What Can I say? Addressing User Experience Challenges of a Mobile Voice User Interface for Accessibility

Eric Corbett
Google Research
Mountain View, CA 94043
eorbett@gatech.edu

Astrid Weber
Google Research
Mountain View, CA 94043
aweber@google.com

ABSTRACT

Voice interactions on mobile phones are most often used to augment or supplement touch based interactions for users' convenience. However, for people with limited hand dexterity caused by various forms of motor-impairments voice interactions can have a significant impact and in some cases even enable independent interaction with a mobile device for the first time. For these users, a Mobile Voice User Interface (M-VUI), which allows for completely hands-free, voice only interaction would provide a high level of accessibility and independence. Implementing such a system requires research to address long standing usability challenges introduced by voice interactions that negatively affect user experience due to difficulty learning and discovering voice commands.

In this paper we address these concerns reporting on research conducted to improve the visibility and learnability of voice commands of a M-VUI application being developed on the Android platform. Our research confirmed long standing challenges with voice interactions while exploring several methods for improving the onboarding and learning experience. Based on our findings we offer a set of implications for the design of M-VUIs.

Author Keywords

Accessibility; Voice user interfaces; Universal voice control

ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous

INTRODUCTION

The use of voice as a control modality in mobile HCI is ubiquitous and growing. For instance, most major mobile operating systems provide information access via voice commands to smart assistants such as Google Now, Siri, and Cortana. Also, text entry via voice dictation has become a standard feature for smart phones. These voice features provide alternate control modalities for the on-the-go context of mobile HCI which often times introduces situational impairments such as limited visual attention while driving or limited hand availability during certain activities. In these scenarios voice interactions can be more convenient to users than physical interactions; however, general best practices of HCI and interaction design advise against voice as a primary interaction modality as physical interactions are often considered most efficient [3,13]. On the hand, this dynamic is reversed for people with limited hand dexterity. For these users, voice as a primary control modality can be the most efficient form of interaction or in extreme cases the only viable method of interaction [11]. As such, accessibility HCI researchers have explored the use of *Voice User Interfaces* (VUIs) as the primary control modality. This has been done in a variety of contexts. For instance, Sears [9] developed a speech-based cursor for control and navigation of desktop computer interfaces, Harada [17] developed a system "VoiceDraw" that allows for voice driven freeform drawings on a computer while Soronen et al [27] looked at speech control of a television guide interface.

While accessibility research has made significant progress with VUIs in the area of desktop computing, mobile HCI has presents new challenges and use cases that have not been solved yet. Furthermore, while some of the research done on desktop VUIs can be generalized to mobile a great deal of it is simply not directly applicable. In general, mobile device HCI offers its own unique challenges for interaction design [29] and accessibility [20] that requires research and design to be specific to these conditions. Recently Zhang et al made significant progress for accessibility mobile HCI introducing the JustSpeak application [33]. JustSpeak is a *Mobile Voice User Interface* (M-VUI) application that allows for complete voice control over Android devices. JustSpeak provides an unprecedented level of access and control for users with limited hand dexterity by allowing voice interactions to replace touch gestures to perform common mobile device activities such as launching applications, navigation, composing and sending messages, and making phone calls.

Despite the high level of accessibility that a M-VUI application like JustSpeak makes possible for people with limited hand dexterity it is in some sense a double edged sword. For one, the use of voice interaction as the primary

MobileHCI '16, September 06 - 09, 2016, Florence, Italy

ACM 978-1-4503-4408-1/16/09…$15.00
DOI: http://dx.doi.org/10.1145/2935334.2935386

control modality for navigation on mobile devices is still novel and unfamiliar to users. In addition to that there are long standing issues of discoverability and learnability with voice interactions in general [14]. Discoverability of voice commands is a fundamental challenge in any application that uses voice as an input modality [21,32]. Users will often have inconsistent and inaccurate mental models of what can and can not be said to interact with a system. The ephemerality of voice as an input method does not allow for the same level of learning and discoverability of direct manipulation interactions which are aided by affordances and metaphors to guide users [1, 26]. For instance, in [25] [27] Sears showed that improper voice command use was a significant usability concern in a voice dictation application as users would struggle to discover and learn commands. Feng et al [12] found in a similar study that even after exposing users to lists of command users would forget or still misuse commands. Hu [18] and Feng [12] discovered that overtime users gain efficiency in command discovery and usage pointing out that it is in the initial stages of voice interaction that users need the most assistance to effectively onboard them to proper use. In all, these works show that user centered research and informed design is required for a more effective voice interaction model.

In this paper we provide a case study of mobile voice interaction design that addresses improving the discoverability and learnability of an unreleased M-VUI application code named VoiceNavigator. An initial usability study with VoiceNavigator revealed that users face several challenges for effective and satisfactory interaction. Based on the findings of the initial study, a prototype version of VoiceNavigator with features and functionality informed by adapting previous HCI research on software learnability and discoverability was developed and tested. We close our paper with describing several design implications for discoverability, learnability accessibility for M-VUI interaction design.

RELATED WORK

Learnability and Discoverability of VUIs

Learnability is one of the key aspects to overall usability of a technology [23]. According to Grossman [7] learnability is defined as *"the ease with which new users can begin effective interaction and achieve maximal performance." Learnability* is generally measured in terms of the time it takes for a user to reach a reasonable level of usage proficiency. This temporal aspect is important to consider as learnability is most challenging during user's initial encounters with a technology [23].

