Keelan Evanini, Eugene Tsuprun, Veronika Timpe-Laughlin, Vikram Ramanarayanan, Patrick Lange, David Suendermann-Oeft

Educational Testing Service, Princeton and San Francisco, USA

kevanini@ets.org, etsuprun@ets.org, vlaughlin@ets.org, vramanarayanan@ets.org, plange@ets.org, suendermann-oeft@ets.org

Evaluating the Impact of Local Context on CALL Applications Using Spoken Dialog Systems

Bio data

Keelan Evanini is a Research Director at Educational Testing Service and oversees research on automated assessment of non-native spoken language for large-scale assessments. He received his PhD in Linguistics from the University of Pennsylvania in 2009 and has worked at ETS Research since then.

Eugene Tsuprun is a Research Systems Specialist at the Cognitive, Accessibility, & Technology Sciences Center at Educational Testing Service. He works on developing voice user interfaces and front-end applications for the HALEF system. He holds an M.A. degree in English Language Learning from the University of Minnesota.

Veronika Timpe-Laughlin is Associate Research Scientist at the center for English Language Learning and Assessment at Educational Testing Service. Her main research foci include L2 pragmatics and interaction competence, task-based language teaching, and CALL. Prior to joining ETS in 2013, she worked and taught in the area of applied linguistics/TESOL at TU Dortmund University, Germany.

Vikram Ramanarayanan is a Research Scientist at Educational Testing Service's R&D division in San Francisco. Vikram's research interests lie in applying scientific knowledge to interdisciplinary engineering problems in speech, language and vision and in turn using engineering approaches to drive scientific understanding. He holds M.S and Ph.D degrees in Electrical Engineering from the University of Southern California.

Patrick Lange is an Associate Research Engineer in the R&D division at Educational Testing Service. His work at ETS is focused on building the infrastructure and systems for automated assessment of natural language. His main project is leading the engineering efforts of the open-source spoken dialog system HALEF. Patrick received his M.Sc. by Research degree in Computing Science from Staffordshire University, UK.

David Suendermann-Oeft is Research Director at Educational Testing Service heading the Dialog, Multimodal, and Speech (DIAMONDS) research center. David received a PhD in Electrical Engineering from the Bundeswehr University Munich in 2008 and has held leading positions in academia (e.g. DHBW Stuttgart) and industry (e.g. SpeechCycle, EMR.AI) since.

Abstract

The development of prototype Spoken Dialog Systems (SDS) for computer assisted language learning (CALL) applications has become easier in recent years due to dramatic improvements in the performance of automatic speech recognition (ASR) systems and the availability of open-source tools for the components of the SDS pipeline. Such SDS-based language learning prototypes have the potential to create an interactive, engaging language learning environment and to provide real-time, individualized feedback to learners. However, while an initial prototype SDS that successfully processes a limited range of expected learner responses can be developed rather quickly, the iterative refinement of the system to enable it to accurately respond to the wide range of learner responses—responses that differ due to students' first (L1) and/or second language (L2) proficiency, cultural backgrounds etc.—can be extremely time-consuming and challenging. This can limit the usefulness of SDS-based CALL applications to specific learner populations and contexts.

In this study, we attempt to address challenges related to the scalability and generalizability of SDS-based CALL applications through an analysis of common types of problems that these systems encounter when they are used in multiple contexts. Specifically, we used the open-source HALEF SDS framework to design interactive conversational tasks for L2 English learners for a wide range of situations such as ordering food in a restaurant, interviewing for a job, and disputing a bill. These conversational tasks were deployed on the Amazon Mechanical Turk crowdsourcing platform and conversational responses were collected from over 2,000 L2 speakers of English representing over 50 different L1 backgrounds. The responses were transcribed and analyzed to determine characteristics of learner responses that resulted in sub-optimal system behavior since they did not conform to the initial patterns that the prototypes were designed to handle. In this presentation, we will present a taxonomy of the types of variability that were observed due to L1 and local context, covering the following aspects of the learner's speech: grammar, vocabulary, pragmatics, and cultural knowledge. These findings will help establish best practices for developing SDS-based CALL applications that can more accurately process a wider range of responses due to differences in learner characteristics and can therefore be applied in a broader set of educational environments. We will also discuss strategies that can be used by SDS designers to make the applications more robust to variability.

