

Accessing Media Via an Audio-only Communication Channel: A Log Analysis

JOHANNE R. TRIPPAS, University of Melbourne, Australia

DAMIANO SPINA, RMIT University, Australia

MARK SANDERSON, RMIT University, Australia

LAWRENCE CAVEDON, RMIT University, Australia

Studies of interaction log analysis are a common tool to investigate behavioural data and contribute to insights into users' interaction patterns with a system [11, 18]. We present a log analysis from a bespoke conversational system, *RealSAM*¹, an audio-only interaction media assistant in which users can navigate and interact with media content through natural language. The novel assistant is designed for people with a vision impairment or other disability that prevents a person from accessing printed material. The exploratory analysis was conducted to provide an initial insight into the communication and interaction behaviours. We focus on understanding how users utilise the application. The results are twofold, we highlight the (i) implications for the design of future voice-enabled systems such as "infinite-reading" mode, enhanced interaction management enabling file navigation or time-compression techniques, and (ii) challenges of analysing conversational logs and suggest guidelines making these logs more accessible for future research.

ACM Reference Format:

Johanne R. Trippas, Damiano Spina, Mark Sanderson, and Lawrence Cavedon. 2021. Accessing Media Via an Audio-only Communication Channel: A Log Analysis. In *Woodstock '18: ACM Symposium on Neural Gaze Detection, June 03–05, 2018, Woodstock, NY*. ACM, New York, NY, USA, 10 pages. <https://doi.org/10.1145/1122445.1122456>

1 INTRODUCTION

Displaying search results for people with a visual impairment is problematic. Systems such as Siri allow users with a visual impairment to pose queries, but they will not receive answers to their query via audio unless it is a factoid question (e.g., "How deep is the ocean?"). For non-factoid or ambiguous questions, this user group relies on additional assistive software (e.g., screen reader, VoiceOver², or TalkBack³) to translate the written search engine results page into speech [17]. Thus, a user with a visual impairment who uses Siri to search must switch to using assistive software to read out the search results furthering the challenge of accessing information, increasing the likelihood of unsatisfactory interaction [1].

We use a new commercial system's, *RealSAM*, logs to understand the interaction behaviour between users and the application. *RealSAM* is a bespoke application for people with a visual impairment, in contrast to Siri or Alexa, with which users can interact and search for audio material, such as podcasts, news articles, and audiobooks, exclusively via an audio-only interaction channel. The application is tailored to provide accessible media for people who are visually impaired.

¹<https://realthing.ai/products/realsam-accessible-phone-media-player-assistant/>

²<http://www.apple.com/au/accessibility/osx/voiceover>

³<https://support.google.com/accessibility/android/answer/6007100?hl=en>

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2021 Association for Computing Machinery.

Manuscript submitted to ACM

This paper is structured as follows. Section 2 provides an introduction to the *RealsAM* application and its target audience, including an overview of the interaction methods and the content which can be accessed. We then describe the available dataset in Section 3, followed the results of the log analysis in Section 4. We show how *RealsAM* is used over time, including the interaction frequencies based on pre- and self-defined interaction categories. We describe sessions which consist of a single interaction and introduce search interaction behaviours. We conclude the results section by displaying the text-to-speech (TTS) output settings of *RealsAM* users. Section 5 discusses the results and limitations of the study. Finally, we conclude this paper with a summary and conclusion in Section 6.

2 REALSAM APPLICATION

RealsAM consists of a Samsung Galaxy Pocket with a single-app Android ROM installed on it (see Figure 1). This device has a central button on the bottom front of the device, the *talk button*. When users press this button, they can either start their spoken interaction or interrupt (i.e., *barge-in*) the device. Only the volume buttons on the device work; however, the other buttons and touch functionality of the screen are disabled for accessibility reasons. Users can also turn on a *hands-free mode* which allows them to interact with the device without pressing the talk button. However, in this mode, *RealsAM* will only start listening again after it has finished speaking and thus users cannot interrupt.



Fig. 1. *RealsAM* device.

RealsAM provides the following five categories of content:

- (1) **Podcasts:** Listen to podcasts from sources such as the Australian Broadcasting Corporation (ABC) or the British Broadcasting Corporation (BBC).
- (2) **Newspapers:** *RealsAM* currently indexes news from ABC News, The Conversation, The New Daily, and a wide range of papers provided by the Vision Australia Library, including The Age, The Sydney Morning Herald, and The Australian.
- (3) **Books:** *RealsAM* provides access to the books offered by The Gutenberg Project⁴ and Bookshare⁵.
- (4) **Service:** *RealsAM* allows users to check the current time, weather conditions, and geographical location.
- (5) **Device:** *RealsAM* provides commands to configure the device, check the battery level, or listen to announcements from *RealThing*.