Chen [15] points out several factors that collectively contribute to learnability:

- The availability and effectiveness of training/tutorials
- The system's ability to actively assist users in becoming proficient in use
- Degree of user past experience and skills
- Effective feedback and affordances of system
- The degree of difficulty users face in the process of discovering system functionality

The last point, Discoverability, is particularly difficult for voice interactions as they are inherently invisible. Yankelovich [32] discusses how the ephemerality of voice interactions introduce two fundamental challenges in regards to learnability and discoverability:

1. Users will assume the system can understand more than it is actually capable of
2. Users will be unaware of functionality that is available.

Karsenty further discusses how the opaque nature of VUIs inhibit user's ability to generate accurate mental models [21]. In fact, there is a direct relationship between mental model formation, discoverability, learnability and overall usability of a technology. Looking more specifically at the role of discoverability reveals how it factors in several of Norman's [24] design principles. For instance, consider the principle of *Mapping* which states that a control should appropriately relate to its effect in the world. Good mapping needs to take advantage *"of physical analogies and cultural standards"* which in turns simplifies both learnability and discoverability. Mapping voice commands in VUIs is much more complicated due to lack of physical analogies and wide variety of cultural standards in natural language. The most widely used approach for achieving mapping in VUIs is the so called *"Say What You Can See"* approach [32] which is simply that any voice interaction available should be clearly labeled and visible on the interface. Christian et al [8] and also James [19] work provide examples of this technique. This has shown to be effective for desktop applications but is much more difficult to achieve on mobile devices which do not have the same amount of screen real estate to adequately label every possible interaction. Another challenge occurs with Norman's principal of *Constraint* which are physical forcing functions that are used to limit possible interaction at particular times in order to simplify user experience. Voice interactions are much more difficult to constrain. Prompts [31] and feedback [28] have been used by VUI researchers in order to constrain voice interactions but these can only ever provide an approximation compared to the effectiveness of physical constraints described by Norman. Finally, there is the principle of *Affordance* which are attributes of an object that provide conceptual insight into how to interact with them. The ethereal nature of voice controls makes perceptions of affordances nearly impossible.

In summation, VUI researchers have taken many different approaches to addressing the fundamental challenges of voice interaction described by Yankelovich and Karsenty. Additionally, HCI research has a wealth of techniques and approaches to learnability and discoverability that is most crucial in the initial user experience with a technology which is especially true of VUIs. For the work here we are interested in applying these bodies of knowledge to improve the learnability and discoverability of VoiceNavigator which seeks to provide accessible use of mobile phones for people with limited hand dexterity. Our work suggests that techniques and design principles that address learnability and discoverability of desktop voice interaction design are limited in the context of M-VUIs. We will elaborate on these limitations in our user studies and how we attempted to address them in our prototype design. In the next section, we introduce the VoiceNavigator application followed by the results user evaluation that prompted us to explore alternate interaction design techniques.

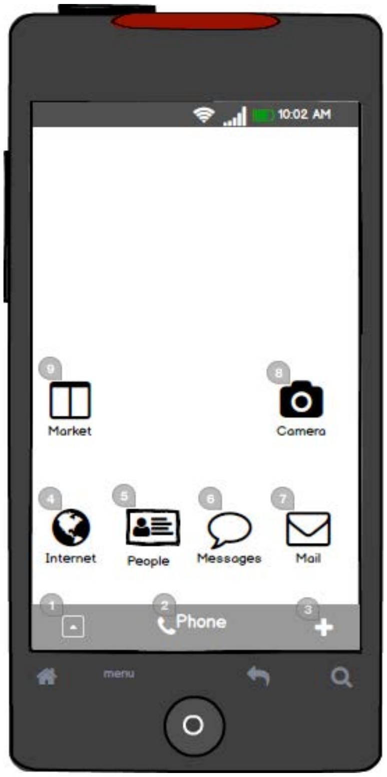


Figure 1. Wireframe of VoiceNavigator running on home screen of a mobile device.

VOICENAVIGATOR DESCRIPTION

VoiceNavigator is an M-VUI application currently under development (Figure 1). VoiceNavigator utilizes speech recognition to provide voice interactions that emulate touch gestures allowing users with limited hand dexterity to perform common mobile device activities such as launching applications, navigation, composing and sending messages, and making phone calls. VoiceNavigator works at the operating system level to identify all possible interactions available in a current screen. For instance, take the common task of launching an email application while on the home screen of a phone and then refreshing the inbox for latest messages. To launch the email application one must physically double tap the icon and then find and select the refresh option. This sequence can be difficult and time consuming to execute for someone with a strong hand tremor. VoiceNavigator allows users to perform the same task via voice commands. To do so the user needs to say: "Open Email" and then "Refresh" to update the inbox. Additionally, VoiceNavigator also offers a series of global actions that can be used from any screen to navigate (e.g. "go home" to return to the homescreen), modify the state of the device (e.g. "turn on Wi-Fi"), show system windows (e.g. "open recent applications"), in addition to launching apps. Users can say "Help" or "What Can I Say?" to open a menu with a list of voice commands and access to the tutorial. In addition to speaking the name of the application or desired interaction VoiceNavigator also provides numbered labels that are superimposed over icons on the screen. For example, in the email application all available features (refresh, compose, drafts, etc.) are typically graphically represented by various icons. In these situations, it may not be immediately clear what functions are available as the icons may or may not be accompanied with a text label. To assist users in these scenarios, VoiceNavigator places a small numerical label adjacent to each function. For instance, in figure 1, a user can say “phone” or the number “2” to bring up the dialing menu.

