of wake word started initially to avoid the push-to-talk to switch the context explicitly as it moved towards achieving the natural speech interface's goal [Këpuska 2011]. Thus, wake words became an explicit way of obtaining attention from IPAs gaining contextual meaning [Jung and Kim 2019]. There are also specific social situations where wake words could be significant to protect someone's privacy or to avoid some illegal handling. Since most of the smart speakers are now able to make an online purchase on command, this feature could be easily misused without personalized voice recognition [Candid 2017]. There have also been numerous cases where kids have ordered Gifts online via their parent's smart speakers [Candid 2017]. Personalized wake words can be great in order to avoid such a situation. However, wake words also became the way to initiate and stop the conversation acting like a repair button [Jung and Kim 2019], further increasing the repetition. Very little research has been done around the repetition problem as it is on its early rise, while the evaluation of IPAS is still mostly based on task, but not user satisfaction [Berdasco et al. 2019].

The study [Ammari et al. 2019] analyzed 82 Amazon Alexa device's log files and found out that out of a total of 193,655 commands given 51,491 consisted only of wake words like "Amazon", "Alexa", "Echo" and "Computer". However, the same study states that Amazon devices having problems with false triggers could be the reason behind it. The study's primary purpose was to discover how people use voice assistants and it reports: music, search, and IoT were the most performed tasks. Interestingly, the authors mention that they had to

use the wake words as stop-words while calculating the TF-IDF values of the terms as the repetition of wake words were highest. They found that terms like "stop", "play", "skip", "shuffle","song", "lullaby", "music", "sing", "radio", "pause" had the highest TF-IDF value in the music category.

Furthermore, the other problem regarding wake words is also false activation. This study [Dubois et al. 2020] provides the answer to the misconception people generally have about continuous recognition. The study states that the devices are not continually recording but could record up to an average of 8 seconds and send the recording to their respective cloud service if falsely activated. Although there is lots of research going under different sectors of the CUI field, repetition of wake words has not attracted many researchers.

Conversely, there has been ongoing research and a high level of investment in order to detect those cue words correctly. Apple went on to develop a personalized "Hey Siri" in order to improve the accuracy of its key phrase detection [Apple 2018]. Furthermore, lattice-based improvements for voice triggering using Graph Neural Networks (GNN) have already been proposed, which could be further used for user-intent classification [Williams 2020]. Similarly, Google recently release a new feature where one could set the sensibility for "Hey Google" recognition [Li 2020].

Figure 1: Voice Recogniton Market, Source: CB Insights

While top tech firms are trying to capture the $49B voice market[CBInsights 2019] (Figure 1), investment has been made by each of the firms in a variety of areas like accuracy and language to name a few, but no tech firms seem to have any in-depth research on wake words repetition problems. As mentioned earlier [Amazon Technologies 2017] Amazon has filed a patent for the dynamic wake word, but time shows when it will be implemented on their smart speakers. Google is continually focusing on featuring multiple languages and word accuracy (Figure 2) as it lies behind Apple in terms of accent recognition and

language option, however, a comparison done in terms of answers provided by the top 4 voice assistant (Google Assistant, Siri, Alexa, and Cortana) does show that Google Assistant and Alexa are better than Siri and Cortana [Berdasco et al. 2019].

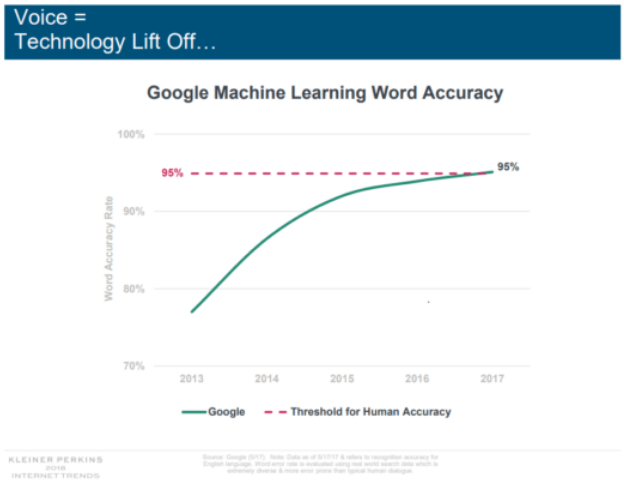


Figure 2: Googles Approach, Source: KBCP

In the meantime, Amazon and Google [Huffman 2018] have already implemented continuous recognition as their new feature on respective smart speakers and voice assistants. Users could follow up with their queries and continue conversation remaining in the context. There is some relief, but still, there are specific contexts where one could wish their voice assistant to just listen to and follow the commands.

However, the implementation of continuous recognition in voice assistants does reflect that Google and Amazon realize the problem with the repetition of wake words. But, the short follow up time reflects the problem of continuous recognition in a different direction than a technical one. The report from this study [Edu et al. 2019] suggests that most of the speech synthesized attacks are made by exploiting the wake words and continuous listening features of smart speakers.

First and foremost is the privacy problem. Microsoft Market Intelligence team reported that 41% of users were concerned about trust, privacy, and passive listening [Olson 2019]. Besides, both Google Home and Alexa got into controversies due to the recording of private conversation and sending it to random people [Cuthberstson]. Although there has been a clear explanation of such a situation as a mistake, there is always a question about spying for commercial purposes [Abdi et al. 2019]. Mainly, no one wants to be eavesdropped, which lies diametrically in the opposite direction of continuous recognition. Surprisingly, most of the smart speaker users have a false mental model about recordings of their interaction; some thought their data would be automatically deleted after

a short time, and others did not know they could review their interaction and delete them [Malkin et al. 2019]. Similarly, this study [Zeng et al. 2017] shows that users' mental models about privacy concerns highly depended on technical background knowledge. The majority of the users are confused about their data storage, and how long the providers keep the data [Abdi et al. 2019].

The trust issues not only remain on the privacy and data monetization but primarily also on the ability of IPAs to perform a primary task [Cowan et al. 2017]. The same study mentions that despite the technical advancement, there has been little work around the user's experience of IPAs. The study mainly focuses on the interaction with IPAs performing different tasks and reports the issues around the interrupting nature of hands-free actions, about accents and speech recognition, embarrassment during social use. Similarly, the authors of "Like having a really bad PA" [Luger and Sellen 2016] presented that users tend to avoid using IPAs in social situations. While the reason may be the cultural norms presented in [Cowan et al. 2017] stating that the interaction between humans and IPAs is still not socially acceptable for various reasons, the question arises as to how much of that could be because of the repetitive use of wake words.

To summarize, there has been very little research on the context of avoiding repetition of activation words in contrast to detecting it with accuracy, and privacy concerns. As important as it is for the user to have a great experience using smart speakers and voice assistant applications, the same goes for the big firms to increase the engagement of users with such devices. Thus, if the future is moving towards the seamless conversation between virtual assistants and humans, there should be a balance between continuous recognition and repetition of cue words.

3

BUILDING CONCEPT

3.1 SURVEY AND ANALYSIS OF THE RESULT

Before beginning with a process of developing a concept, a survey was done amongst 23 participants within the age group of 18-35. The purpose behind the survey was to find out the tasks users frequently perform using their voice assistant, which could further help in choosing a context. Aside from that, it was essential to know whether users would like to avoid wake word repetition for a particular time. The survey questionnaire was built around to find the general problems users face with voice assistants and also particularly the problems with wake words.

It was an online survey. Although there were 24 participants in total, 1 of the participants had never used a voice assistant and replied "no" to every question. For the convenience and to justify the answers recorded by those who had the experience, that response was deleted. Following charts shows the questions and their responses:

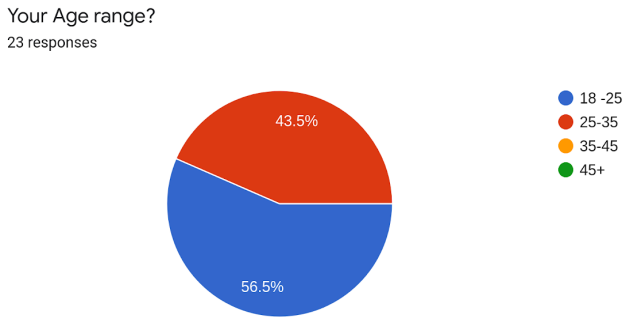


Figure 3: Age group of participants

Thirteen of the respondents are from the age group 18-25 years, while the rest of the 10 are form 25-35 (Figure 3). All of the 23 participants mentioned that they have experience using voice assistants or smart speakers. As in Figure 4, eight of the participants say that they use it rarely (1 being once in a month), while only 2 of them claimed to use it daily (5 being daily). The rest 13 vary from once or twice in a week.

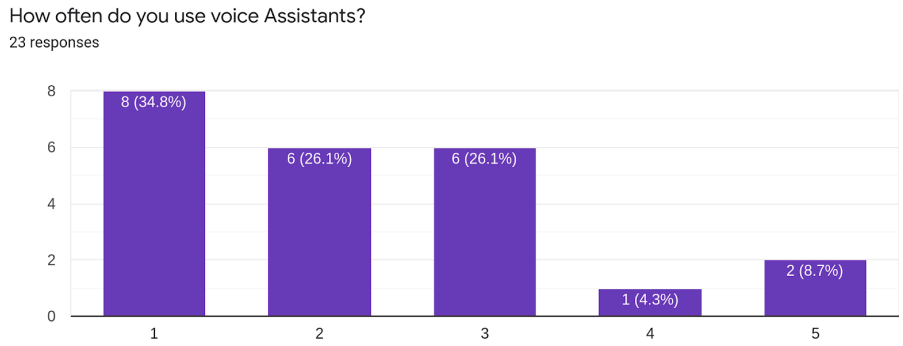


Figure 4: How often users interact with IPAs?

The question in Figure 5 was a multiple-choice question. Answers were predefined along with an option to mention any if it was missing on the choices given. While setting alarms and remainder topped the chart, educational purposes, asking random questions, and weather updates are also performed regularly. However, playing music is surprisingly one of the least performed tasks within those users.

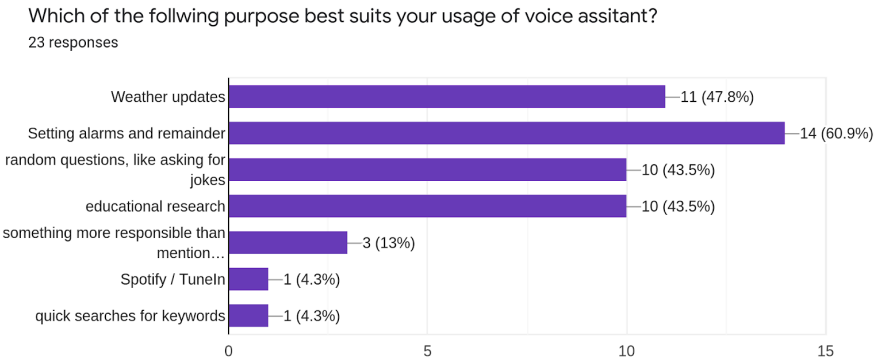


Figure 5: Frequent activities performed by the users.

More than half of the participants say they would like to have a custom wake word (Figure 6). To the follow-up question where one could reply with their favorite name, 4 of them replied that they don't know yet, while the rest of them suggested using a pet name, some rock-stars name or nicknames, or even a joke name like a derivative of a friend. Almost half of them would like to have a feature where voice assistants could listen to them for a certain period of time, while a little more than half are conscious about it (Figure 7).