RealsAM uses sound cues (i.e., ear-cons or discourse markers) to guide the user through the system. For example, a *falling tone* and a *tick tock sound* means that *RealsAM* is considering the user's request and will respond soon. When a

⁴<https://www.gutenberg.org>

⁵<https://www.bookshare.org>

user submits a command, the device *RealsAM* presents the first five results to the user with an option to hear more results. Thus, one “result page” consists of five results. An example interaction is shown below.

User: Which newspapers do you have?

RealsAM: I have the following newspapers:

1. ABC News
2. Adelaide Advertiser
3. The Age
4. The Australian
5. Australian Financial Review.

Please select one or say continue.

User: Number 3

RealsAM: OK, selecting The Age. The first page of 29 unread headlines from the News Section:

1. Faulty fire system puts lives at risk
2. Mum’s the word in Melbourne
3. Greens go for...

User: Read me the Finance section from the Australian.

[*barge-in*]

Interactions with *RealsAM* are classified based on system-defined rules triggered by pre-mapped voice inputs. For example, *RealsAM* starts reading news headlines when a user input “*read me the news headlines*”. The system classifies this interaction as “* *headlines* *” and is an illustration of the inherent linguistic and functional limitations of this restricted system.

3 DATASET

The log set includes interactions between 17 February 2014 and 17 May 2016. Input interactions can be seen as a voice command to the system. This voice command is then translated into a text command using automatic speech recognition (ASR) and from this point onwards it is treated as a text command. The output text is translated with TTS for the user to listen to. The audio output is in contrast to many multi-modal systems where the input is by voice but returns results using the standard mobile or desktop interface (i.e., the screen).

Each interaction or voice input has a timestamp (beginning of interaction), anonymised user ID, output interaction from the system, voice type and speed, and the system rule triggered by the input received. However, no information is recorded as to whether the user barged-in to the application and there are no end timestamps.

4 REALSAM LOG ANALYSIS

We first present the general descriptive statistics about the logged *RealsAM* interactions and examine the pre-identified *RealsAM Interaction Categories*. We then continue to group these Interaction Categories in *Interaction Themes* allowing us to investigate how people use *RealsAM* through communication, one-interaction, and search sessions. The final part of this section discusses the user settings of the TTS output.

4.1 General and Session Descriptives

The *RealsAM* interaction log consists of 411,201 interactions from 236 unique users. An interaction comprises of an action from the user and a reaction from the system. Interactions are grouped in sessions where a session lasts until there are at least 15 minutes of inactivity [8, 13]. The interaction log contains 46,859 sessions. On average, users spent 19.74 minutes per session⁶. The average sessions per user were 199 (median is 23, mean is 198). There were 8.77 interactions per session. A total of 24,507 sessions (52.29%) consisted of only one interaction.

When we examine the *RealsAM* session patterns over a 24-hour time frame, we observe that more sessions occur in the mornings throughout the 7-day week. However, when comparing weekdays (Monday to Friday) and weekend (Saturday and Sunday) days, we notice a trend that users interact more frequently with *RealsAM* during weekday morning hours than weekend mornings as seen in Figure 2. After 2pm on weekdays, the number of sessions declines while on the weekends the number increases.

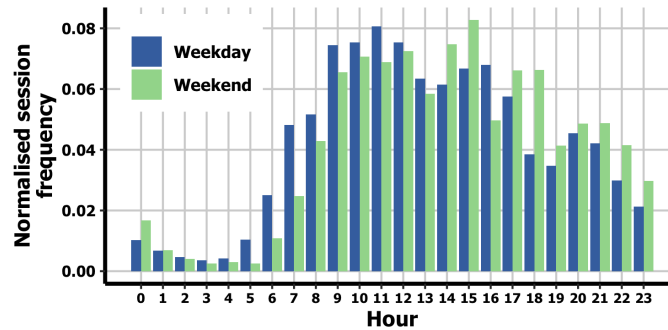


Fig. 2. Normalised session frequency in 24 hours on weekday and weekend days.