PRELIMINARY USER EVALUATION

The purpose of this study was to observe users with limited hand dexterity who are new to VoiceNavigator in their first interactions. The study aimed to answer the following questions:

- How can the usability of VoiceNavigator be improved?
- Do users know what to say to navigate VoiceNavigator?
- Are users discovering the core functionalities of VoiceNavigator?
- Do the VoiceNavigator Tutorial enable participants to use the application independently?

We recruited nine participants who have a variety of motor impairments that limit their hand dexterity (see Table 1) by collaborating with local essential tremor groups, rehabilitation centers and a Parkinson’s association.

User	Gender	Age	Disability	Mobile Phone	Length of time with impairment
1A	Female	51-60	Arthritis	Android	3-6 years
2A	Male	41-50	Spinal cord injury	Android	+18 years
3A	Female	24-30	Split level Quadriplegic	Android	3-6 years
4A	Male	24-30	Spinal cord injury	Android	3-6 years
5A	Female	24-30	Spina bifida	iPhone	+18 years
6A	Female	51-60	Multiple sclerosis	iPhone	7-17 years
7A	Male	41-50	Muscular dystrophy	Android	+18 years
8A	Female	41-50	Muscular dystrophy	iPhone	+18 years
9A	Male	24-30	Essential tremor	Android	+18 years

Table 1. Preliminary user evaluation participants.

Methods and Participants

We conducted ten usability study sessions (90 min each) with participants who have a variety of motor impairment that make it difficult or impossible for them to use a smartphone due to their inability to easily operate a touch screen. Studies were conducted in video equipped UX labs. Audio and video documentation was collected from each session with permission from participants. The VoiceNavigator software was tested on a Nexus 5 Android device running the operating system Android Kitkat.

After a brief introduction participants completed the VoiceNavigator tutorial that launches automatically when the application is first activated. Upon completion of the tutorial participants were allowed to explore the application freely before being instructed to complete a set of common use cases. Example use cases were to switch to a different Wi-Fi network, composing an email and creating a detailed calendar entry. Participants were encouraged to articulate questions and were provided assistance if they were unable to complete a specific task. At the end participants were asked to reflect on their experiences, provide feedback, and suggestions how to improve the user experience.

Results

How can the usability of VoiceNavigator be improved?

One of the biggest usability issues detected was the fact that VoiceNavigator was not able to offer a continuous listening mode at the time of the study. Participants expected to be heard and understood at any point in time and be listened to as long as they continued talking, even when pausing to reflect in between. Participants disliked the visual clutter caused by the fact that every touch target came with a box and a number associated. Users occasionally even missed relevant information due to the fact that parts of the visual interface were blocked by the overlay from their view. When switching from navigation to dictation mode, users were unsure at which point their voice input was perceived as dictated text (eg. “and then I want you to go home”) versus navigation commands (“go home” to navigate back to the homescreen).

Do users know what to say to navigate VoiceNavigator?

We found that users are guessing for the most part when trying to figure out what to say to archive a certain action to happen. It gets harder to guess what to say when there are many visual elements without obvious names/labels on the screen. Some users would consult the “What can I say?” (“WCIS?”) menu. But not everyone remembered that this documentation of voice commands was available to them and what to say in order to open it. The numerical system VoiceNavigator offers is perceived as a convenient fall back mode. Especially when going through a longer user journey the numbers became popular shortcuts and bridge the gap when users didn’t know what to call a certain touch target. Some users developed the strategy to think about how they would perform a certain physical interaction and then translate that into a voice command.

Are users discovering the core functionalities of VoiceNavigator?

Guiding novice users in their onboarding experience by offering a tutorial proved to be very effective. All users were able to apply commands that they learned in the tutorial later on within the ecosystem of their phone. They added to that knowledge by trying additional commands. Once a command led to a successful interaction, users remembered it and were often able to apply the same command in a similar situation. A few users learned additional commands throughout the journeys by consulting the “WCIS?” menu. All study participants wished for a more comprehensive structure of the WCIS? menu.

Do the VoiceNavigator Tutorial enable participants to use the feature independently?

All participants felt confident to try VoiceNavigator after they had completed the tutorial. They were able to apply simple commands after the tutorial such as launching the VoiceNavigator, scrolling, navigating the system UI by numbers or by articulating the names of touch targets. The test users articulated a bigger need for learning and secure practicing opportunities before entering their applications independently. For that purpose, they wished for a safe practice and experimenting mode in the context of their most used applications. As the particular interaction model of VoiceNavigator was new to all participants and significantly different to their previous experiences with voice interaction systems, several users expressed the wish for an introduction and demonstration of the core functionalities in form of a video. The list of “What can I say?” commands was appreciated but users wanted it to be better structured, contextually relevant to their current challenge and always in reach while interacting with the system. A tutorial can build the foundation for enabling users to do first steps within VoiceNavigator. For continuous, independent and successful interactions, users will need additional support and help throughout the onboarding process.

Discussion

We found that users need additional clues and help to be able to use VoiceNavigator in the context of complex tasks. Especially in initial interactions users are essentially guessing how to formulate commands. Their previous experience with voice interaction systems influences their mental model of possible interactions as well as their preferred interaction style. The tutorial was essential to onboard users to the new interaction model and teach them product specific details such as the numbers associated with touch targets. Even interaction commands that they learned and applied within the tutorial were not remembered by all participants when asked to perform tasks later on that required these commands. Their willingness to learn a new system and to tolerate weaknesses and shortcomings of the system appears to be directly linked to the severity of impairment and need of the user to have voice as an alternative access form available to them.