To the optional question whether there was any other problem regarding voice assistant, the 7 responses given, are as follows:

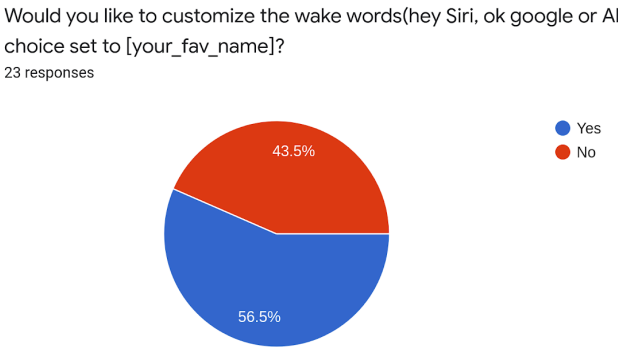


Figure 6: Users opinions on custom wake words.

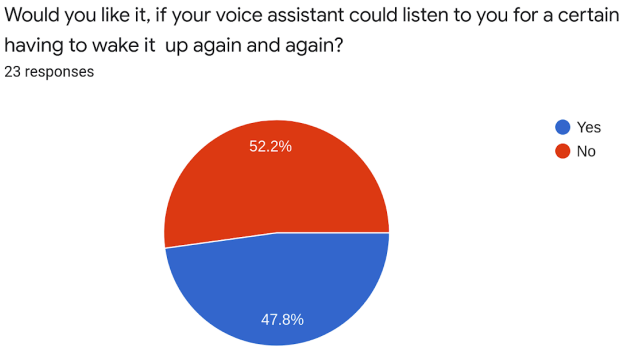


Figure 7: Users opinions on continuous recognition.

- "Gets activated way too often without saying the wake word."
- "No"
- "That it hears/registers the talking although I don't use the wake words"
- "sometimes Alexa gives the wrong answer to my carry and when I correct her she does not remember"
- "It sends over the FBI way too often due to misunderstandings :("
- "Doesn't recognize words properly and displays different results than expected"
- "Ascent problem in different languages"

Ignoring the humorous answer provided by the user enjoying their anonymity, some of the problems seem to be genuine ones or better said familiar ones. Wake word being falsely detected, not recognizing the words properly, and ascent problems are actually being researched and improved by big tech firms as mentioned in Chapter 2.

3.2 RESULTS FROM SOME OTHER SOURCES

Regarding the disadvantages of an open survey and the number of participants, there was a need to find some more prominent sources. According to the survey [NatioalPublicMedia 2020], the number of smart speakers or voice assistant users has even increased during the corona pandemic. Their most recent survey result shows that playing music and getting weather information is one of the most performed tasks (Figure 8).

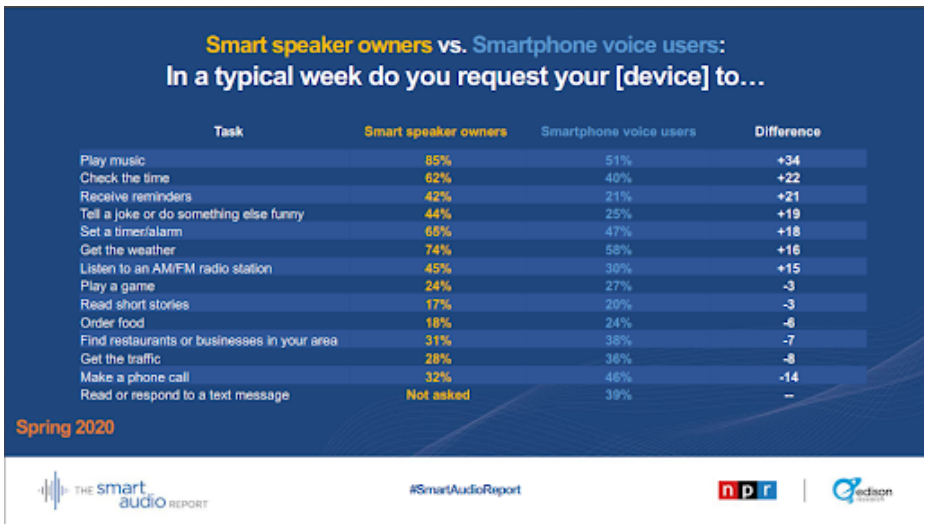


Figure 8: Most performed task in smart speaker vs Voice Assistants

Similarly, 65% of the participants from the same survey agreed that they would prefer a smart speaker with a screen rather than one without (Figure 9). Thus, it signifies visual preferences, although the consumption of audio content is on the rise.

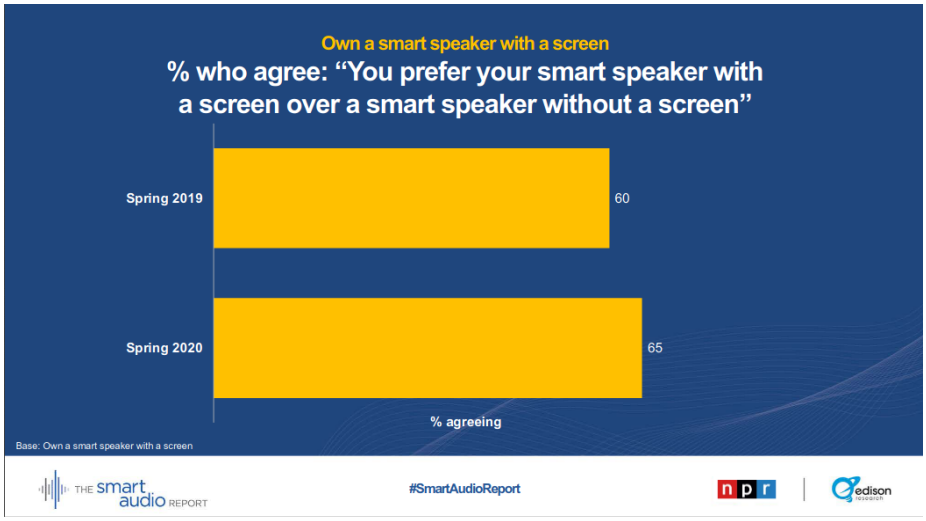


Figure 9: Smart speaker with or without screen

3.3 CHOOSING CONTEXT AND FINDING SOLUTION

The results from Section 3.1, coupled with the one in Section 3.2, reflect that users ask their voice assistant to perform web searches and play music quite often than the other activities. Although setting remainder and alarms topped the chart of the survey performed, it would be the least relevant context where one needs continuous recognition. Requests like finding nearby restaurants and asking for weather updates could be easily merged as a web search.

Therefore, web searches and playing music are chosen as the context in the app. Both of the contexts demand continuous recognition. Analyzing the voice assistants or smart speakers to a prior date, both of the contexts need wake word repetition. So, being able to reduce the repetition of wake words while using both contexts provides a direct possibility to compare the experience. Thus, enabling to answer whether the avoidance of wake words eases interaction with IPA in specific contexts.

3.3.1 *Problems with current IPAs*

As the context is chosen, some problems, as mentioned earlier, are stated by the participants of the survey, along with general problems that need to be considered while implementing the solution. Some of them are as follows:

- The false activation of wake words
- Repetition of wake words for a subsequent request
- Mandatory placement of the wake words at the beginning of the request
- Customization of wake words
- Not remembering the context for the follow-up questions
- Not recognizing the queries and displaying different results than expected
- Not recognizing the accents of the user

Although solving all these problems could be tempting, the project's time and specification have some constraints. Solving problems like remembering the context and reacting intelligently to the follow-up question is beyond the expectations and specifications of this project. However, the rest of the problems could be solved or at least reduced.

3.3.2 *Failed Approach*

The first approach was to modify existing voice assistant apps so that the wake words repetition could be hindered. The attempt to work with Google Assistant failed miserably and cost lots of time. The main goal was to be able to get continuous listening feature work. However, Google Assistant required a manual "start" and "stop" before every query; thus, failing to achieve the most desired feature. The option to integrate Dialogflow was explored to avoid the manual start and stop mechanism each time. However, Dialogflow required an individual extension for each context with predefined intent and actions. Thus, deviating from the specification will be two extensions from Google Assistant and not an app. Other open-source voice assistants like Robin, Aaya, and many more presented similar problems. Either continuous recognition did not work at all, or they were faulty, or demanded higher skill to get it to work.

3.3.3 *Final Approach*

The solution that avoids wake words in the chosen context gets the highest priority as the hypothesis of this study revolves around creating such experience and comparing it. After failed approaches to modify the preexisting voice assistant apps, developing a new one is considered. The idea is then to develop a new app that could be activated by a wake word, be in the chosen context for the desired time, and listen continuously.

The app's main objective is to perform the task with very minimum repetition of wake words per query while remaining in a context. First, it detects the wake word spoken by the user. It then enters a web searching context. Users can perform voice queries from this context. The recorded speech gets synthesized to text and gets passed to perform a web search. The results get displayed on the new tab. Users can get back to the context by pressing the back button and could perform another search without necessarily uttering the wake word. The loop runs until the user desires to change the context by asking to open Spotify with the command "open Spotify." In this case, the app will launch the Spotify app in the background and listen to the playback controls.

3.3.4 *Tools, Library and their selection*

- Android Studio
- Text to speech
- Speech to text
- Voice recognizer

- DroidSpeech2.0
- PocketSphinx-5prealpha
- Spotify remote SDK

Technologies like speech to text, text to speech, voice recognition are essential requirements for this project. Furthermore, continuous and real-time recognition is required to avoid manual start and stop mechanism for better user experience. An offline voice recognition procedure to detect the wake word is ideal considering privacy concerns, and the in-determinism nature of waiting, while the online voice recognition engine passes the query as text, to perform the web search.

The failed approach to use Google Assistant already gave the knowledge about google speech recognition API. There were options like Microsoft Speech Recognition Service, IBM Watson, Speechmatics, and many more. However, the third-party library, "DroidSpeech2.0", fills the void of Google's continuous recognition. It does not require the manual "Start" and "Stop" and protects from the unexpected error that occurs with Google's speech recognition whenever the new update is there. Thus, the library is selected to get real-time speech recognition.

Similarly, CMU Pocketsphinx Android Library [Huggins-Daines et al. 2006] is used to detect the wake word. It is entirely offline, free, and allows keyword as well as key-phrase detection. Its portability and speed and the freedom to make its dictionary are essential factors to include it in this project. The updated version PocketSphinx5Prealpha [1] is used on this project. The playback of music should run on the background, and the listening process could be for hours. So, this offline recognition library would support the long term service.

Consequently, to play the music, third-party app Spotify is chosen as it is one of the most used apps in the category music and audio with more than 286 million monthly active users. [Iqbal 2020]. The Spotify Android Software Development Kit (SDK) is used to interact with the Spotify app running in the background. The API provides the metadata and the context of the currently playing track. It allows initiating playback of tracks and can issue basic playback commands.

Finally, after deciding the technology to use, it is time to design the User interface and put all the pieces together. But before that, at least one project solved the repetition of the wake word problem in some new way, which is described in the section below.

1 Pocketsphinx-5Prealpha `https://sourceforge.net/projects/cmusphinx/files/pocketsphinx/5prealpha/`

3.3.5 *Similar Projects*

There are lots of projects on a different scale that work around customizing the wake word problem. Nevertheless, one of the projects built around minimizing the wake words repetition is Houndify (Figure 10). It has features to customize the wake words, and the conversation gets also activated with an eye-contact. Thus, no need for a wake-word if the user would be able to look at the screen of their phone.