4.2 How People use *RealsAM*

We removed all the stopwords, including unrecognised voice input and created a frequency list of the most highly used terms.⁷ We found the most frequent term from the users was “next”, corresponding to 21.72% of the total input terms as seen in Table 1.

Table 1. Top-five frequent input terms.

Input Term	Count (%)
Next	143,399 (21.72%)
Number	38,569 (5.84%)
Read	22,980 (3.48%)
Headlines	18,681 (2.83%)
Back	16,653 (2.52%)

A total of 43,918 distinct pre-mapped rules were recorded in the log. We sorted these rules in 87 interaction categories, including the categories “Null”, defined by the application, and “Other”, which we could not classify. The Null and Other categories accounted for 12.8% and 5.8% respectively of the total logs.

⁶Inactivity is not included as part of the session time.

⁷We used the SMART stopwords list.

We sorted the remaining 85 interaction categories by investigating the voice input transcripts. For example, if the pre-mapped voice input recorded “* headlines *” we examined all rows within the log containing this particular input to conclude that this rule is indeed related to asking for news headlines. We then classified this pre-mapped input accordingly.

Thus, a total of 85 interaction categories were created with the most frequently used categories presented in Table 2. The table shows that several interactions are similar and could be categorised in a *theme*. For example, the category *next article* and *next response* are both navigational interactions indicating reading out the next response and therefore belong to the newly defined theme *Interaction Management*. The classifying processes were conducted iteratively by the main author and reviewed by the other authors.

Table 2. Most frequently used *RealSAM* interaction categories.

Interaction Category	Count (%)
Next article	93,309 (27.88%)
Select response	52,365 (15.65%)
Next response	24,302 (7.26%)
News headlines	16,893 (5.05%)
ASR error recovery	16,819 (5.03%)

We grouped the 85 interaction categories through an examination into themes. These themes create a further abstraction while reducing the number of categories for a more meaningful analysis. The interactions categories are divided into the next five themes:

- (1) **Search (S)**: a user searches for a specific document,
- (2) **Browsing (B)**: a user wants to hear the news headlines,
- (3) **Interaction Management (IM)**: how a user interacts with the device, such as “next”, “stop”, or “resume spoken document”,
- (4) **Device and Service (D&S)**: interactions related to operating *RealSAM*, such as changing the voice or checking the battery and weather⁸, and
- (5) **Error Handling (EH)**: the device attempts to recover from errors.

Figure 3 shows that Interaction Management is the most commonly used. The second most commonly used is Error Handling, followed by Device and Service. The high Interaction Management would be expected given that this category includes the commands to use the device such as resuming a spoken document, navigating to the next section, or repeating an article.

4.2.1 One-Interaction Sessions. As mentioned, 52.29% of the sessions consisted of one interaction. The 15 most frequent interaction categories cover 79.55% of the one-interaction sessions as presented in Table 3. The Search theme did not contain any one-interaction sessions.

4.2.2 Search Sessions. The Search theme consisted of 3,399 (7.25%) sessions where users posed one or more queries. The total number of queries in the Search theme was 6,888, consisting of 2,238 news article searches (32.49%), 2,106 podcast searches (30.57%), 629 book searches (9.13%), and 1,915 (27.80%) unclassified searches. These unclassified searches were due to users posing an unspecified query that the system could not classify in any specified interaction categories.

⁸Weather information is stored on the server and classified as a service.

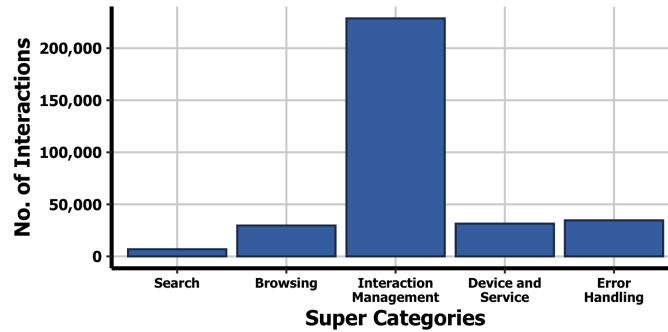


Fig. 3. Interaction frequency of themes.

Table 3. Most frequent one-interaction session categories.