Based on the results of this study it became clear that enhanced discoverability and learnability mechanisms would be needed to improve the user experience of VoiceNavigator. In the next section we highlight some of the concepts we developed to improve the learnability and discoverability of the system.

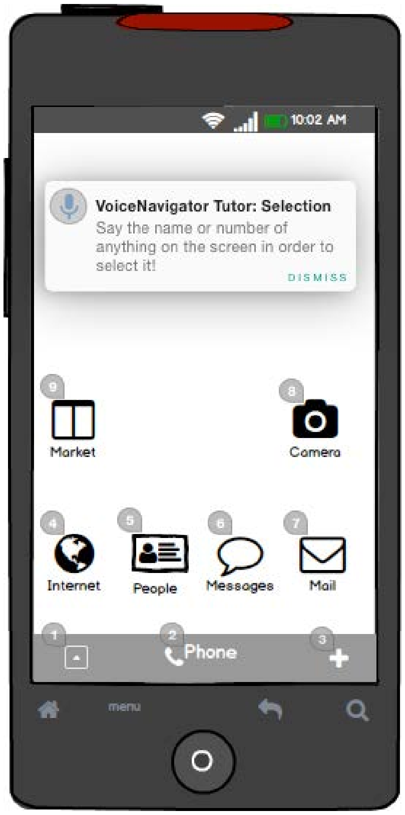
IMPROVEMENT FOCUS

Learn “as-you-go”

The availability and effectiveness of tutorials and training material is a key element in learnability. The initial study revealed that some of the concepts covered in the VoiceNavigator tutorial were not being transferred to actual use. We considered the work of Hakuline et al [16] which looked at the results of training users to interact with a VUI using an interactive tutor vs static web tutorial. They conducted an experiment in which each group was trained prior to using a VUI email application. The results showed that users who were trained with the interactive tutor developed a more favorable mental model of the system. The interactive tutor was effective in improving user learning in the first 20 minutes of interacting with the system.

Building out a full interactive coach was beyond the scope of the work here but the main takeaway from the Hakuline study was that learning appears to be more effective if it occurs in context of the actual use environment. To further inform our approach, we took Krilsler et al [22] in consideration. Their research work evaluated how to provide contextually situated tutoring for the teaching of HotKey usage. They developed an application named “HotKeyCoach” that launches usage suggestions as pop-up prompts to the participant. The application supports user learning more effectively by integrating into the context of use.

Based on these insights we decided to reduce the introduction tutorial to three screens in an attempt to move learning directly into the context use. We wanted the in-application tutor to have the conversational tone of similar to what Hakuline introduced but with the minimal design and intrusion of Krisler’s HotKeyCoach. Drawing on these works, we prototyped the following feature (Figure 2):


VoiceNavigator Tutor: Selection

Say the name or number of anything on the screen in order to select it!

DISMISS

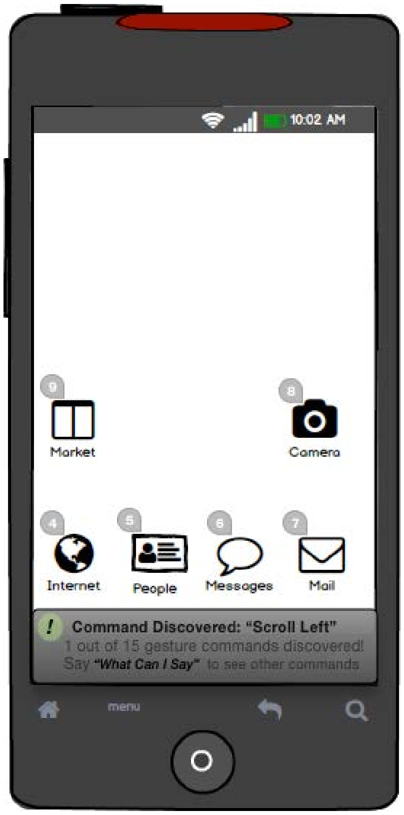
Figure 2. In application tutor message. This message automatically dismisses itself after user completes the suggestion.

There were eight tutor messages in total. Each message covered key interactions of VoiceNavigator. These tutor messages would only appear on the home screen. The message box would highlight then dissolve after completion.

Discovery Based Learning

In our initial study we observed how users would take a guess-and-see approach to interaction with Voice Navigator. It would seem this trial and error approach is a natural human inclination to voice interaction as it has been observed in previous VUI research as well. We were interested in determining how we could harness this behavior in order to improve learnability; in other words, improving learnability through *guessability*. In [30], Wobbrock states *"a user's initial attempts at performing gestures, typing commands, or using buttons or menu items must be met with success despite the user's lack of knowledge of the relevant symbols. This requires high guessability."* To explore harnessing the potential of learnability, we considered Discovery Based Learning which was first introduced in the area of learning psychology [2] but has found much applications for HCI researchers seeking to improve learnability [5]. Discovery-based learning *"encourages learning by exploring and interacting with the environment, wrestling with questions, and performing experiments.* [10]" Dong et al [10] applied this concept to develop a learning game "Jigsaw" in Adobe Photoshop designed to teach users how to use the tools and techniques of the software. The game promoted learning in a playful, gamified manner.