Conference paper

1. Introduction

With the continued increasing use of English in global workforce and academic settings, the need for effective and scalable methods for helping English learners improve their English speaking proficiency continues to grow. One approach that is becoming more viable through recent improvements in automated speech recognition and artificial intelligence technology is the use of interactive spoken dialog systems (SDS) in computer assisted language learning (CALL) applications. These SDS-based CALL applications provide an environment for the learner to practice interactive speaking tasks without the physical presence of an English instructor and receive feedback at any time during the day, and can therefore be very helpful for English learners, especially ones with relatively lower proficiency (since the SDS-based tasks that currently work best given the state-of-the-art technology are constrained and do not elicit open-ended complex speech). However, the performance of an SDS-based CALL application can be hindered due to variations in the spoken responses that are provided by learners across different contexts, for example, different L1 backgrounds, different countries, and different proficiency levels. While it is not possible for

an automated system to account for all of these types of variation, many of them can be handled in an efficient and structured manner.

In this paper, we first briefly describe an instantiation of a SDS-based CALL system that was used in a large-scale global data collection with English learners representing a range of different contexts. Then, we describe the main types of variation that can be problematic for the system (including differences in vocabulary, grammar, pragmatics, and cultural knowledge) and illustrate the approaches that can be taken by SDS task developers and engineers to account for the variation.

2. SDS System and Data Collection

The SDS CALL tasks were developed using the HALEF system framework (Ramanarayanan, et al., 2017). HALEF is an open-source, standards-compliant, cloud-based, modular SDS architecture that has been used to develop a variety of interactive educational applications. Figure 1 provides a high-level illustration of the main components of an SDS system.

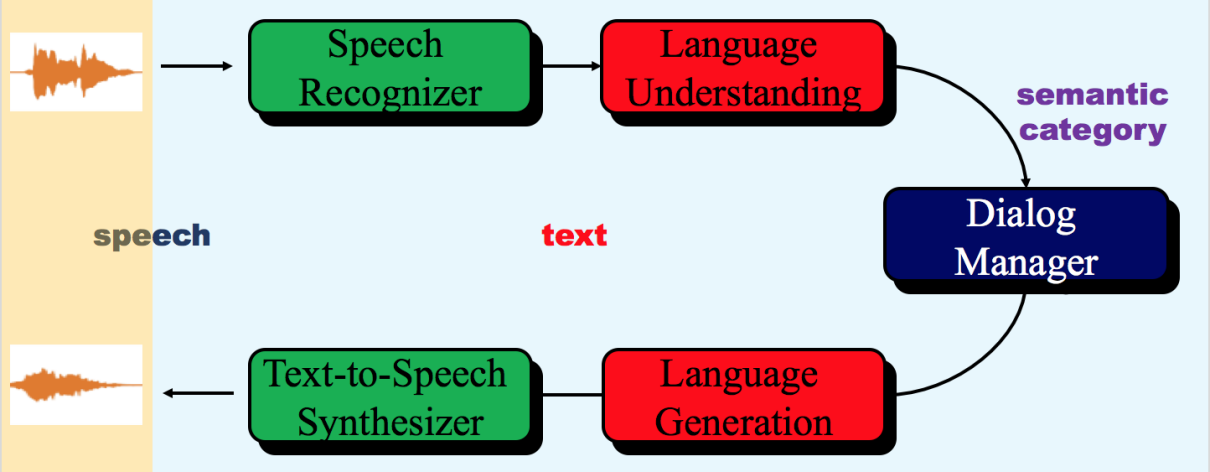

Figure 1. High-level overview of the main components of an SDS system

As shown in Figure 1, the user of an SDS system (in this case, the language learner) provides a spoken response that is first converted to text using an automatic speech recognition (ASR) system. While ASR accuracy has improved dramatically in recent years, the process can still introduce errors, especially in cases that are more challenging for state-of-the-art ASR systems, such as non-native speech. The output of the speech recognizer is the processed by a language understanding component that parses the text to assign it to a semantic category from a list of pre-defined semantic categories that are relevant for the current dialog state, i.e., system prompt. Based on this automatically detected semantic category, the dialog manager determines the next action that should be taken by the SDS. Then, the language generation component produces the text that will be contained in the next system prompt, the text-to-speech synthesizer converts this text to an audio response that is played to the user, and the cycle repeats. In practice, when the number of possible system responses is limited, the language generation component can simply select from a list of pre-specified system responses (instead of generating text automatically on-the-fly) and the system can use pre-recorded audio files prepared by human voice talents instead of text-to-speech synthesis; these two approaches were used in the current system, since they typically provide a more natural user experience.