Interaction Category	Theme	Interaction Category Count (%)
Access source	IM	3,873 (17.43%)
Check the battery level	D&S	1,805 (8.13%)
Next article	IM	1,790 (8.06%)
No match found	EH	1,768 (7.96%)
Select response	IM	1,588 (7.15%)
Check the weather	D&S	1,426 (6.42%)
ASR error recovery	EH	1,120 (5.04%)
News headlines	B	780 (3.51%)
User guide	D&S	748 (3.37%)
List books	B	681 (3.07%)
Next response	IM	500 (2.25%)
Time	D&S	496 (2.23%)
Part of command missing	EH	478 (2.15%)
Response to "hello" input	D&S	332 (1.49%)
Go back	IM	287 (1.29%)

NOTE: Browsing (B), Device and Service (D&S), Error Handling (EH), Interaction Management (IM), Search (S)

The average query length for the voice queries was 3.29 words ($SD=1.49$, $max=24$) which were obtained after lowercase conversion, tokenisation, and stopword removal and 4.87 words ($SD=1.88$, $max=31$) without the removal. Query characteristics presented in Table 4 show that 56% of the queries were unique.

Table 5 shows the most frequent terms in search queries. Popular terms suggest that search was used as a mechanism to access specific sources (e.g., ABC, report, show) or to find content related to a given topic (e.g., rugby, wallabies).

4.3 Text-to-Speech Output

This section investigates the voice and speed of the TTS output per interaction. A female Australian voice and 1.0x speech reading rate were the default settings, but six different voices and other speeds are available. Interactions were performed 51.78% of the time in these default settings, where 60.12% had the default speed, and 71.82% used the default female Australian voice. Of all the interactions, 39.54%, were performed either with a slower (18.26%) or faster (21.28%) voice speed (see Figure 4).

Table 4. Query characteristics.

	Count
Total number of queries	6,888
Unique queries	3,872
Most frequent queries:	
Read articles about rugby	406
Read articles about wallabies	214
Play me the health report	69
Play me the science show	63

Table 5. Most frequent query terms.

Query Term	Count	Query Term	Count
Rugby	622	ABC	214
Wallabies	288	Australia	199
Report	265	Science	179
Show	236	Health	172
Vision	232	Margaret	139

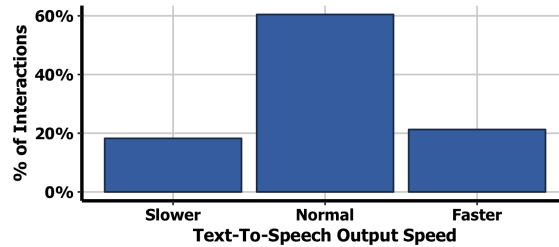


Fig. 4. Speed of the output in the interactions.

5 DISCUSSION

The results show that navigational “next” interactions such as “*next article*” or “*next response*” were frequently used commands. We propose to add an “infinite-reading” mode enabling users to listen to document titles more efficiently. This mode resembles a search engine’s infinite scroll mode which automatically loads and displays next search results when the user reaches the end of the page. Thus, the infinite-reading mode would continue reading the document titles until a user interrupts the system. Furthermore, enabling users to navigate more freely will inform them which documents are available in the “information space” [14, 15].

Two thirds (66.37%) of the one-interaction sessions shown in Table 3 can be interpreted as *good abandonment*. This is where a user accesses the device with a clear goal, retrieves the information, and then leaves the device [7]. System defined interaction categories, such as *Access source*, where a user inputs “*read me ABC news*”, are also considered good abandonment, as are classifications such as checking the battery level or the weather and accessing the news headlines. In contrast, 19.05% of one-interaction sessions can be seen as *bad abandonment*, which is where a user leaves without achieving their goal [7]. Bad abandonment classifications often happened when an error occurred such as *no match*

found, ASR errors, or part of the command is missing. The remaining 14.58% corresponded to noise in the logs (errors splitting the sessions or null interactions).

Concerning the Search theme, the average spoken *RealsAM* query length (4.87 words) is similar to that reported in a recent study of spoken queries from a commercial search engine (4.2 words), but is longer than the length of typed queries (3.2 words) [6]. Guy [6] also reported in this study that one-word queries were rarer in voice (12%). In our dataset, one-word queries were uncommon and only accounted for 1% of the queries. Other researchers have reported that voice queries are on average one word longer than typed mobile queries [19] while Schaller et al. [10] suggested that it may be easier to create long queries with a voice interface than with a keyboard. Although users cannot type queries into *RealsAM*, and we cannot make a direct comparison between typed or spoken queries in *RealsAM*, the longer average voice query and the lack of one-word queries may indicate that users find it more natural to create longer queries. Surprisingly, only a fraction of the interactions involved search behaviours nevertheless, interaction management was high. This may indicate that the search function is less accessible or “discoverable”. Future systems could maybe implement search functions more present, further alleviating the reliance on information management commands.