Based off Dong's work, we prototyped the following feature (Figure 3):


Command Discovered: "Scroll Left"

1 out of 15 gesture commands discovered!

Say *"What Can I Say"* to see other commands

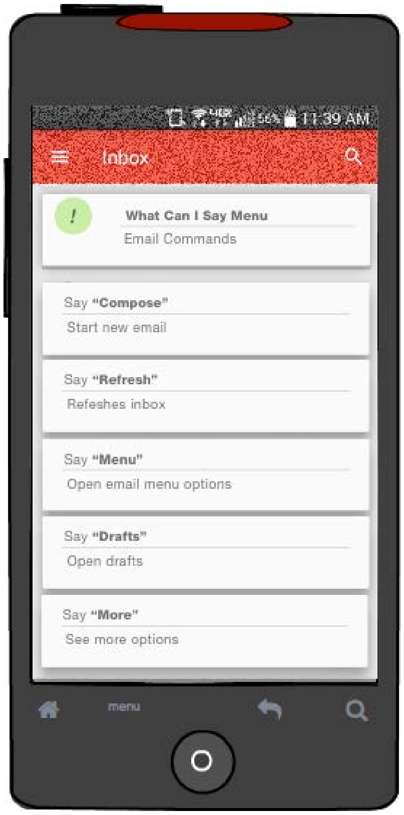
Figure 3. Discovery message. These appear whenever a command is used for the first time.

These "Discovery prompts" aim to provide feedback and guidance in guessability. By providing confirmation of a command and also revealing the existence of other similar commands we hoped to encourage discovery and guessing in a more structured manner. The discovery prompts would slide in from the bottom portion of the screen whenever a voice command was used for the 1st time.

Contextualized Help

The structure and utility of the list of commands in the "What Can I say Menu" was another area that had shown a need for improvement. Users wanted help material to be contextually situated as opposed to a static menu of commands that can be used in general. Furthermore, it would appear that even naming the command to active the menu "What Can I say?" is problematic as it does not convey expected functionality. We observed that users took the phrase "*What Can I Say*?" quite literally to be asking a question to the system in current context; as in "*What Can I Say **Here**?"* as opposed to "*What Can I Say **in General?***" Carroll et al [6] also addressed the importance of contextualized, user relevant help material. In two projects, the Minimal Manual [6] and Guided Exploration [4], Carroll demonstrated how help material should be designed to provide immediate assistance that is relevant to the task or goal at hand as opposed to an overall and fundamental learning of the system.

Based off Carroll's work and our own research findings, the What Can I say Manual was re-designed as shown below (Figure 4):


What Can I Say Menu
Email Commands

Say "Compose"
Start new email

Say "Refresh"
Refeshes inbox

Say "Menu"
Open email menu options

Say "Drafts"
Open drafts

Say "More"
See more options

Figure 4. Example of contextualized help menu in an email application.

The new menu was designed around actions available in the current context (help menu items would be different in an email application vs a calendar app).

PROTOTYPE USER EVALUATION

A second user study was undertaken to elicit initial feedback on the design and functionality of our new learnability and discoverability features. We aimed to answer the following research questions:

- Do the in app tutorial messages improve knowledge of commands?
- Do the discovery messages improve knowledge of commands?
- Do the context specific "What Can I Say" menu improve knowledge of commands?
- How does the new intro experience affect initial learning?

Methods and Participants

We conducted a total of five study sessions (90 min each) with participants who have a variety of motor impairments that limit their hand dexterity (see Table 2). The prototype of VoiceNavigator was used on a Google Nexus 5 device running the Android Marshmallow operating system. We decided to recruit new participants for this study in order to avoid potential carry over from previous interactions with VoiceNavigator. Similar to the previous study, participants first completed the tutorial and were allowed to explore the application for a few minutes in an unguided manner. We then instructed them to complete the exact same the tasks as outlined in the previous study. Each study session we ended with feedback and reflection phase with the participants.

User	Gender	Age	Disability	Mobile Phone	Length of time with impairment
1B	Female	51-60	Essential Tremor	Android	7-17 years
2B	Male	51-60	Essential Tremor	iPhone	+18 years
3B	Male	51-60	Essential Tremor	iPhone	+18 years
4B	Female	24-30	Muscular Dystrophy	Android	+18 years
5B	Male	24-30	Spinal Cord injury	Android	7-17 years

Table 2. Prototype user evaluation participants.

Results

Do the In App Tutorial Messages improve knowledge of commands?

All participants were able to remember and use commands introduced by the tutor after it concluded. They were also successful in completing all the In App Tutor Messages. Several users expressed concerns about being able to dismiss messages as they were afraid that they might need that information later again and would not know how to retrieve it. We also found that the tutor messages did not significantly obfuscate the interface. The In App Tutor messages shortened the lengthy learning experience, shifted successfully the learning experience into the app environment and provided direct experience applicable to daily app usage.

Do the Discovery Messages improve knowledge of commands*?*

We found that the discovery messages competed around user's attention and made them feel rushed. Participants reported to feel as if they might have missed something. Others felt that the messages were irrelevant to them in that particular moment as they did not contain useful information for their exact app interaction. While the messages failed to motivate users to explore in a guided manner, one user said: "*Maybe I don't care what I did, as long as what I did was successful" (P4B).*

Do the context specific "What Can I Say" (WCIS) Menu improve knowledge of commands?