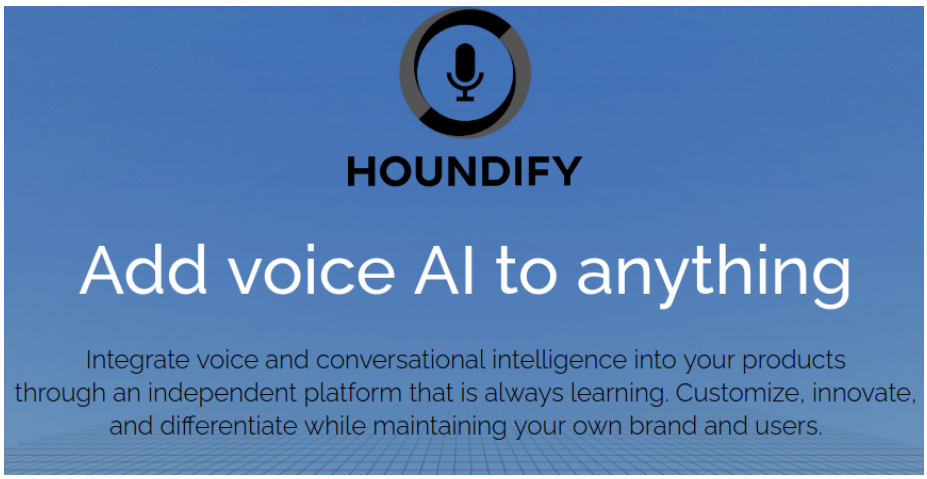


Figure 10: Houndify Voice Assistant, Source:SoundHound

4

INTERACTION DESIGN AND IMPLEMENTATION

4.1 DESIGN OF USER INTERFACE (UI) : ITERATION 1

The raw design of the user interface came from the concept and the requirement for this project. The minimalist way is chosen not to confuse the users and include only the required features. Three consecutive scenarios are imagined to happen as the user proceeds through choosing and performing a task. The first one is the "Welcome screen" where the user should speak the wake word and can set wake-word sensibility, the second one is the "Query screen" that requires users to speak their query, and the third one is the "Playback info screen," which presents the Spotify playback status to the user while Spotify is pounding music on the background (Figure 11).

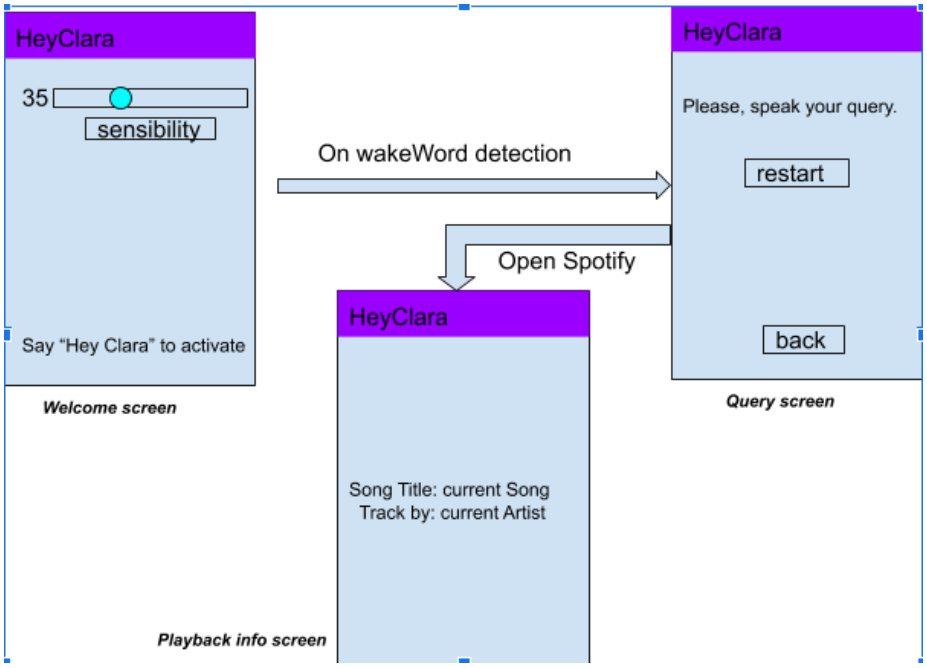


Figure 11: Raw design of User Interface (UI)

The web search is performed to all the queries except the one that includes "open Spotify." Thus, the fourth screen is involved, and it displays the Google search and its result (Figure 12). On back pressed on their phone, users can go back to the web search activity and inaugurate it again.

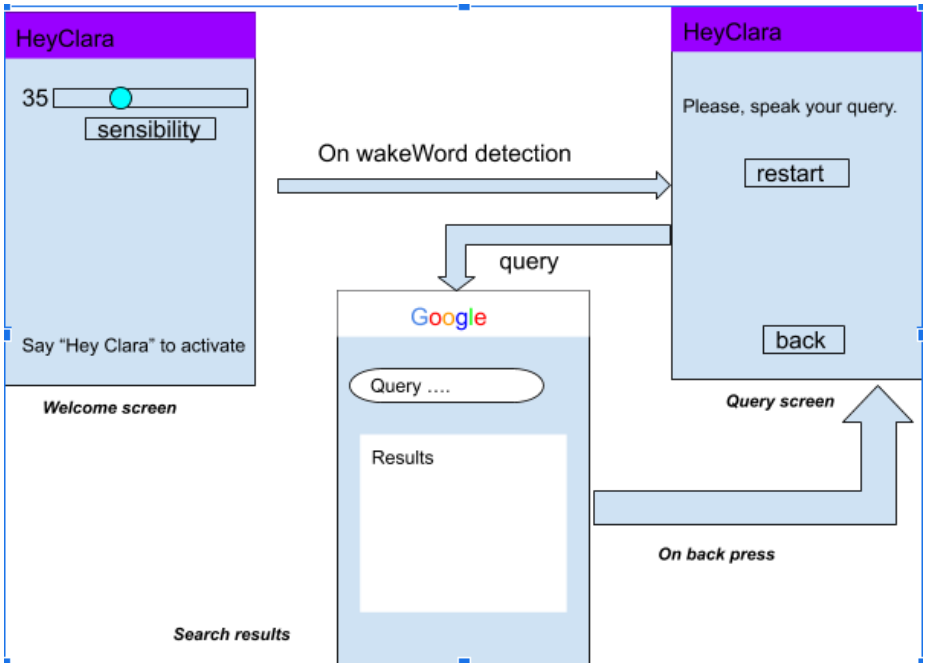


Figure 12: Raw design of UI; WebSearch Activity

The Welcome screen has the app name "HeyClara" at the top bar, while at the bottom, it pops up the key-phrase message to the user to hint what to say to wake the app up. Furthermore, it has a live tracking seek-bar for adjusting the wake word's sensibility while text-view on its left shows the current numeric value.

Users get to the Query screen after the app detects the key-phrase. The app will then greet the users, and the display will show a text preview of "Please, speak your query," which later shows the preview of the final result speech recorded. The "restart" button and the "back" button are there for their respective function.

The Playback info screen is responsible for showing the metadata information such as the song's title and its artist. All the screen design uses XML.

4.2 IMPLEMENTATION : ITERATION 1

Finally, the first design iteration covered UI's basics, and now it was time to implement. Android Studio became the first choice to create a project as it is an android app. It was easy to create the project using IDE. "HeyClara" is the given name. As the front end design was ready, it helped to implement the backend section.

The app has three activity classes. The "MainActivity" is the launcher, and it shows the "Welcome Screen." Besides, it is responsible for listening to the wake word and detecting it. "Hey, Clara," is the default wake word. The "Web-

SeacrhActivity" is responsible for recording the query and performing the web search intent controlled by a "defineAction" method. To control the playback of Spotify, "LaunchSpotify" is responsible.

4.2.1 *The MainActivity*

As soon as the app is opened, it listens for the wake word. Users could also adjust the sensibility of the wake word to ease the detection process or tighten it. The screen suggests a pop-up text to users to speak, which shows the key-phrase information. Figure 11 shows a sketch preview.

Similarly, the back end needs to start listening and check for the correctness of the uttered wake word. The android PocketSphinx Library records the spoken word and handles the accuracy of detection, while the displayed seek-bar handles the adjustment of sensibility of wake-word detection. Implementing a seek-bar for sensibility and overall implementation is based on this projects [Paulus 2017] and pocketsphinx-andorid-demo[1]. It tracks the changes in current time and sets the sensibility of detection within a range from 1 to 100. 1 being very passive and allowing very few detection, resulting in missing the rightly spoken words. The detection rate increases as the value slide towards 100 but also results in false triggers. The default value is 35. This fragment of the app is active until it detects the wake-word. After detection, it opens the fragment, which is responsible for taking queries.

4.2.2 *The WebSearchActivity*

Right after the wake word is detected, the "Query screen" (Figure 12) is displayed. The speech to text technology used greets the user with "Yes please, how can I help you?". At the same time, the screen displays "Please, speak your query" text. The restart button is right in the middle of the screen, while the back button is at the bottom middle.

Meanwhile, the instance of Droidspeech (the android library used for continuous detection) initializes and starts listening to the spoken words. It then passes the final result of spoken words converting speech to text to a method. The method "defineAction" is implemented to differentiate which action should be performed, checking the query. According to the query content, whether to search on the web or to open Spotify is decided.

If the method decides for web search, the intent web search is opened and searches the google for the query. Users can come back to the intent to search another query again by pressing the back button on their phones. The speech recognition will be listening to the query as soon as the user returns to the

1 pocketsphinx android demo `https://github.com/cmusphinx/pocketsphinx-android-demo`

"Query screen." If the recognition seems to be off and not listening, users can click the "restart" button while clicking the "back" button will lead them to the welcome screen.

4.2.3 *The LaunchSpotify*

However, if the query contains "open Spotify" within it, then the Spotify app will start on the background; meanwhile, the app displays the "playback screen" (Figure 11) to the users. This playback fragment allows users to control basic playback features. Users could give commands like "Stop," "Next," "Pause," "Resume," "Previous" for controlling the respective action. Users can freely formulate their playback requests where they can place these commands anywhere in their sentences. For example, "please play next song" or just "next" will skip the song and play the next song. Only the query, including the command, as mentioned above, gets recognized. The instance of Pocketsphinx's speech recognizer allows this prolonged and continuous listening and detecting of commands. The recognizer sents the detected command to the method *"playback_control"*, which simply executes the command by communicating with Spotify API. As long as the playback fragment is active, no wake-words are required. However, LaunchSpotify plays the songs from the hard-coded playlist. Accessing the user's playlist is allowed only after authentication, and additionally, a premium account is needed to achieve that feature, which could be a possible drawback for the user study. Nevertheless, the objective of exploring user experience without a wake word repetition could still be achieved.

4.3 PILOT TEST

Finally, it is time to test the app. To get pre-review and feedback, some of my close friends and my supervisor, who had experience using voice assistants, tested the app. This test aimed to find any obscurity in UI and gain feedback on essential features. The testers checked both of the contexts; web search and Spotify playback controls.

4.3.1 *Feedback*

After playing around with the app, the testers proposed the following feedback:

- The welcome screen should have a text that clarifies what to say to activate.
- The "restart" and "back" buttons on the Query screen were unclear of their functionality.

- To have feedback on the Query screen to let the users know it is listening.
- To have an option to cancel a query.
- To let the app run in the background while listening to Spotify and still receive commands for its playback control.

4.4 REDESIGN OF UI AND RE-IMPLEMENTATION : ITERATION 2

Considering the feedback from the tester, the UI is redesigned. The welcome screen now suggests the users speak the wake phrase, and the Query screen suggests that it is listening (Figure 13). Furthermore, the Query screen now has the button "HINTS" that will suggest if clicked on how to use the app. The query screen could also show the live speech preview to the users and offers a canceling option with a regress meter as soon as the recording finishes.