For this study, we analyzed responses provided in the context of a task designed to help language learners practice placing orders in a coffee shop.[6] In this task, English learners are presented with a menu that includes the following items: coffee, cappuccino, latte, mocha, tea, bagel, and croissant. The following instructions are printed on the screen:

The boss asked you to pick up her breakfast on your way in to work. She wants one drink and one food item.

The system starts out the conversation with the following prompt: *Hello, welcome to the Coffee Spot. What can I get for you today?* Then, after recognizing which item(s) the learner selected, the system asks follow-up questions about the order, such as *Would you like that coffee hot or iced?* and *Would you like that bagel toasted?* After completing the follow-up questions, the system asks if the learner has completed the order (*Would you like anything else?*). Finally, when the learner's order is complete, the system presents the bill to the learner and asks how they would like to pay for their order. After the conversation is complete, the system can provide task completion feedback to the learner, i.e., whether the learner successfully ordered one drink and one food item. Figure 2 presents an example flowchart (developed using the OpenVXML[7] design tool) for the Coffee Shop task that specifies the system prompts at each dialog state as well as the conversational branches taken by the system for each category of learner response (some details have been omitted from the full version of the task for clarity).

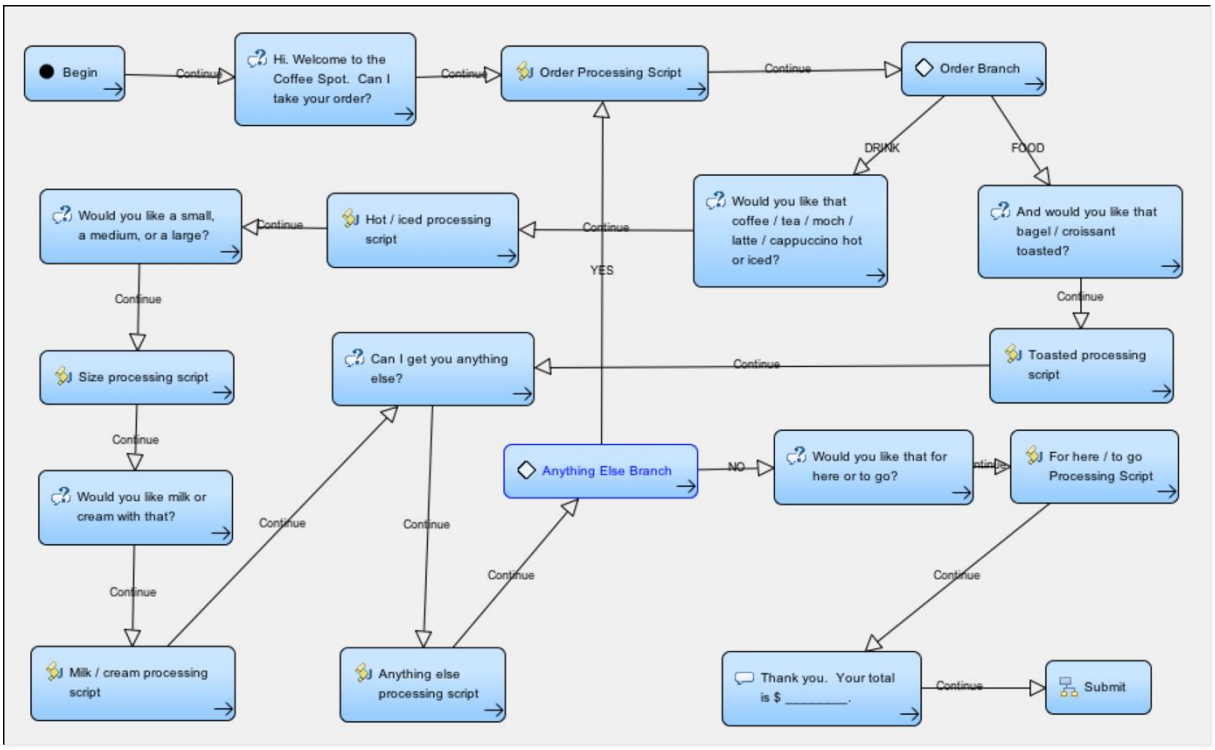

Figure 2. OpenVXML flowchart for the Coffee Shop interactive SDS CALL task specifying the system's behavior for each category of learner response at each dialog turn.