All participants were delighted by the contextualized WCIS menu as it often contained hints for them that were directly applicable to their current challenge. We found that users expected commands to be executable from within the menu. Moving forward the menu needs clear indication of scrollable content and an option for dismissal. One user expressed their delight the following way: *"I can see this being something I'm going to be using all the time as it is easy to access and easy to dismiss." (P3B).*

How does the new Intro Experience affect initial learning?

The new introduction experience provided a clear and streamlined experience. Users who were exposed to both options to compare, preferred the new version for being more practice oriented and showing exercises in the context of real world applications. At the same time user feedback suggested that the visual saliency of numbers needs to be improved. All users were able to perform basic navigation and selection tasks independently after completing the introduction: *"This seems very polished; what I would expect from a professional application" (P4B)*

Discussion

The new tutorial provided a streamlined onboarding experience that prepared users for their first interaction. Future work will need to quantify the best cut-off point for tutorial length. All users appreciated the contextualized "What Can I Say" menu for providing quick access to relevant resources when in need. The In App Tutor Messages were appreciated but its effects on user efficiency are inconclusive. The discovery messages were less effective than anticipated as the visual attention span is limited and users overall did not perceive the content to be relevant/. Moving forward VoiceNavigator should continue to shift as much learning content into the real app environment as possible. At the same time, it will be essential to work on concepts that help users in managing their attention span and direct their focus to those interaction commands that are most relevant to them in any given moment.

IMPLICATIONS FOR M-VUI DESIGN

Contextualized Help

Contextualized help functionality is potentially the most useful asset to learnability and discoverability in the context of M-VUIs. The help content should revolve around actions that are available and relevant in the current context rather than general information. Furthermore, each action displayed in the help should be actionable from the menu screen rather than making the user close the menu, recall the correct command and then speak it to perform the action. Allowing users to perform the action while in the menu screen reduces the cognitive load and lowers the amount of steps involved to perform an action. From a design perspective, this means the menu should take the form of a transparent overlay that provides users with available actions without significantly obscuring the current screen environment.

Overall, our research findings suggest that the onboarding and learning experience of M-VUI need to be designed for action rather than overall knowledge. A helpful way to frame why this is important was provided by Carroll et al [5] in describing the "Active User Paradox" wherein: *"New users tend to jump right in when introduced to application systems. If an operation is referred to in their training materials, they want to try it out at once. Rote descriptions and practice are resisted..."* In the context of M-VUI, the Active User Paradox appears to be especially prevalent. This may be a natural response to the context of mobile HCI being designed for on-the-go and quick interactions. In other words, it may be that users have an even lower bar of patience for learning in mobile HCI than they have when seated in front of a desktop computer. This suggests additional research in the area of learning capacity and how device (desktop vs mobile) and interaction style (voice vs direct manipulation) affect it.

Key-Takeaway:
M-VUI help material should be designed around actions available and relevant in a specific context. Furthermore, actions should be executable from within the help material rather than having additional steps to recall commands after exiting.

Assimilation Bias

Unlike other popular voice applications on major mobile operating systems (Google Now, Siri) VoiceNavigator does not operate as a conversational agent that provides verbal feedback or accepting conversational input. Instead, VoiceNavigator operates as a utility feature with a limited set of commands and scope. While it does allow for some high level global commands and synonyms for commands it is not designed for a high level natural voice interaction. This is problematic as users assume the mental model of interaction from other popularly available conversational agents (Siri, Cortana) to VoiceNavigator. This is also known as "negative transfer" which is when current learning is inhibited by prior experience. Carroll et al [5] discussed how negative transfer creates "Assimilation Bias" which: *"can mislead users about how a system will work, but also it can in a sense put blinders on them, preventing them from fully anticipating the function available. This negative effect of prior knowledge can be especially debilitating, because often a learner may be completely unaware of the problem."*

While natural language and conversational feedback work well for information queries, a more limited command vocabulary may be more effective in the context of complete hands-free control and navigation that VoiceNavigator seeks to enable. Limited command vocabularies tend to be more accurate than conversational styles (in terms of speech recognition error rates) [14,25]. However, ultimately regardless of what form of interaction is more appropriate for the context of use (conversational vs command) it would appear that for M-VUIs conversational style is preferred and desired at least from a user experience perspective. This has motivated current research within the development team to works towards exploration of how VoiceNavigator can function in a more conversational manner.

Key-Takeaway:
The design of any M-VUI should take into consideration how the experience with conversational voice agents will affect user's expectations, behaviors, and ultimately satisfaction with a new application.

Designing M-VUIs for Limited Visual Attention

Designing around the limited screen space of mobile devices is a well known problem for mobile interaction designers. Voice Navigator's initial design outlined all available actions on the screen by drawing a square box highlighting the actionable item in addition to a number label associated with it. As shown, in the first study we conducted, this design scheme can be difficult at times when the screen becomes cluttered causing significant confusion and frustration.

We addressed this concern by introducing a more minimal design in the prototype version which used the number labels only without the grid structure. This appears to be an effective design choice; users when exposed to both designs for comparison overwhelmingly preferred the new design. While this new design made steps forward in improving the visual design, two of the new features introduced: the tutor messages that appear towards the upper half of the screen and the discovery messages that slide-up from the bottom portion of the screen inadvertently introduced new challenges. Although these techniques have been shown to be effective in use on desktop environments for learning and training, they appear to be much less appropriate on mobile due the limited screen space and visual attention.