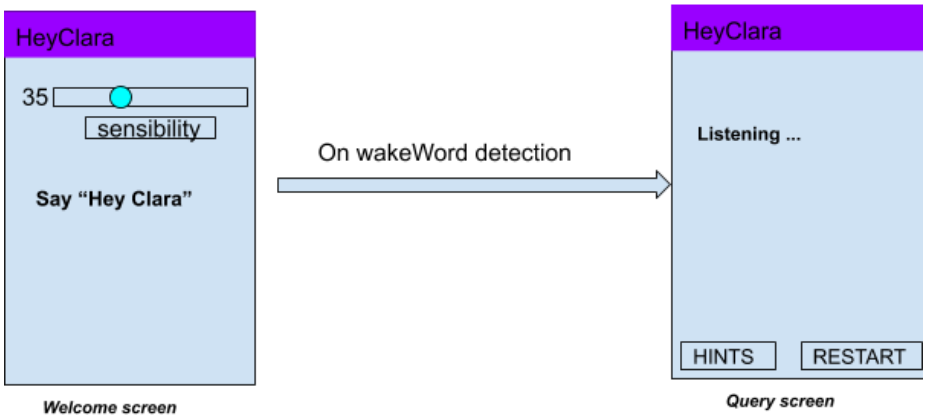


Figure 13: Re-design of UI

The Figure 14 and Figure 15 shows a preview of the query "show me the current weather." The speech preview updates the words spoken as they are recognized. As shown in figure (Figure 15 Query Screen (v)), if the user does not cancel the query, then the web search is performed; otherwise, the cancellation will return to the initial state as shown in figure (Figure 14 Query Screen (i)). Users can continue pleading any further requests from this state following the same pattern.

However, if users include "open Spotify" in their request, the app launches Spotify and itself in the background. But before that, it checks the presence of the Spotify app on the phone. If not found, the activity leads the user to Play-store where they could download it. The user then requires to log in to Spotify. If the user is logged in, they can lock their phone's screen, and still, the LunchSpotify activity will listen to the user's command as long as Spotify and the activity launching it are active.

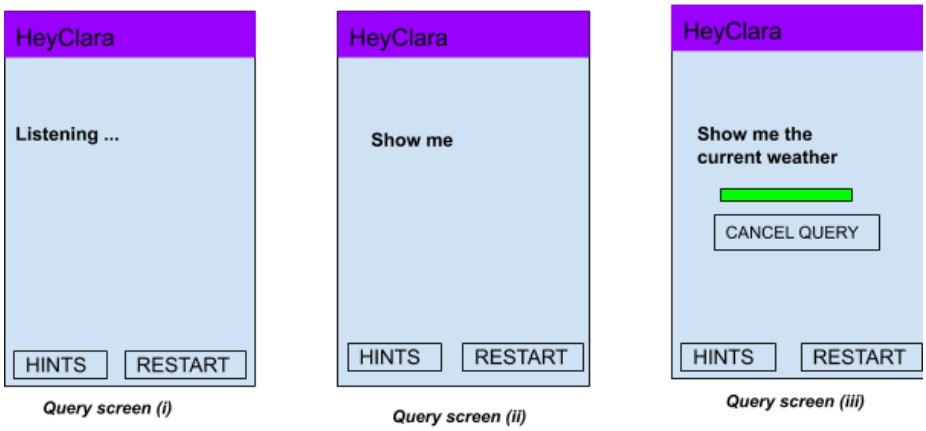


Figure 14: Live Speech Preview and Cancel Option

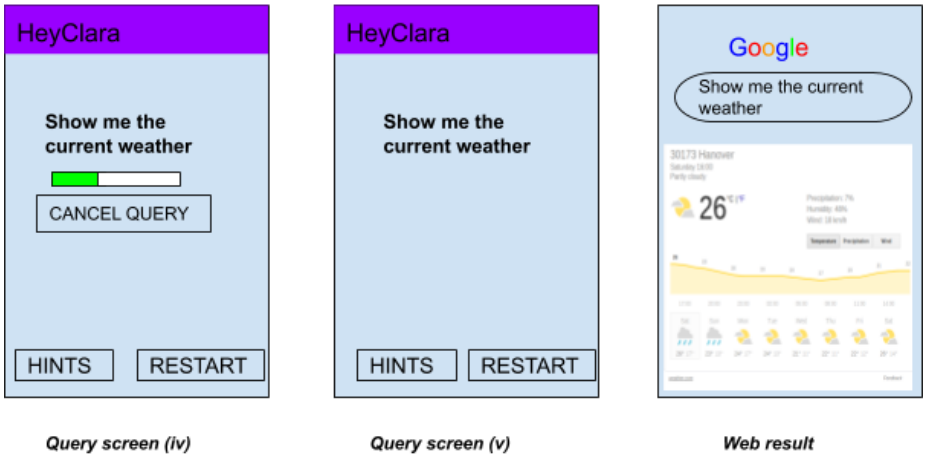


Figure 15: Live Speech Preview and Result

4.5 PILOT TEST 2

After implementing the feedback, the app is tested again on the test device Samsung Galaxy A5 with Android version 8.0.0. The app can now launch Spotify and itself in the background successfully. It will remain on the playback context and receive the playback commands even when the screen is locked. Furthermore, the users could even have control over their playlist with a little effort, where they can go to Spotify and play any song from the playlist after the app launch the Spotify in background.

The customization of the wake word was not possible as the offline dictionary used to detect the wake word is based on a phonetic dictionary. It was thus limiting the option to have one's wake word. There was a possibility of suggesting some names and letting the user choose it from the given option. However, it did not sound like a customization option as the user could still be unsatisfied

with the given names. That is why the focus remains on the repetition and not in the customization.

4.6 USE CASE TABLE AND FINAL DESIGN

The following tables (Table 1 to Table 6) shows use case tables with the different action scenarios of the app. Figure 16 is a screenshot preview of the final design performing an example query: "Can you please find me the current weather?"

Use Case ID	1
Use Case Name	Run the app
Actors:	User
Description	The Welcome Screen will be displayed with a message in the middle of the screen showing “Say Hey, Clara”. It then starts listening to the wake word and moves to Query Screen if detected.
Pre-conditions	The app is successfully installed in an android device running version greater 5.0.
PostConditions:	
Trigger:	The user taps the icon of the app with the name HeyClara.
Normal flow:	1. The user taps the icon of the app. 2. The System loads the Welcome screen.
Alternative flows:	
Extensions:	
Exceptions:	“App stopped working,” If the device does not support it.
Notes and Issues:	

Table 1: Use Case 1: Run the app.

Use Case ID	2
Use Case Name	Activate the app with wake word
Actors:	User
Description	The app will switch to Query Screen if the wake word is detected otherwise, it listens for the wake word again.
Pre-conditions	The Welcome Screen is active.
PostConditions:	
Trigger:	The user speaks the key Phrase “Hey, Clara”.
Normal flow:	1. The System listens to the key-phrase. 2. The user speaks the key-phrase. 3. The system decodes and tries to recognize it.
Alternative flows:	
Extensions:	2.1 If recognition fails, then the system listens again i.e step 1. 2.2 If recognition is successful, then the Query screen is displayed.
Exceptions:	
Notes and Issues:	

Table 2: Use Case 2: Activate the app.

Use Case ID	3
Use Case Name	Adjust the sensibility.
Actors:	User
Description	The Welcome Screen is active and the user adjusts the sensibility by tapping and sliding through the seek-bar.
Pre-conditions	The Welcome screen is active.
PostConditions:	
Trigger:	The user taps on the seek-bar adjuster.
Normal flow:	1. The user taps on the seek-bar adjuster and moves it towards left or right through the bar. 2. The System shows the current value on the left of the bar.
Alternative flows:	
Extensions:	
Exceptions:	
Notes and Issues:	Value towards 100 will trigger more correct as well as false detection, while value towards 1 will trigger less correct as well as less false detection.

Table 3: Use Case 3: Adjust the sensibility.

Use Case ID:	4
Use Case Name:	Search any query on the web.
Actors:	User
Description:	The query screen will show the recorded speech and will search it on the web if not canceled.
Pre-conditions:	The wake word has been detected and the query screen is listening.
PostConditions:	Spotify needs to be installed and user should be logged in if they decide to launch Spotify.
Trigger:	The user speaks his query.
Normal flow:	1. The System waits for the user's voice query. 2. The User says the query. 3. The System shows a live speech preview. 4. The User finishes speaking. 5. The System shows the final speech result and cancel option. 6. The User waits. 7. The System decides the action.
Alternative flows:	2.1 If the query includes "open Spotify", then it will run Spotify in the background. 2.2 If the query does not contain "open Spotify", then google search for the query is performed and the result is displayed.
Extensions:	7.1 If the user presses the "cancel Query" button, then the system jumps to step 1.
Exceptions:	
Notes and Issues:	Return to query screen pressing the back button to make subsequent queries.

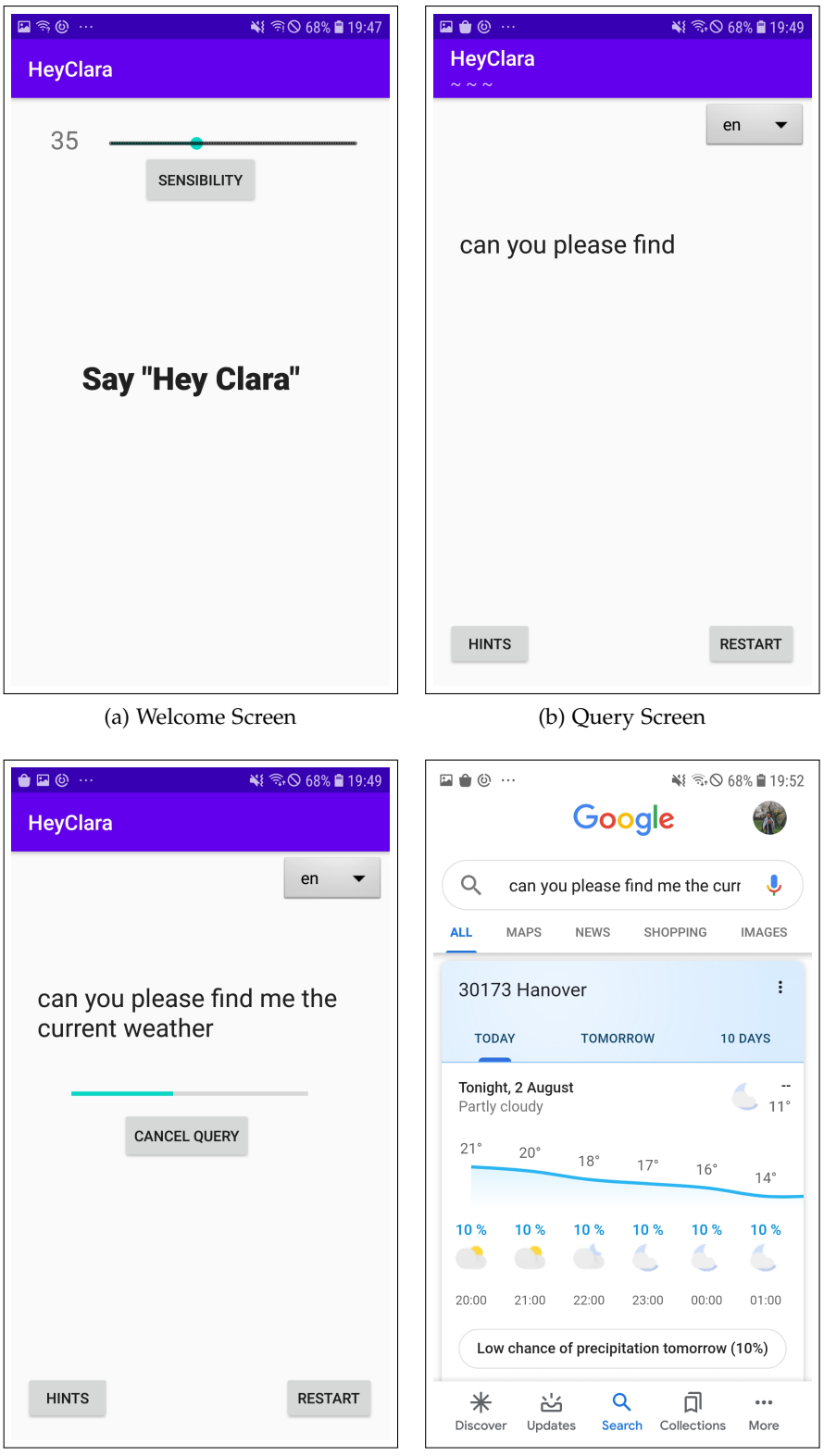
Table 4: Use Case 4: Search a Query.