This task is relatively restricted in nature, since the main vocabulary that the learner is expected to use in each response is contained either in the stimulus materials (the menu) or

[6] A sample version of this task is available at http://englishtasks.org.
[7] https://github.com/OpenMethods/OpenVXML

the system prompts, and the learners typically provide responses that range in length from a single word to a short sentence. The task is designed to provide beginning learners an opportunity to practice basic vocabulary and grammatical structures required for transactional interactions as well as appropriate politeness strategies.

The Coffee Shop SDS CALL task was deployed to the Amazon Mechanical Turk crowdsourcing platform[8] in order to obtain a large number of responses that can be used to tune the language models used by the ASR system and revise the semantic categories that are recognized by the system and the branching paths in the conversation. The data considered in this study consists of 7,345 utterances provided in 849 different conversations that non-native speakers conducted with the Coffee Shop task through the Mechanical Turk platform between June 2016 and February 2017. The English learners in this sample are represented by a total of 52 different L1s; the 10 most frequent L1s in this sample are shown in Table 1 along with the number of learners for each.

L1	Number of Mechanical Turk Participants
Hindi	222
Spanish	115
Tamil	113
Telugu	78
Malayalam	64
Portuguese	31
Gujarati	18
Marathi	14
French	14
Urdu	12
Other	168

Table 1. Distribution of the L1 backgrounds of the Amazon Mechanical Turk participants who interacted with the Coffee Shop SDS CALL task

As Table 1 shows, a substantial percentage of the participants are from India, which is consistent with the demographics of the overall Amazon Mechanical Turk population. However, many other L1s are represented in this data set as well, providing ample opportunities to study variations across the learner responses due to differences in local context.

3. Designing an SDS CALL Application to Handle Sources of Variability Based on Local Context

In this section, we provide a taxonomy of several sources of variability that can arise when deploying an SDS-based CALL application in diverse contexts, explain why the sources of variability can be problematic for the SDS application, and discuss common approaches for enabling the SDS to be robust to these types of variability.

3.1 Vocabulary

One source of variability that can pose difficulties for an SDS-based CALL application is when learners from different backgrounds provide responses that use unexpected vocabulary. An example of this can be found in the range of responses provided by

[8] https://www.mturk.com/mturk/welcome

different learners to the following system prompt (the L1 of the learner who provided each sample response is indicated in parentheses after the response):

SDS prompt: *Would you like that for here or to go?*
Responses: *Make it take away.* (Kannada)
Hi, I want some, some coffee to take away. (Spanish)
It's on a go. I I need it as a parcel. (Hindi)

Based on the task designer's initial expectation of the types of responses that learners would provide to this question, the SDS application was originally deployed with a rule-based language understanding component that recognized the following key words and phrases for the two semantic categories:

HERE: *here, stay*
TO_GO: *to go, carry out, take out*

The sample responses listed above are problematic, since the semantic category clearly should be TO_GO, but learners used variable ways of expressing this that were not initially expected. Since these responses do not contain any of the key words that the natural language understanding component recognizes for this category, the value for this variable (whether the order should be for here or to go) is undefined, and the SDS system would take a default action in order to continue the conversation (such as reprompting for the requested information or continuing on to the next prompt).

The most effective way for addressing this type of local variation in vocabulary usage is to systematically transcribe and provide semantic annotations for a large number of responses in order to capture as many of the different variants as possible. Then, SDS system designers can use the annotated responses to develop additional key words and phrases for the semantic categories for a rule-based natural language understanding module. In this case, the phrases *take away* and *as a parcel* could be added to the list for the TO_GO semantic category. In addition, these annotated responses can be used by system designers to develop statistical natural language understanding models, which are typically more robust than rule-based approaches when a sufficient amount of responses have been annotated to represent the most likely variations for each dialog state (Suendermann et al., 2009).

The following two conversation snippets provide examples of lexis that were not anticipated initially by the system designers but that can be handled by the system after they have been added to the appropriate semantic categories, i.e., *cool* should correspond to the same category as *iced* and *big* should correspond to the same category as *large*.