Key-Takeaway:
M-VUI interface design should take a minimal design approach respecting the limited screen space and be considerate not to overload user's visual attention span.

Training in Context

The initial study suggested providing an extensive tutorial prior to the actual use is ineffective for learning. Realizing this we attempted to shift learning into the actual usage environment with the in-application tutor messages. Additionally, we wanted to explore techniques to assist user's natural tendencies for guessing and trial-and-error approaches. Prior research on discovery-based learning seemed to be a promising approach to do so. Unfortunately, our particular implementation of discovery-based learning presented new challenges due to screen space and limited visual attention span. Therefore, it is difficult to make strong claims for the efficacy of these approaches.

Key-Takeaway:
M-VUI initial training should seek a balance in introducing users to interaction by providing a brief overview then shifting learning into use context. Discovery-based learning seems to be a promising technique but a variety of approaches how to implement it effectively need to be explored.

Complete Mobile Accessibility

As described above learnability and discoverability are particularly challenging in the context of invisible voice interfaces. When it comes to the context of users with accessibility needs the situation is even more challenging. For most mainstream users voice interactions offer convenience and an additional way to do things that matter to them on their mobile devices. In contrast, for users with severe motor impairments affecting the dexterity of their hands, an M-VUI can be an empowering experience enabling complex interactions such as opening, reading, answering, forwarding and archiving an email conversation. This represents end-to-end accessibility whereas existing mobile voice interaction via Siri or Google NOW do not offer completely hands free interactions for all tasks. We encourage the research community to think about the complete user experience from the perspective of those who do not have the ability to bridge the gaps of current mobile voice interaction software. VoiceNavigator has the great potential to fill some of these gaps and eventually enable a truly hands free M-VUI.

Key-Takeaway:
Accessibility researchers should consider the entirety of the interaction needs of users with limited hand dexterity when designing M-VUIs.

CONCLUSION

As mobile devices and the use of voice interactions continue to proliferate, mobile HCI research that addresses the specific challenges for learnability and discoverability is needed. Our work in this paper provides a case study describing our efforts to improve the learnability and discoverability of an M-VUI application code-named VoiceNavigator which seeks to provide completely hands-free voice control access to a mobile phone. Our initial user evaluation highlighted several usability challenges for VoiceNavigator. We attempted to address these challenges by prototyping concepts informed by previous research on learnability, discoverability and VUI interaction design. We found that while previous research from desktop HCI can be useful for inspiration or initial starting points, some specific techniques are not directly applicable. We argue for the future development of voice interaction design theory aimed specifically at the mobile space. As mobile HCI continues to grow in equity and scale, the time has come for this field to define its own theories rather than having to constantly borrow and translate from desktop HCI. An exemplar of this need, as shown in this work is in the space of voice interactions. Voice interactions hold great promise for mobile HCI but are in many ways still a new frontier. In this paper, we explored one possible frontier in terms of voice accessibility for individuals with limited hand dexterity. We anticipate exploration into other application areas along with development of theory and design principles rooted in mobile to accompany it.

REFERENCES

1. James H. Bradford. 1995. The human factors of speech-based interfaces: a research agenda. *ACM SIGCHI Bulletin* 27, 2: 61. http://doi.org/10.1145/202511.202527
2. Jerome S. Bruner. The act of discovery.
3. Ron Van Buskirk and M LaLomia. 1995. A Comparison of Speech and Mouse/Keyboard GUI Navigation. *Conference companion on Human factors in*: 89791. Retrieved from http://dl.acm.org/citation.cfm?id=223447
4. John M. Carroll, Robert L. Mack, Clayton H. Lewis, Nancy L. Grischkowsky, and Scott R. Robertson. 2009. Exploring Exploring a Word Processor. *Human–Computer Interaction* 1, 3: 283–307. http://doi.org/10.1207/s15327051hci0103_3
5. John M. Carroll and Mary Beth Rosson. 1987. Paradox of the Active USer. In *Interfacing Thought: Cognitive Aspects of Human-Computer Interaction*. MIT Press, 80–111. http://doi.org/10.1017/CBO9781107415324.004
6. John Carroll, Penny Smith-Kerker, James Ford, and Sandra Mazur-Rimetz. 1987. The Minimal Manual. *Human-Computer Interaction* 3: 123–153. http://doi.org/10.1207/s15327051hci0302_2
7. Fang Chen. 2006. *Designing Human Interface in Speech Technology*. Springer. Retrieved February 13, 2016 from https://books.google.com/books?hl=en&lr=&id=sjJo_8QfEo8C&pgis=1
8. Kevin Christian, Bill Kules, Ben Shneiderman, and Adel Youssef. 2000. A comparison of voice controlled and mouse controlled web browsing. *Proceedings of the fourth international ACM conference on Assistive technologies - Assets '00*: 72–79. http://doi.org/10.1145/354324.354345
9. Liwei Dai, Rich Goldman, Andrew Sears, and Jeremy Lozier. 2004. Speech-based cursor control: a study of grid-based solutions. *Assets '04: Proceedings of the 6th international ACM SIGACCESS conference on Computers and accessibility*: 94–101. http://doi.org/http://doi.acm.org/10.1145/1028630.1028648
10. Tao Dong, Mira Dontcheva, and Diana Joseph. 2012. Discovery-based games for learning software. *Chi 2012*. Retrieved from http://dl.acm.org/citation.cfm?id=2208358
11. J Feng, S Zhu, R Hu, and A Sears. 2011. Speech-based navigation and error correction: A comprehensive comparison of two solutions. *Universal Access in the Information Society* 10, 1: 17–31. http://doi.org/10.1007/s10209-010-0185-9
12. Jinjuan Feng, Clare Marie Karat, and Andrew Sears. 2005. How productivity improves in hands-free continuous dictation tasks: Lessons learned from a longitudinal study. *Interacting with Computers* 17, 3: 265–289. http://doi.org/10.1016/j.intcom.2004.06.013
13. Jinjuan Feng and Andrew Sears. 2004. Are we speaking slower than we type?: exploring the gap between natural speech, typing and speech-based dictation. *ACM SIGACCESS Accessibility and Computing*, 6–9.
14. G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais. 1987. The vocabulary problem in human-system communication. *Communications of the ACM* 30, 11: 964–971. http://doi.org/10.1145/32206.32212
15. Tovi Grossman, George Fitzmaurice, and Ramtin Attar. 2009. A Survey of Software Learnability: Metrics, Methodologies and Guidelines. *Proceedings of the 27th international conference on Human factors in computing systems (CHI'09)*: 649–658. http://doi.org/10.1145/1518701.1518803
16. Jaakko Hakulinen, Markku Turunen, and Esa-pekka Salonen. 2005. Software Tutors for Dialogue Systems. *LNAI*: 412–419.
17. Susumu Harada, Jacob O Wobbrock, and James A Landay. 2007. VoiceDraw: A Hands-Free Voice-Driven Drawing Application for People with Motor Impairments. *9th international ACM SIGACCESS conference on Computers and accessibility*, ACM, 27–34.
18. Ruimin Hu, Shaojian Zhu, Jinjuan Feng, and Andrew Sears. 2011. Use of speech technology in real life environment. *Universal Access in Human-Computer*: 62–71. http://doi.org/10.1007/978-3-642-21657-2_7
19. F James and J Roelands. 2002. Voice over