Use Case ID:	5
Use Case Name:	Launch Spotify.
Actors:	User
Description:	The app will launch Spotify in background and goes itself in background and listens to the playback commands.
Pre-conditions:	The query screen has recorded the open Spotify command.
PostConditions:	Spotify needs to be installed and user should be logged in if they decide to launch Spotify.
Trigger:	The user includes "open Spotify" in his request.
Normal flow:	1. The User include "open Spotify" in a request. 2. The System correctly records and shows the query in live screen preview. 3. The User waits and does not cancel the query. 4. The System lunches the Spotify app and itself in background.
Alternative flows:	
Extensions:	3.1 If the user presses the “cancel Query” button, then the system jumps to step 1.
Exceptions:	
Notes and Issues:	Both of the app runs in background and HeyClara listens for the playback commands.

Table 5: Use Case 5: Launch Spotify.

Use Case ID	6
Use Case Name	Control Playback of SPotify
Actors:	User
Description	The Spotify app is playing music on the background via LaunchSpotify activity. It listens for the playback commands “stop”, “next”, “previous”, “pause”, “resume”. If any of the commands is detected, respective action is performed.
Pre-conditions	1. The user requested “open Spotify” on the voice query. 2. The Spotify app is playing in the background via LaunchSpotify activity.
PostConditions:	Spotify and HeyClara are running on the background where LaunchSpotify is the active fragment.
Trigger:	The user gives any of the commands mentioned in description.
Normal flow:	1. The System listens to the commands. 2. The user gives any of the commands that include keywords mentioned in the description. 3. The System decodes and performs the respective actions.
Alternative flows:	3.1 If the System fails to recognize, then it goes back to step 1.
Extensions:	3.1 If the system recognize the command, then it performs the respective action and goes back to step 1.
Exceptions:	
Notes and Issues:	The app will be listening to the commands while running in the background.

Table 6: Use Case 6: Control playback of Spotify.



(a) Welcome Screen

(b) Query Screen

(c) Final Speech

(d) Result

Figure 16: Screen Shot: Final design.

5

EVALUATION

After the completion of the final design and implementation, it was time for the user study. Using a qualitative method in-depth interview was done amongst the users after using the app to explore the experiences. As the main objective of the study was to find out the user experience in a given context avoiding the wake words repetition, the qualitative method would support the most open-ended questions about their experience. In-depth interviews would then generate rich and explanatory answers, which was the reason this method was chosen. There were 11 participants in this study. All of the participants were either a Bachelor or a Masters Student and had experience with voice assistants or smart speakers.

5.1 USER STUDY REQUIREMENTS

The users with an android device and prior experience to a voice assistant were allowed to participate in the user study. A pre-user-study survey questionnaire was prepared. All the participants were required to fill the provided form with their full name, age, sex. They were also required to name the voice assistant or smart speaker they had experience with and the information on how often they used it. Furthermore, each of them read the consent and agreed on the terms and conditions that they do not have any problem sending the data collected by the app to the moderator of this study.

All users were contacted via WhatsApp video call and made sure that they understood the terms and conditions. They were also instructed to watch the training video before installing the app and using it. Afterward, they were asked to fill the forms, and the training video and *heyClara.apk* were sent to perform the installation.

5.2 TRAINING

The three minutes and forty seconds long training video showed the users how to interact with the app. The goal was to make users familiar with the welcome screen, query screen, and show them how to perform web search and control the Spotify playback. The demo video demonstrated how to:

- Activate the app

- Perform a query and find its result
- Perform subsequent queries
- Cancel a query
- Play Spotify on background
- Use different commands like “stop”, “resume”

All the users were also requested to contact in case of any problems regarding app functions. The idea of sending the training video to the users was to minimize the learning time.

5.3 QUALITATIVE METHOD FOR USER STUDY

The users were asked to use the app for at least one hour within a week. They were free to choose the context of using the app, like studying or traveling. The app could listen to queries from the phone’s microphone and earphones for the web search context, but to achieve better response for controlling Spotify, they were highly recommended to use earphones. Since the app will be listening to the whole time, it could even listen to the music being played and sometimes react to it. The offline speech recognition library used was also not trained to recognize the background music and ignore it. Thus, the use of earphones was recommended.

After a week, all the users were contacted again via WhatsApp video call. So, it was a Tele-one-on-one interview with a structured set of questions. Based on the prepared questions (see Appendix) regarding WebSearch, Spotify Playback, and overall experience, they were interviewed for 20 to 40 minutes each. The decision to interview rather than just letting them fill out the form on their own was to get some exploratory answers to the questions. This way, the questions could also be formulated in many ways to make sure the user understood it. Besides, rather than just asking to rate and quantify the experience, the follow-up questions could also be placed to get clear answers. The rate of response will be higher, and consequently, the answers given would be of more depth. The answers to the close-ended questions were filled up in a pre-prepared form. Additionally, follow-up questions and answers during the interview were noted down as quotes or keynotes.

Furthermore, the app would write the queries asked by the users and the commands given to Spotify in a *log.txt* file in the internal storage of the user’s phone. The file could then be analyzed to find out different characteristics of the study; for example, the recognition accuracy rate could be calculated by analyzing the canceled number of queries to the total number of queries.

5.4 RESULTS

5.4.1 *Users Information*

As mentioned above, there were 11 users for this study within an age range of 18-35 years. Three of them were female, while the rest of the 8 were male. All of the participants had used at least one of the voice assistants or smart speakers in the past. The Figure 17 shows details about which ones they used. Almost

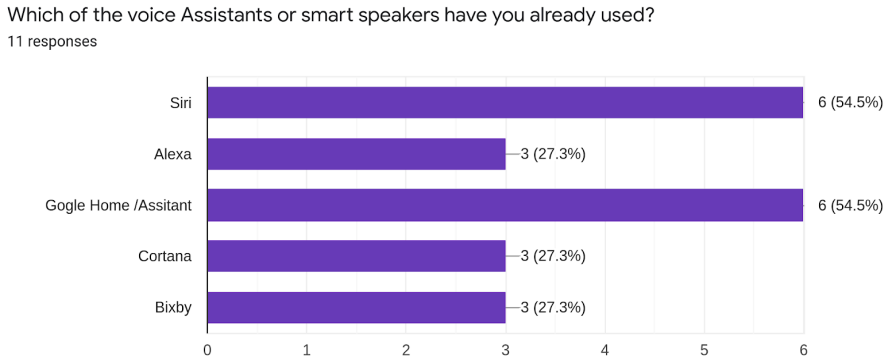


Figure 17: Voice Assistant or Smart Speakers used by user in the past

half of them had used Siri, and Google Assistant or Google Home and the other half were familiar with either Cortana, Alexa, or Bixby. The users were already familiar with the term "Wake-Words." They knew voice assistants in general. However, only one of them is a frequent user while rest of them said to use IPAs infrequently.

5.4.2 *Device and Android Versions used*

The Table 7 shows the different versions of android and smartphones used by the users while testing the app.

Almost all of them had newer versions of the android app, while one of the users having version 5.1.1 was the lowest version used. The minimum requirement for the app was android version 5.0, and to have at least one user slightly above it did help to ensure that it can also run in older versions. The app was successfully installed in all of the users device.

Sr. No	Device	Android
1.	Samsung Note 10	Android v. 10
2.	Ule Note 7	Android v. 9
3.	Samsung Galaxy S7	Android v. 8
4.	Umidigi A5 pro	Android v. 9
5.	Sony Xperia Z	Android v. 5.1.1
6.	Samsung	Android v. 9
7.	Samsung S7 H	Android v. 8
8.	One Plus 6	Android v. 10
9.	Samsung S9	Android v. 10
10.	LG G6	Android v. 9
11.	MotoG 5s Plus	Android v. 8.1.0

Table 7: Device and Android Versions used

5.4.3 *Qualitative findings (Web Search Context)*

5.4.3.1 *Recognition of wake words:*

The users were asked about the experience or problems they had while activating the app via wake words. Almost all of them agreed that the recognition was quite accurate and easy. The Figure 18 below illustrates their response.

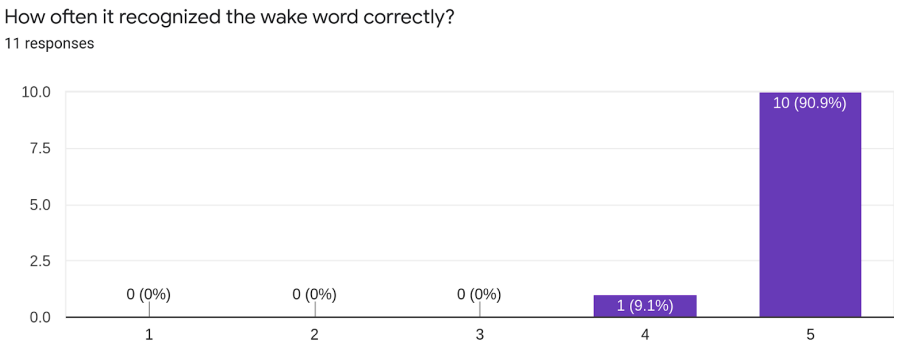


Figure 18: Recognition of Wake Word (1 being rarely, 5 being almost every time)

The users were quite satisfied with the wake word detection. The option to set the sensibility could have impacted the satisfaction rate. However, some of the participants also suggested that they did not use the sensibility function and thought it was unnecessary. In contrast, one of the participants favored the option of setting it to the value that suits the individual.

5.4.3.2 *Not to repeat wake words:*

All participants said that it was a better experience using voice assistants without repeating the wake word. Some of them even said it was easier not to be compelled to repeat the key phrase. However, some participants needed some time to adapt to avoiding wake words.

"I said "Hey Clara" in my first few searches even when I knew it was not necessary. It was because I was used to saying, "Hey Google" quite often." [-Q1]

"At first I said Hey Clara and followed immediately with a request. Then I remembered that I have to wait before she activates, and only after that, I can put in a request. It took me a few searches to realize the difference in wake word usage." [-Q2]

Participants notice the change in the pattern of the interaction, and some of them compared their experience with the one they had.

"It was easier if one wanted to search quite often but also with short breaks." [-Q3]

"It was not that of a difference while searching a few times, but definitely a better experience for the long run." [-Q4]

To summarize, participants liked the feature not to repeat wake words in general but depended mainly on the duration of interaction with IPAs. They seem to be okay with wake words if the interaction is just for one or two requests but wanted to avoid it in the long term. The continuous recognition for long-term interaction in the app allowed them not to hurry to follow up with other questions so that they could search their queries with ease and as needed.

5.4.3.3 *Query recognition and live speech preview:*

The users were also quite satisfied with the app recognizing their queries. The Figure 19 shows the response to the question of how often it correctly recognized the user's queries.

Eight participants said the recognition was correct almost every time, while three of the participants also encountered some problems in recognition. To the follow-up questions asking the possible reason behind incorrect recognition, one of the users stated background noise could have affected it, while the other user also blamed his English skills. However, they were amazed at how good it was in understanding the informal spoken language.