SDS prompt: *Would you like that coffee hot or iced?*
Responses: *Uh a cool one.* (Saurashtra)

SDS prompt: *Would you like a small, a medium, or a large?*
Responses: *I like a big.* (Hindi)

Some instances of observed vocabulary variation among speakers with different L1s can clearly be ascribed to different local norms; for example, *take away* is more common in varieties of English that are influenced by British English, and *as a parcel* is common in India. On the other hand, other instances of vocabulary variation may be due to learners' incomplete knowledge of the target English vocabulary which causes them to use a semantically-related word that they are more familiar with or that is easier to acquire based

on L1 transfer (e.g., a cognate); the response with the use of *cool* instead of *iced* above may be an example of this. Either way, the SDS system can be made more robust to these types of vocabulary variation through an iterative process of transcribing responses, annotating their semantic categories, updating the natural language understanding component of the SDS, and redeploying the CALL application to collect additional responses.

3.2 Grammar

Responses provided by English learners to the Coffee Shop SDS-based CALL task also exhibited a wide range of grammatical variation due to the fact that the learners represented many different L1 backgrounds and proficiency levels. The responses provide a few examples of unexpected grammatical patterns:

SDS prompt: *Hi. Welcome to the Coffee Spot. What can I get for you?*
Responses: *I want beverage mocha and food item bagel. (Punjabi)*
I want bagel food. (Tamil)
I like tea. (Telugu)
I want the coffee. (Saurashtra)
Hello uh, I want to a cappuccino. (Spanish)

While these types of responses contain unexpected grammatical patterns due, primarily, to L2 errors, they generally do not need to be explicitly handled by the SDS application. Since the learner's intended communicative goal is apparent in each case through the presence of expected key words (e.g., *mocha*, *bagel*, *tea*, *coffee*, *cappuccino*) the SDS system can be designed to successfully recognize the appropriate semantic category and continue with the conversation. In this case, the solution for making the system more robust to difference among learner responses due to different L1 and proficiency backgrounds is underspecification; i.e., the SDS system identifies the semantic category of the response based on a generic key word, such as *coffee*, rather than on a more specific phrase, such as *a coffee* or *have a coffee*. This is in contrast to the solution discussed for vocabulary variation in Section 3.1, which was to add additional knowledge about potential vocabulary variation to the SDS natural language understanding models. In the case of grammar, an SDS designer is only required to explicitly include information about variations into the system's models when the grammatical form is part of the targeted construct for the speaking task. For example, if the task was being used by a language instructor as part of a lesson on English article usage, then the system would need to be designed to recognize the differences between responses such as *I would like bagel.* and *I would like a bagel.* This would be done by following the same approach outlined in Section 3.1 for vocabulary: transcription, annotation, and redesign of the natural language understanding component of the SDS.

3.3 Pragmatics

The following responses demonstrate responses that are not pragmatically appropriate, since they do not use politeness strategies that would be expected in this conversational situation with an employee at a coffee shop.

SDS prompt: *Hi. Welcome to the Coffee Spot. What can I get for you?*
Responses: *Give me a bagel. A bagel.* (Malayalam)
I need some coffee. (Hindi)
I want tea. (Bengali)

If an SDS system is initially designed to look for specific request strategies, such as *I would like a* _____ or *Can I get a bagel?*, then it may not be able to process pragmatically

inappropriate responses like the ones featured above. In this case, underspecification (i.e., detecting key words) is again a potential solution, especially for an initial version of the system, unless the system is intended to be used to provide feedback to learners about the pragmatic appropriateness of their responses.