Workplace (VoWP): voice navigation in a complex business GUI. *Proceedings of the fifth international ACM*: 197–204. Retrieved from http://dl.acm.org/citation.cfm?id=638285

20. Shaun K Kane, Chandrika Jayant, Jacob O Wobbrock, and Richard E Ladner. 2009. Freedom to roam: a study of mobile device adoption and accessibility for people with visual and motor disabilities. *Proceedings of the 9th international ACM SIGACCESS conference on Computers and accessibility - Assets '09*, 115–122. http://doi.org/10.1145/1639642.1639663

21. Laurent Karsenty. 2002. Shifting the Design Philosophy of Spoken Natural Language Dialogue : *International Journal of Speech Technology* 5, 2: 147–157.

22. Brian Krisler and Richard Alterman. 2008. Training towards mastery. *Proceedings of the 5th Nordic conference on Human-computer interaction building bridges - NordiCHI '08*: 239. http://doi.org/10.1145/1463160.1463186

23. Jakob Nielsen. 1994. *Usability Engineering*. Elsevier Science. Retrieved February 13, 2016 from https://books.google.com/books?hl=en&lr=&id=DBOowF7LqIQC&pgis=1

24. DA Norman. 2002. The design of everyday things. Retrieved January 12, 2015 from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Design+of+everyday+things#0

25. Andrew Sears, Jinhuan Feng, Kwesi Oseitutu, and Claire-Marie Karat. 2003. Hands-Free, Speech-Based Navigation During Dictation: Difficulties, Consequences, and Solutions. *Human-Computer Interaction* 18, 3: 229–257. http://doi.org/10.1207/S15327051HCI1803_2

26. Ben Shneiderman. 2000. The limits of speech recognition. *Communications of the ACM* 43, 9: 63–65. http://doi.org/10.1145/348941.348990

27. Hannu Soronen, Santtu Pakarinen, Mervi Hansen, et al. 2009. User Experience of Speech Controlled Media Center for Physically Disabled Users. *Proceedings of the 13th International MindTrek Conference: Everyday Life in the Ubiquitous Era*: 2–5. http://doi.org/10.1145/1621841.1621843

28. Stefanie Tomko and Roni Rosenfeld. 2004. Shaping Spoken Input in User-Initiative Systems. *In Proceedings of the 8th International Conference on Spoken Language Processing (ICSLP), Jeju Island, South Korea.*: 2–5.

29. Jacob O Wobbrock. 2006. The Future of Mobile Device Research in HCI. *Access*, 131–134. Retrieved from http://faculty.washington.edu/wobbrock/pubs/chi-06-wkshp.pdf

30. Jo Wobbrock and Hh Aung. 2005. Maximizing the guessability of symbolic input. *CHI'05 extended abstracts*: 5–8. http://doi.org/10.1145/1056808.1057043

31. Nicole Yankelovich, Gina-Anne Levow, and Matt Marx. 1995. Designing SpeechActs. *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '95*: 369–376. http://doi.org/10.1145/223904.223952

32. Nicole Yankelovich. 1996. How Do Users Know What to Say? *Interactions* 3, 6: 32–43. http://doi.org/http://dx.doi.org/10.1145/242485.242500

33. Yu Zhong, T V Raman, Casey Burkhardt, Fadi Biadsy, and Jeffrey P Bigham. 2014. JustSpeak: Enabling Universal Voice Control on Android. *Proceedings of the 11th Web for All Conference on - W4A '14*: 1–4. http://doi.org/10.1145/2596695.2596720