"Clara even understood the friendly spoken language. I asked her as I was asking my friend." [-Q5]

Furthermore, all of them agreed that the live speech preview on the screen was quite useful. It helped to confirm if the recognized query is the desired one. Some participants even took it as helpful feedback as they could see that it was

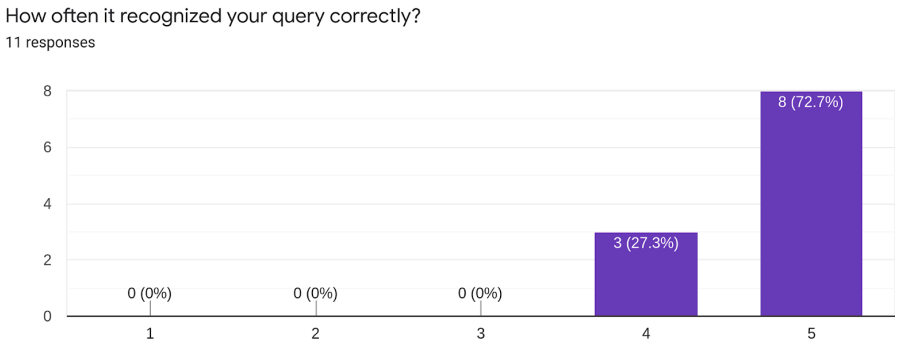


Figure 19: Query recognition (1 being rarely, 5 being almost every time)

listening to what they were saying. As Figure 9 suggested that 65% of the users like to have a smart speaker with a screen, the same result could be seen here. All the participants favored the visual feedback of their request. However, it is essential to differentiate the context; for web search, it could be important but may not be for other contexts.

5.4.3.4 *Option to cancel query:*

Some of the users explicitly mentioned this feature as one of the most delightful features to have. One of them even complained about Siri lacking this feature.

"I hate when Siri does not understand me and replies with, "Sorry I don't have an answer for that". So, it's better to cancel the request if falsely recognized." [- Q6]

"It's convenient for the users who interact in a language other than their mother tongue. As long as the accent problem exists, the feature to cancel the query should be required." [- Q7]

All the participants highly agreed that the option to cancel the query was beneficial. Similarly, the user also mentioned the time given (4 Seconds) to cancel the query was enough to check for correctness. They also favored the short delay for correct results.

5.4.3.5 *Continuous listening:*

None of the users were concerned about the continuous listening feature of the app. While reminding them about privacy, some of them replied:

"I searched for a few queries in a row, and as soon as I was done with it, I closed the app. There were short breaks in between the queries but no longer than a minute. So, listening was not a concern to me." [- Q8]

"I searched something and went back to the screen; after that, I talked with my mom, it was still listening. Since I spoke in another language than Clara could understand, she searched random words it interpreted in google. After that, I left the screen active only when I wanted to search for something; otherwise, I just closed it." [- Q9]

The users did not seem to have privacy problems regarding continuous listening. However, if other people were talking in the surroundings, it could react to it, which might present one other random search result.

5.4.4 *Qualitative findings (Playback Context)*

5.4.4.1 *No wake words at all:*

All the users liked the feature to be able to control Spotify with voice commands. Each of them agreed that freedom, not even to say the wake word was significant. When asked to compare the experience with other assistants, all the users who had similar experience mentioned that the app was highly efficient.

"To listen and control playback options, Clara is way more efficient. However, other smart speakers are good at searching for the song." [- Q10]

"I listen to music while studying. So, to give commands now and then, Clara has advantages." [- Q11]

Although the participants did like the freedom to avoid wake words while controlling the playback, all of them wished for more features.

5.4.4.2 *Running on background:*

Similarly, all the users liked that the app could run in the background and still listen to the commands. Furthermore, they were also satisfied with the battery usage as it was running in the background and said they did not notice any significant change. One of them even experimented with playing Spotify on a different device than their phone and controlling it.

"I switched the device Spotify was playing on, from mobile to my PC, and controlled it via voice. I really enjoyed it." [- Q12]

The participants liked the ability of the app being on the background and remaining in context to listen to the commands.

5.4.4.3 *Freedom to formulate the commands:*

As the users were made aware of the options to formulate the command sentence on their own, they claimed to have tried it. They even tried with different variations in personality.

"I formulated my commands as per mood. Sometimes politely and sometimes even angrily." [- Q13]

"Whenever she did not listen to me the first time, I formulated my sentence differently. The precision was higher when I said a longer phrase than just the command word." [- Q14]

Namely, all of them liked not being bound to a specific syntax structure and having the freedom to say the commands as they wanted. Some of them even claimed to have had better accuracy whenever they included the command words in their sentence rather than just uttering them alone.

5.4.4.4 *Contexts:*

Users listened to music in different contexts as they wanted (Figure 20). Some of the most used contexts were while studying and cooking. One of the users had a problem with Spotify and did not use it in any context (– in fig below) at all. They also explained some challenges and problems they faced as per contexts.

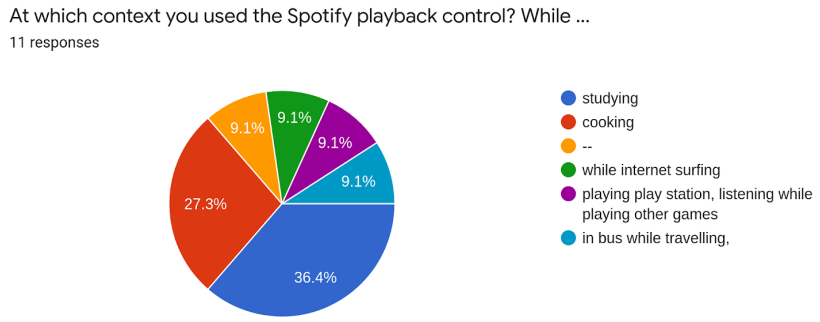


Figure 20: Contexts where the app was used.

The majority of them encountered some underlying recognition problems.

"I liked that I did not have to worry about touching my phone with wet hands as I listened to it while cooking. However, It did not understand me quite often, and I had to repeat the commands once or twice or even for the third time. Maybe it was because of the background noise." [- Q15]

"I used it while playing FIFA at PlayStation. Sometimes she reacted to my in-game reaction." [- Q16]

The other user said to have used it while traveling through the bus on the way to work and back home. To the follow-up question, whether the user was concerned about giving voice commands at public places and if it were recognized correctly. User replied :

"Ach! There were not many people, and Clara could listen to my normal voice as I was on earphones. I did repeat the commands on a few occasions, but was not that bad at all." [- Q17]

The recognition did seem to be affected by background noises. Similarly, there were some false triggers, too, making it less accurate in some scenarios. Users shared cases where one should repeat the commands or reformulate it to have better precision. However, the use of earphones and a less noisy environment seems to increase the recognition of the commands given.

5.4.4.5 *Problems encountered:*

Apart from the one user who did not use Spotify at all, all of the other users reported problems regarding recognition. They also mentioned that the recognition was not the same as it was while searching the web. Different factors like the quality of the microphone of the user device or earphones, the accent, the background noise, and the quality of the recognition used in the app all should have played the role. Furthermore, it signifies the difference in power between Google's speech recognition and the Pocketsphinx library. They slightly rated the recognition a little lower than in the web search context (Figure 21).

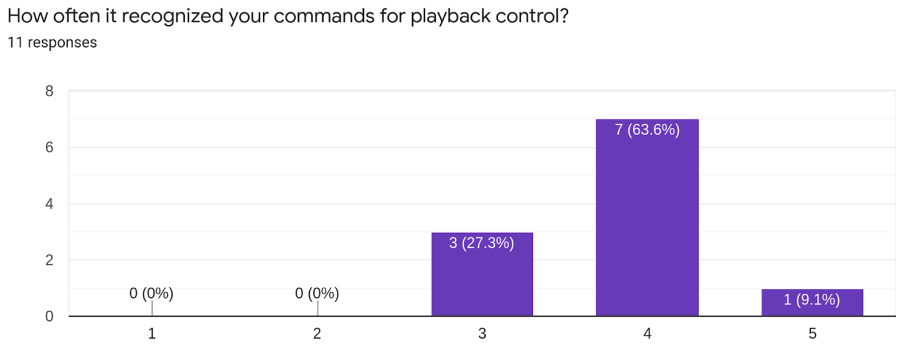


Figure 21: Recognition of playback commands (1 rarely, 5 almost every time)

They were also asked about the specific commands with which they had problems. The majority (6 participants) of them said they had a problem with "Next" (Figure 22). Some of them even thought that their free version of Spotify could be the reason behind the limited usage of the "next" option.

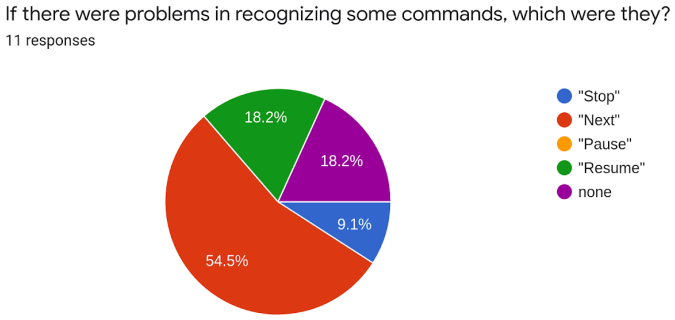


Figure 22: Problem with some playback commands.

However, the inconsistent nature of the recognition makes it difficult to point in a particular direction for this problem. The user's accent, the loudness, and the state of the recognizer and the quality of the device used are all the variables that could play a significant role in one particular case. Despite some problems, the users did adapt and used the freedom to formulate the commands to get better results.

5.4.4.6 *Additional playback control they wish it had:*

When asked about additional functionalities, they wish Clara had almost all replied with "Song on demand." All of the participants stated that it would be great if Clara could have functionalities to control the whole Spotify. Searching songs, bands, and saving to their playlist were some of their wishes.

"It would be great if I could search for the bands and play their songs." [- Q18]

"I would love not only to open Spotify but also to close it via Clara." [- Q19]

"I will definitely love to search the songs of my wish and save it on my playlist." [- Q20]

Although every participant liked the feature to control Spotify with voice commands and without wake words, they wished that the app had more control than the present. It states that the users will likely be inclined towards the apps or devices where they had freedom from wake words and could have more control over the music apps.

5.4.5 *Qualitative findings (Overall App Experience)*

5.4.5.1 *Features liked by users:*

There were a variety of responses to the question which features you liked in the app. Each of them appreciated different features. Some of them mentioned the non-repetition of wake words, while others mentioned Spotify's voice control.

"To not repeat "Hey Clara", playing Spotify in the background and still controlling it, and the live speech preview and the option to cancel. So, I liked it all." [- Q20]

"I liked the control over Spotify with my voice and mostly used it while cooking." [- Q21]

"Spotify control was a nice feature. I even used it on my PC via phone." [- Q22]

While most of them liked the feature to control Spotify, some of them also appreciated the minimalist approach of the app and recognition ability during a web search.

"I liked the minimalist nature of the app. It was quite easy to use. " [- Q23]

"I really liked the accuracy of voice recognition. It was really easy to search on the web as I don't like to tip that much." [- Q24]

Some participants thought the live speech preview and cancel option were good. While one of them mentioned that the feature to set sensibility was nice to have.

"Web search was good, and to cancel the query if misrecognized was a better option." [- Q25]

"I liked the function to set the sensibility of wake words and was amazed that she could understand even the friendly language during a web search." [- Q26]

All of the participants did mention at least one of the features without having to think. While most of the features like controlling Spotify and voice recognition and the option to cancel the query were highly mentioned, some of them also appreciated the ease of usability.

5.4.5.2 *Features that were not good enough:*

Participants were also encouraged to speak about the features they did not like freely. Some of them mentioned that the sensibility option was useless, while others complained about the background's brightness.

"The background was too bright. I would have loved to have a night mode theme." [- Q27]

"The logo and background could have been made better." [- Q28]

"Sensibility was not required. Setting it to around 50 as default value would have done." [- Q29]

The participants did want to have a subtle logo and a better background. While the usability of the app was appreciated, the design has a place to improve. Furthermore, some of the participants also mentioned that the controls of Spotify playback could have been better, whereas one of the users criticized the app's less integration option.