The following examples demonstrate a different type of variation in pragmatics:

SDS prompt: *Hi. Welcome to the Coffee Spot. What can I get for you?*
Responses: *Surprise me.* (Spanish)
Uh, surprise me. (Malayalam)
Uh good morning uh what's your specialty in this store? (Mandarin)
I want something uh very special for her. So I want to give them uh sort of a surprise to her. So what would you suggest? (Gujarati)

In these instances, the responses provided by the learners were unexpected since they didn't provide a direct answer to the question, i.e., they didn't include an order for a food or drink item. In order to adapt the CALL application to successfully process these types of responses instead of taking the default action for unrecognized responses (such as reprompting), the SDS designer would need to add an additional semantic category for the natural language understanding component to detect and add an additional branch out of this dialog state corresponding to the new semantic category so that the dialog manager could take a different action, such as providing the following prompt: *I would recommend our cappuccino, since our store just purchased a new, top-of-the-line cappuccino maker. How does that sound?*

3.4 Cultural Knowledge

A final source of variation based on different local contexts that can be problematic for SDS-based CALL applications is caused by different cultural practices and expectations that are relevant for the tasks. For example, consider the following three responses provided by English learners in India:

SDS prompt: *And would you like that bagel toasted?*
Responses: *Of course toasted.* (Malayalam)
Yeah, I do, absolutely. (Hindi)
Yeah, toasted. Extremely high. (Telugu)

These responses were not initially expected by the designers of the application, since they were not aware that it is very uncommon in India to sell untoasted bread-like products in cafes and restaurants. In order to provide a more natural conversational experience for English learners in this local context, the SDS task would need to be redesigned without this prompt. Alternatively, the original version could be retained for general use and a context-specific one could be developed for use in India that excludes this prompt (or that includes a revised version of the prompt, such as *You'd like the bagel toasted, right?*). However, this approach of developing different versions of the conversation flow for different local contexts can quickly become untenable for a CALL application that is targeting a global market, due to the large number of variants that would be required. A more feasible approach is to use this knowledge of local variation to design a generic version of the application that is robust to these differences and can therefore be applied in a wide range of different local contexts. Additionally, these dialogues also convey types of culture-specific knowledge in language use insofar the learners progress through the routinized and highly scripted exchange that typically takes place in a U.S. coffee shop, thus learning about these exchanges and how they take place in a given English speaking context.

4. Conclusion

In this paper we have presented examples of variation in responses to an SDS-based CALL application that we observed in a large-scale crowdsourcing data collection with English learners from 52 different L1 backgrounds. Variations in vocabulary, grammar, pragmatics, and cultural knowledge caused by different L1 backgrounds, proficiency levels, locations, etc. can lead to responses that were not expected by the SDS designers when the tasks were initially developed. Several different approaches can be taken by system designers to enable an SDS-based CALL application to process these responses successfully (including underspecification, development of more comprehensive natural language understanding models based on semantic annotations, prompt redesign, and inclusion of additional branches). Due to the wide range of variation across different contexts, it is never possible to build an application that accounts for all possible responses in an SDS-based CALL application; however, through an iterative application of these approaches for different learner populations, it is possible to develop applications that are robust to most sources of variation.

CALL in Context

This contribution directly addresses the following question posed by the Call in Context call for papers:

How can/should we detect and formulate to what extent learners and teachers are different?

For this study, we implemented a series of interactive, conversational CALL tasks based on Spoken Dialog Systems (SDS) that were designed to help learners of English improve their speaking skills. Focusing on a single task that provides learners practice with placing an order in a coffee shop, we analyzed a large number or responses provided by learners from 52 different native languages (L1s). This analysis demonstrated that variation in the types of responses provided by learners across different L1s due to differences in vocabulary, grammar, pragmatics, and cultural knowledge can lead to suboptimal performance of the SDS-based CALL application. It is necessary for the designer of an SDS-based CALL application to understand all of these types of variation based on local context in order to develop a robust system; however, in some cases, the most appropriate approach is to apply underspecification so that the system, in effect, ignores the context-driven variants. The decision about whether these variants should be detected by the system or not depends on the specific construct of spoken language proficiency that the CALL task is intended to target.

References

Suendermann, D., Evanini, K., Liscombe, J., Hunter, P., Dayanidhi, K., and Pieraccini, R. (2009). From rule-based to statistical grammars: Continuous improvement of large-scale spoken dialog systems. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Taipei, Taiwan, April 2009, pp. 4713-4716.

Ramanarayanan, V., Suendermann-Oeft, D., Lange, P., Mundkowsky, R., Ivanov, A.V., Yu, Z., Qian, Y., and Evanini, K. (2017), Assembling the jigsaw: How multiple open standards are synergistically combined in the HALEF multimodal dialog system. In *Multimodal*

Interaction with W3C Standards: Towards Natural User Interfaces to Everything, D. A. Dahl, ed. New York: Springer, 2017, pp. 295-310.