With the third and fourth most frequent queries “*play me the health report*” and “*play me the science report*”, the user is presented with a search result list. This list consists of the podcasts containing the corresponding query terms (i.e., “*health report*” or “*science report*”) anywhere within the document despite “*health report*” or “*science report*” being the titles of the podcasts (i.e., source). This is in contrast to the first and second most searches “*rugby*” and “*wallabies*” which are general query terms (i.e., topic). However, *RealsAM* only reads out the titles for these podcasts, and as these may not contain the query term, the podcasts’ relevance may be unclear to the user. Therefore it may be helpful for the users to hear their query words in the context of the found document. For podcasts, this may mean that users listen to a snippet extracted from the podcast audio in order to understand the context of their query word [12].

Almost half of the interactions were conducted in the original speed and with a female Australian voice, while 48.23% of the interactions were in a different speed or voice. This kind of personalisation provides users more freedom in their interactions with the content, we have evaluated the effect of audio transformations (i.e., prosodic modifications) and our initial results suggest that some of the proposed prosodic modifications lead to better comprehension and identification of the answers in a snippet at the expense of slightly degraded naturalness of the audio signal [5]. Future research could investigate whether skimming or time-compression techniques, such as pause-based skimming, would be useful [3, 4].

5.1 Limitations

The quality of the logging process, as well as the system’s linguistic and functional limitations, hindered the analysis of the interaction logs. For example, *RealsAM* was updated several times during the data capture process and therefore had different pre-defined rules in place. Simultaneously, each input from the user was logged through text, but no audio file was present to check whether the ASR had correctly recognised the user’s input. ASR input errors may have resulted in 12.8% of the logs with a Null input from the system; however, we could not check this. Furthermore, the *RealsAM* logs did not indicate if a user had barged-in during the output of the text. For example, it was not possible to establish whether the user listened to all results before making a decision of which results they would like to select. Even though, our dataset can be considered old, this is the first log analysis of a bespoke conversational system. Even though many researchers have suggested design guidelines [9, 16] from a user’s perspective for conversational systems,

little is known about the research log analysis aspect. Lastly, we could not utilise the timestamps series fully due to the different speeds in voices and the lack of end timestamps.

5.2 Conversational Log Analysis Suggestions

This paper’s implications are for both researchers working and creating similar logs and system developers who are logging audio-only interactions. Suggestions from this paper include:

- (1) Log the start and end time for each utterance.
- (2) Log each interaction of the user and system separately.
- (3) Log where and when the user interrupted the system output or indicate whether the user listened to the full output.
- (4) Where possible, retain the audio to check ASR errors or add ASR term confidence values in the output transcription [2].

6 SUMMARY AND CONCLUSION

This log analysis aimed to explore interaction and communication behaviours between users and *RealsAM*, a bespoke audio-only application for accessing media. This analysis’s strength is that we were able to investigate people’s in-context interactions with the application. The log analysis provided insight into users’ behaviours and media accessed and how users satisfied their information needs. The discussed findings suggest that a truly conversational system needs further research and development to establish how people want to interact with content over voice without pre-conceived constraints placed on them by the system. We suggest implementing “infinite-reading” modes, time-compression techniques, and enhance discoverability and personalisation methods.

Our analyses suggest that audio-only interactions systems are still in the early stages of their development, as reflected in the need for improvement in navigational commands, query intent recognition, and skimming techniques over audio. This paper concludes that audio-only interactions are not straightforward to log and need to be designed carefully.

ACKNOWLEDGMENTS

We thank RealThing for their support in this research and the anonymous reviewers for their useful suggestions.