"Spotify controls were not good enough as web search. I had to repeat the commands." [- Q30]

"Why just Spotify and why not youtube and other apps?" [- Q31]

That said, there are many rooms for the improvement of the app regarding the integration with other apps and recognition of control commands. The overall design and the background needs to be polished as well. The feedback reflected the area of focus and the area where it lacked.

5.4.5.3 *Expectation for the improved version:*

Most of the expectations of the participants overlapped about the same point demanding the integration with other apps. They wished the app could control other apps just like Spotify.

"It would be cool if the app could open other apps like Spotify and be controlled via voice. I would also like the app to remind me about the future events I set." [- Q32]

"I would like to have a call function. Maybe also the ability to control other apps ." [- Q33]

"To be able to control Facebook and youtube, just like Spotify would be a great addition." [- Q34]

Following this, some of the users wanted more playback control options with Spotify. They all desired to perform different tasks, like searching for a song or a playlist. Some of them also wanted better accuracy on the recognition of commands.

"More options over Spotify control would be great, and also with improved accuracy." [- Q35]

"I would like to search for a playlist, and it would be great if I could also control the volumes or even close Spotify." [- Q36]

"The option to search for a specific song or band would be great. It will also be good when Clara knows when to stop listening during a web search." [- Q37]

The users wanted more features to have as they all wished to control other apps, just like Spotify. Namely, they wanted all the features that a voice assistant or smart speakers in the market have. Some of them even asked whether they could continue using the app and requested an improved version if made.

5.4.5.4 *User Interface:*

Participants were then asked to rate the different features of the app. The questions find their opinion about, for example, the learning curve and interactiveness of UI. The following figures show their responses.

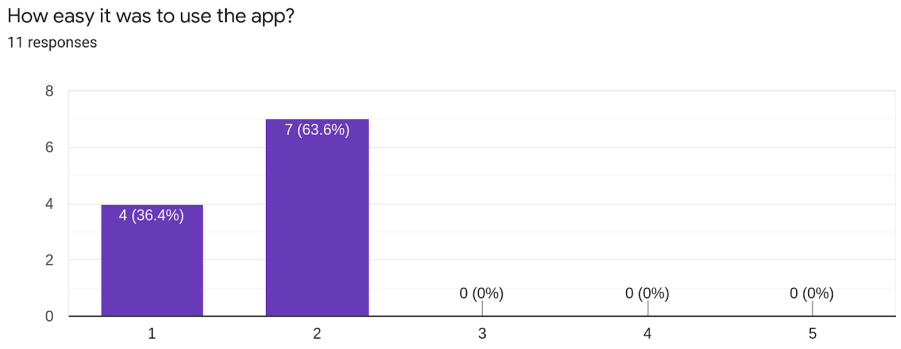


Figure 23: Easy to use? (1 very easy, 2 easy, 3 normal, 4 hard, 5 very hard)

While all of them said it was easy to use, four participants thought it was super easy. The experience they had with voice assistants in general and the training video might have helped. However, the app was also designed to be simple and to the point. Not to overwhelm users with many features and direct on the experience that was bound to be collected was the main focus. Thus, making it quite easy to use by anyone.

Participants also highly agreed that the app was interactive, and the feedback was quite easy to interpret.

The live numeric value update while setting the app's sensibility and vibration after detecting wake word were all well recognized by the users. The message "Say, Hey Clara," and the greeting after the activation was also noticed by the users correctly. The live speech preview of the recognized words was highly appreciated. The cancel button's appearance with a regress bar as soon as the recognition completed was also correctly interpreted by all of the participants. However, one of the participants did comment saying, *"The color red could have been better to signify the regression."*

Furthermore, they were also explicitly asked about the clarity in understanding the functionality of the buttons used.

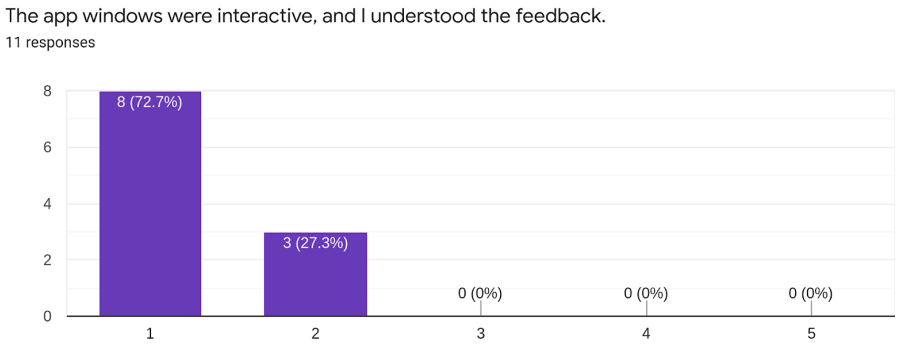


Figure 24: Interactiveness and feedback (1 highly agree, 2 agree, 3 neutral, 4 disagree, 5 highly disagree)

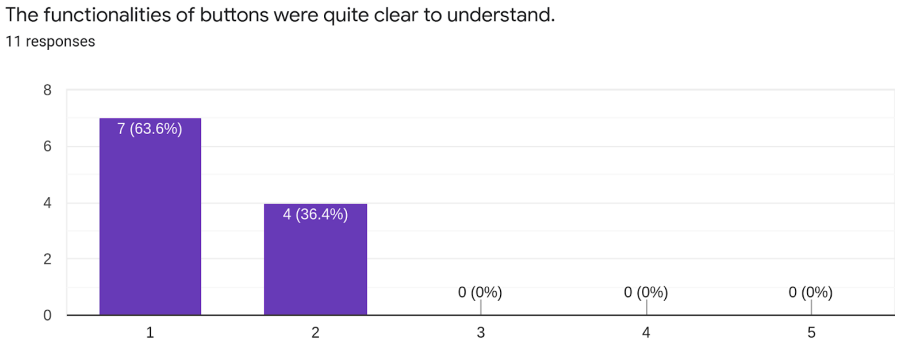


Figure 25: Functionality of buttons (1 highly agree, 2 agree, 3 neutral , 4 disagree, 5 highly disagree)

All of them agreed that they understood the purpose of the buttons used. The motive was also to minimize the use of buttons. Only the most required ones were set. Sensibility and Hints buttons were just to give information, while restart would start the recognition, and the app will greet again, giving the feedback. It could have helped the users understand the functionality by trying out once or even just by looking.

5.4.6 *Quantitative Findings*

While 3 of the participants could not find the *log.txt* file on their phone, the rest of the eight files were analyzed. Some participants were more inclined towards one context than the other. On average, there were 46 query searches and 48 playback commands given. Altogether there were then 96 queries on average including playback commands.

The number of canceled queries to the actual number of queries searched on the web is around 30%. Thus, raising a question about the accuracy of voice recognition. However, the log files show that the user did not perform the web search all the time. Sometimes they were just testing the ability of recognition by saying the name of the family members or digits and numbers probably to check the interpretation, thus resulting in more canceled queries.

Queries Performed	**No. of Times**
Web Search	276
Open Spotify	95
Total	**371**

Table 8: Queries Stats.

Action Performed	**No. of times**
Web Search Canceled	87
Open Spotify canceled	6
Total	**93**

Table 9: Canceled Queries stats:

The web search included weather updates, corona death toll and updates, celebrity search, supermarket nearby, and some song demands queries. Interestingly, there were 12 cases where the user said *"Hey Clara"* before a query search. This signifies that the users were habituated on repeating wake words.

Although users could not search and play the song of their choice, there were still requests like "Play Despacito on Spotify" or "Play song of Red hot chili Peppers". Despite this, there were 341 playback commands taken by the playback context (Table 10). Surprisingly, the most given command is "Next", although most of them reported that "Next" was not recognized easily. The user did not have direct access to their playlist. So, "Next" being used more often is logical. Unsurprisingly, "Previous" is one of the least given commands, while "Stop" and "Pause" had the same function, users favored "Stop".

Playback Commands	**No. of Times**
Next	124
Stop	98
Resume	61
Pause	25
Previous	33
Total	**341**

Table 10: Playback Commands stats.

The log files also suggested that the users were testing the ability of recognition often in the beginning. The playback commands given were quite prevalent within a short period. Most of the participants were trying to figure out precision rather than just using them casually, thus resulting in frequent commands. The average of 96 commands given in an average of one hour of usage further proves that point. The vigorous amount in short period may have influenced their experience as it will definitely be easier to say less words in that short time. Although a frequent user could give a little fewer commands, the advantage is still there without the repetition.

6

DISCUSSION

The user experience analysis suggests that both of the chosen context; web search and playing music was more efficient without repeating the wake words. The app's ability to remain in the chosen context for the desired time further ease the interaction.

The music playback context further highlighted effectiveness. The interaction and feedback while controlling the playback are instantaneous and more frequent than in web search. On the one side, the users could consequently give more commands within a short period, making the avoidance of wake words quite clear. On the other side, users could remain in a context as long as they want without repeating wake words making it more effective. Thus, the user found it to be quite efficient than the experience they had with the repetition of wake words. The flexibility to variate the commands' phrases also minimized the repetitiveness in overall, suiting the different personality approaches.

Surprisingly, the users were not that much concerned about the continuous listening as both males and females were okay with it. It was probably because of their knowledge about the field and basic understanding of what happens to their data. It did not align with [Olson 2019] but supports the mental model reasoning from this earlier study [Zeng et al. 2017]). The offline listening of wake word detection and playback control could have lessened the worries about privacy, but the online listening during web search did not make a concern. Since users could remain in the web search context as long as they need, continuous listening was not even a problem. To eliminate the risk of being passively heard, users just left the context. Similar patterns were also followed by users on this study [Malkin et al. 2019] whenever they were concerned about the listening feature. Furthermore, this way, the user got the advantages of continuous listening without repeating wake words and, at the same time, not being concerned about privacy. Thus, highlighting the advantage of a dedicated context.

Additionally, the high accuracy of natural language recognition achieved was because of Google's powerful speech-to-text API. Thus, increasing the understanding of friendly spoken language and the satisfaction rate of the users. The visual feedback of the recognition during web search followed by an option to cancel helped to search the desired query. Although the accent problem is still present in the used API, the feature to cancel misrecognized queries saved time and avoided dissatisfaction. The importance of precision is reflected here as users preferred a few seconds delay of checking before getting the search results.

The findings also suggested that users tend to interact with IPAs differently depending on the context they are using. On the one hand, the involvement is primary while performing the web search; on the other hand, the involvement is secondary while listening to music. Thus, the hands-free nature of the interaction was needed during secondary involvement as the users were busy performing other tasks like studying, cooking, playing video games. Similar results were also reported by this study [Cowan et al. 2017] stating the interruption being a significant barrier in most contexts.

However, the duration of time spent in a particular context profoundly influenced the user experience. The avoidance of wake words while performing a web search was significant only if the user used it for the subsequent queries. The number of back to back queries performed determined the difference in the experience of minimized wake word usage. Thus, the more the subsequent queries, the more efficient was the experience, the more noticeable was the difference. The threat is that the less use of wake words would mean less authority over the conversation with IPAs, less projection of emotion, thus losing a contextual meaning, as mentioned in this study earlier [Jung and Kim 2019].

To summarize, the dedicated mode for both of the contexts used in this study helped the user to perform the task more efficiently in the longer run. The privacy problems regarding continuous recognition [Edu et al. 2019; Olson 2019], are minimized to the lowest as most of the continuous listening would be offline and thus local, while the online recognition could be toggled as per need hindering casual listening.

7

CONCLUSION AND FUTURE WORK

The results of this study definitely answer "Yes" to whether avoiding wake words ease the interaction with IPAs in some contexts. It certainly demonstrates the effectiveness and advantages of using IPAs in a dedicated mode. As it suggests, the longer the interaction with IPAs, the more efficient it becomes without the repetition of key-phrase.

Furthermore, the study finds that the context dictates the demand for hands-free features depending upon whether the interaction is primary or secondary. Sometimes users prefer visual feedback (web search context) while other times, it could be interpreted as an interruption. While continuous recognition does raise privacy questions, the dedicated mode and offline listening seem to minimize the gulf of mistrust between users and the IPAs.

As stated earlier, the users demanded more integration with other apps, and the future work is wide open. Depending on the apps and the nature of integration, further work can be deployed. The authentication of the user could also let the other playback controls in Spotify. Further, the implementation of machine learning and training with audio files will improve the accuracy of playback control, taking the work to the next level.

The interface design has lots of room for improvement, and the background and logo could be standardized. The continuous listening feature can be optimized with a time limit. The users could also be personalized with essential information, which could be required if the app integrates with, for example, contacts or other messaging apps.

As the study was performed within 11 users, the numbers may not be sufficient to generalize the results. The features provided in the app were a small representation of what regular smart speakers or voice assistants do. So, for the generalization, the study could be performed within more users and more features for a longer time. Although the number of users may not always influence the results, their motivation does. Further, the character of the users and their interest in the context plays a huge part too.

To conclude, this study shed some light on the overall experience of voice assistants or smart speakers regarding wake word avoidance. The minimalist nature of the app with a few features did allow to have a piece of explorative information on the user experience. The context-based approach with just one utterance of wake word allowed the user to operate for the desired amount of time. Further, it shows the contexts where wake words could be wholly unnecessary, and

the interaction could be far more natural and less mouthful. Thus, simplifying the wake word usage helps to achieve seamless interaction with CUIs in some contexts.

BIBLIOGRAPHY

Noura Abdi, Kopo Ramokapane, and Jose Such. More than smart speakers: Security and privacy perceptions of smart home personal assistants. 06 2019.

Shashank Ahire and Michael Rohs. Tired of wake words? moving towards seamless conversations with intelligent personal assistants. In *Proceedings of the 2nd Conference on Conversational User Interfaces*, CUI '20, New York, NY, USA, 2020. Association for Computing Machinery. ISBN 9781450375443. doi: 10.1145/3405755.3406141. URL https://doi.org/10.1145/3405755.3406141.

Inc Amazon Technologies. Dynamic Wake Word detection, December 2017. https://patents.justia.com/patent/10510340#citations.

Tawfiq Ammari, Jofish Kaye, Janice Tsai, and Frank Bentley. Music, search, and iot: How people (really) use voice assistants. *ACM Transactions on Computer-Human Interaction*, 26:1–28, 04 2019. doi: 10.1145/3311956.

Inc Apple. Personalized Hey Siri, April 2018. https://machinelearning.apple.com/research/personalized-hey-siri.

A Berdasco, G Lopez, I Diaz, L Quesada, and Guerrero LA. Experience Comparison of Intelligent Personal Assistants: Alexa, Google Assistant, Siri and Cortana, 2019. URL https://www.mdpi.com/2504-3900/31/1/51.

Michael R. Braun, Anja Mainz, Ronee Chadowitz, Bastian Pfleging, and Florian Alt. At your service: Designing voice assistant personalities to improve automotive user interfaces. *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, 2019.

Wueest Candid. A guide to the security of voice-activated smart speakers, November 2017. https://docs.broadcom.com/doc/istr-security-voice-activated-smart-speakers-en.

Inc CBInsights. How Big Tech Is Battling To Own The $49B Voice Market, February 2019. https://www.cbinsights.com/research/facebook-amazon-microsoft-google-apple-voice/.

Benjamin Cowan, Nadia Pantidi, David Coyle, Kellie Morrissey, Peter Clarke, Sara Al-Shehri, David Earley, and Natasha Bandeira. "what can i help you with?": infrequent users' experiences of intelligent personal assistants. pages 1–12, 09 2017. ISBN 978-1-4503-5075-4. doi: 10.1145/3098279.3098539.

Anthony Cuthberstson. Google defends listening to private conversations on google home – but what intimate moments are recorded? URL https://www.independent.co.uk/life-style/gadgets-and-tech/news/

google-home-recordings-listen-privacy-amazon-alexa-hack-a9002096.html.

Daniel J. Dubois, Roman Kolcun, Anna Maria Mandalari, Muhammad Talha Paracha, David Choffnes, and Hamed Haddadi. When Speakers Are All Ears: Characterizing Misactivations of IoT Smart Speakers. In *Proc. of the Privacy Enhancing Technologies Symposium (PETS)*, 2020.

Jide S. Edu, Jose M. Such, and Guillermo Suarez-Tangil. Smart home personal assistants: A security and privacy review, 2019.

Eoghan Furey and Juanita Blue. Alexa, emotion, privacy and gdpr. pages 1–5, 01 2018. doi: 10.14236/ewic/HCI2018.212.

Consulting Futuresource. Virtual Assistant to Exceed 2.5 Billion Shipments in 2023, December 2019. https://futuresource-consulting.com/press-release/consumer-electronics-press/virtual-assistants-to-exceed-25-billion-shipments-in-2023/.

Scott Huffman. The Future Of Google Assistant, May 2018. URL https://blog.google/products/assistant/io18/.

D. Huggins-Daines, M. Kumar, A. Chan, A. W. Black, M. Ravishankar, and A. I. Rudnicky. Pocketsphinx: A free, real-time continuous speech recognition system for hand-held devices. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 1, pages I–I, 2006.

Mansoor Iqbal. Spotify usage and revenue statistics, July 2020. https://www.businessofapps.com/data/spotify-statistics/#1.

Hyunhoon Jung and Hyeji Kim. Finding contextual meaning of the wake word. pages 1–3, 08 2019. ISBN 978-1-4503-7187-2. doi: 10.1145/3342775.3342805.

Donald E. Knuth. Computer Programming as an Art. *Communications of the ACM*, 17(12):667–673, December 1974.

Veton Këpuska. *Wake-Up-Word Speech Recognition*. 06 2011. ISBN 978-953-307-996-7. doi: 10.5772/16242.

Abner Li. Hey google sensibility now official, April 2020. https://9to5google.com/2020/04/23/hey-google-sensitivity/.

Ewa Luger and Abigail Sellen. "like having a really bad pa": The gulf between user expectation and experience of conversational agents. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI '16, page 5286–5297, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450333627. doi: 10.1145/2858036.2858288. URL https://doi.org/10.1145/2858036.2858288.

Nathan Malkin, Joe Deatrick, Allen Tong, Primal Wijesekera, Serge Egelman, and David Wagner. Privacy attitudes of smart speaker users. *Proceedings on Privacy Enhancing Technologies*, 2019:250–271, 10 2019. doi: 10.2478/popets-2019-0068.

Michael McTear, Zoraida Callejas, and David Girol Barres. *The Conversational Interface*. Springer International Publishing, 1st edition, 2016.

NatioalPublicMedia. The smart audio report, April 2020. URL https://www.nationalpublicmedia.com/insights/reports/smart-audio-report/#download.

Christi Olson. New report tackles tough questions on voice and ai, April 2019. URL https://about.ads.microsoft.com/en-us/blog/post/april-2019/new-report-tackles-tough-questions-on-voice-and-ai.

Wolf Paulus. Custom wakeup-word for an android app, October 2017. https://wolfpaulus.com/custom-wakeup-words-for-android/.

Martin Porcheron, Joel E. Fischer, Stuart Reeves, and Sarah Sharples. Voice interfaces in everyday life. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, page 1–12, New York, NY, USA, 2018. Association for Computing Machinery. ISBN 9781450356206. doi: 10.1145/3173574.3174214. URL https://doi.org/10.1145/3173574.3174214.

Zeynab Raeesy, Kellen Gillespie, Zhenpei Yang, Chengyuan Ma, Thomas Drugman, Jiacheng Gu, Roland Maas, Ariya Rastrow, and Björn Hoffmeister. Lstm-based whisper detection, 2018.

Mi-Suk Seo and Irene Koshik. A conversation analytic study of gestures that engender repair in esl conversational tutoring. *Journal of Pragmatics*, 42(8):2219–2239, 2010. ISSN 0378-2166. doi: 10.1016/j.pragma.2010.01.021. URL http://www.sciencedirect.com/science/article/pii/S0378216610000342. Face in Interaction.

Chao Shi, Michihiro Shimada, Takayuki Kanda, Hiroshi Ishiguro, and Norihiro Hagita. Spatial formation model for initiating conversation. 06 2011. doi: 10.15607/RSS.2011.VII.039.

Pranay DigheSaurabh AdyaNuoyu LiSrikanth VishnubhotlaDevang NaikAdithya SagarYing MaStephen PulmanJason D. Williams. Lattice-based improvements for voice triggering using graph neural networks. 2020. URL https://arxiv.org/pdf/2001.10822.pdf.

Eric Zeng, Shrirang Mare, and Franziska Roesner. End user security and privacy concerns with smart homes. In *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*, pages 65–80, Santa Clara, CA, July 2017. USENIX Association. ISBN 978-1-931971-39-3. URL https://www.usenix.org/conference/soups2017/technical-sessions/presentation/zeng.

A

APPENDIX

Here are the list of questionnaire used during survey and interviews.

A.1 SURVEY TO FIND CONTEXT

The follwing questions were asked to find out the most performed tasks along with some general problems with IPAs.

1. Age? ->AgeGroup(18-25, 25-35, 35-45, 45+)
2. Have you ever used voice assistant or smart speakers?
3. How often?
4. Which ones? -> Siri, Alexa, Google Home/Assitant ...
5. For what puropose? -> see for options**??**
6. Would you like to customize the wake words(hey Siri, ok google or Alexa) and have your own choice set to [your_fav_name]?
7. If yes, what sort of names?
8. Do you like the fact, that you need to say wake words every time before pleading a request to your voice assistant?
9. Would you like it, if your voice assistant could listen to you for a certain period of time, without you having to wake it up again and again?
10. Is there any other problem you have with your Voice Assitant?

A.2 USER STUDY QUESTIONNAIRE

A.2.1 *Consent*

Participation in this usability study is voluntary. All information will remain strictly confidential. The descriptions and findings may be used to help improve the application. However, at no time will your name or any other identification be used.

- I've read the terms and I agree to send the data collected by the app to the moderator of this study.

A.2.2 *Pre-study questionnaire*

1. lastname, firstname
2. Sex
3. Age Range
4. Name of Voice Assistants or Smart Speaker used.
5. How often?

A.2.3 *Post Study Questionnaire*

The questions were both open-ended and closed-ended. The follow up questions were made accordingly along with the conversation.

1. Device Used
2. How often it recognized Wake words correctly?
   -> 1 rarely, 5 almost every time
3. How was the experience of not repeating wake words during web search?
4. How often it recognized the query correctly?
   -> 1 rarely, 5 almost every time
5. Did the live speech preview of your query on the screen was helpful?
   -> Yes, No
6. Were you concerned about the continuous listening while performing a web search?
   ->Yes, No

7. How often it recognized playback Control commands?
   -> 1 rarely, 5 almost every time
8. If there were problems in recognizing some commands, which were they?
9. At which context you used the Spotify playback control? While ...
10. Did you like the freedom to place the control commands anywhere in your request?
11. Compare to the voice assistant or smart speaker you used, did you like not repeating wake words while controlling playback of Spotify("Hey Clara" in this case) ?
12. Any other playback control you wish it had?
13. How easy it was to use the app?
    -> 1 very easy, 5 very hard
14. The app windows were interactive, and I understood the feedback.
    -> 1 highly agree, 5 highly disagree
15. The functionalities of buttons were quite clear to understand.
    -> 1 highly agree, 5 highly disagree
16. What features you liked about the app?
17. What features you think were not good enough?
18. What do you expect in the improved version of this app?