REFERENCES

- [1] Ali Abdolrahmani, Ravi Kuber, and Stacy M Branham. 2018. "Siri Talks at You" An Empirical Investigation of Voice-Activated Personal Assistant (VAPA) Usage by Individuals Who Are Blind. In *Proceedings of International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*. 249–258.
- [2] Jaime Arguello, Sandeep Avula, and Fernando Diaz. 2016. Using Query Performance Predictors to Improve Spoken Queries. In *Proceedings of the European Conference on Information Retrieval (ECIR)*. 309–321.
- [3] Barry Arons. 1993. SpeechSkimmer: interactively skimming recorded speech. In *Proceedings of User Interface Software and Technology (UIST)*. ACM Press, 187–196.
- [4] Barry Arons. 1997. SpeechSkimmer: a system for interactively skimming recorded speech. *ACM Transactions on Computer-Human Interaction (TOCHI)* 4, 1 (1997), 3–38.
- [5] Aleksandr Chuklin, Aliaksei Severyn, Johanne R Trippas, Enrique Alfonseca, Hanna Silen, and Damiano Spina. 2019. Using audio transformations to improve comprehension in voice question answering. In *Proceedings of International Conference of the Cross-Language Evaluation Forum for European Languages (CLEF)*. Springer, 164–170.
- [6] Ido Guy. 2018. The Characteristics of Voice Search: Comparing Spoken with Typed-in Mobile Web Search Queries. *ACM Transactions on Information Systems (TOIS)* 36, 3 (2018), 30:1–30:28.

- [7] Jeff Huang, Ryen W. White, and Susan Dumais. 2011. No Clicks, No Problem: Using Cursor Movements to Understand and Improve Search. In *Proceedings of Conference on Human Factors in Computing Systems (CHI)*. 1225–1234.
- [8] Bernard J. Jansen and Amanda Spink. 2006. How are we searching the World Wide Web? A comparison of nine search engine transaction logs. *Information Processing & Management* 42, 1 (2006), 248–263.
- [9] Christine Murad, Cosmin Munteanu, Leigh Clark, and Benjamin R. Cowan. 2018. Design Guidelines for Hands-Free Speech Interaction. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct (Barcelona, Spain) (MobileHCI '18)*. Association for Computing Machinery, 269–276.
- [10] Richard Schaller, Morgan Harvey, and David Elsweiler. 2012. Out and About on Museums Night: Investigating Mobile Search Behaviour for Leisure Events. In *Searching4Fun workshop at ECIR2012*. 4 pages.
- [11] Craig Silverstein, Hannes Marais, Monika Henzinger, and Michael Moricz. 1999. Analysis of a Very Large Web Search Engine Query Log. *ACM SIGIR Forum* 33, 1 (1999), 6–12.
- [12] Damiano Spina, Johanne R. Trippas, Lawrence Cavedon, and Mark Sanderson. 2017. Extracting audio summaries to support effective spoken document search. *Journal of the Association for Information Science and Technology (JASIST)* 68, 9 (2017), 2101–2115.
- [13] Jaime Teevan, Daniel Ramage, and Merredith Ringel Morris. 2011. #TwitterSearch: A Comparison of Microblog Search and Web Search. In *Proceedings of Web Search and Data Mining (WSDM)*. 35–44.
- [14] Johanne R. Trippas, Damiano Spina, Lawrence Cavedon, Hideo Joho, and Mark Sanderson. 2018. Informing the Design of Spoken Conversational Search. In *Proceedings of Conference on Information Interaction and Retrieval (CHIIR)*. 32–41.
- [15] Johanne R. Trippas, Damiano Spina, Paul Thomas, Hideo Joho, Mark Sanderson, and Lawrence Cavedon. 2020. Towards a Model for Spoken Conversational Search. *Information Processing & Management* 57, 2 (2020), 1–19. <https://doi.org/10.1016/j.ipm.2019.102162>
- [16] Alexandra Vtyurina, Charles L. A. Clarke, Edith Law, Johanne R. Trippas, and Horatiu Bota. 2020. Comparison of Text and Audio Search Interfaces for Information Seeking Tasks with Varying Complexity. In *Proceedings of International Conference on the Theory of Information Retrieval (ICTIR)*. 61–68.
- [17] Alexandra Vtyurina, Adam Fourney, Meredith Ringel Morris, Leah Findlater, and Ryen W. White. 2019. VERSE: Bridging Screen Readers and Voice Assistants for Enhanced Eyes-Free Web Search. In *Proceedings of International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS)*. 414–426.
- [18] Ryen W. White. 2016. *Interactions with Search Systems*. Cambridge University Press.
- [19] Jeonghe Yi and Farzin Maghoul. 2011. Mobile search pattern evolution. In *Proceedings of World Wide Web Conference (WWW)*. 165–166.