Challenges and Methods in Design of Domain-Specific Voice Assistants

Sarah Mennicken, Ruth Brillman, Jennifer Thom, Henriette Cramer
Spotify
{sarahm, brillman, jennthom, henriette}@spotify.com

Abstract

Most of the currently existing voice assistants, like Alexa, Siri, Google Assistant, and Cortana, are generalists. They act as a unifying voice interface to a myriad of controls but rarely support domain-specific expert functionalities. There are efforts to provide more targeted assistant experiences and capabilities around specific areas of applications. In this paper, we discuss several challenges and opportunities in the design of domain-specific voice assistants. We outline a variety of methods to create and utilize an understanding of domain-specific user language and ideas to prototype and study the envisioned user experiences.

Introduction

Amazon's Alexa, Apple's Siri, Google's Assistant and Google Home and Cortana are well-known examples of general-purpose assistants created with the expertise and data available to major tech companies. In this paper, we give a high-level overview of a variety of design challenges, and make the distinction between designing for a general-purpose assistant as opposed to a domain-specific one. By domain, we mean the types of expertise handled by the assistant. A general-purpose assistant, such as Alexa, Google Assistant, and Siri, works across domains such as providing the user with weather information, setting timers and reminders, driving directions and shopping. General-purpose assistants, by necessity, must cover a broad and wide territory of expertise. A domain-specific assistant is a specialist in one particular area, such as a customer service agent like Nuance's Nina (Nuance Press Release 2017) on Alexa, a banking agent or a music service providing personalized music experiences.

It is worth noting that the challenges discussed in this paper apply to many domain-specific assistants, regardless of the machine learning models that power them. Consider an automatic speech recognition (ASR) component: the ASR will likely have to be optimized to correctly transcribe important, domain-specific words, accounting for differences in accents and possible mispronunciations. However, this challenge will present itself regardless of which machine learning techniques are used. While the specific way to implement a solution to a given challenge may depend on underlying techniques and modeling decisions, the occurrence of the challenge should not be.

A challenge for general-purpose voice assistants is that they need a wide breadth of data. This can include audio data, transcribed text, annotated and labeled text for natural language understanding and knowledge graph inputs. Domain-specific assistants, however, come with different expectations, and require a narrower and simultaneously deeper dataset for training and testing. In this paper, we discuss considerations on how user data can be leveraged to identify what aspects to consider for data collection and how to drive prototyping efforts for the efficient transfer of insights to models and technology.

What makes a voice assistant?

From a technical perspective, a voice assistant is a natural language processing pipeline. It consists of many parts, including automated speech recognition (ASR), natural language understanding (NLU), natural language generation (NLG), and text-to-speech (TTS). It can include search, knowledge graph and agent back-ends, as well as agents of different platforms, all of which have to interface with different natural language components. From a user perspective, design has to consider the expectations the user has around how s/he can phrase her questions to the assistant, the functionality it offers, and how it sounds when it responds. This includes the words the assistant chooses and the sound of its voice. The design and functionality choices will affect how users continue to interact with it, similar to how the voice and the vocabulary of another human affect how someone interacts with them. E.g.,

if one hears the voice of a child using the vocabulary of a 5-year-old, they will adjust their expectations and their own language. People always attribute personality traits to speech, even if it is synthesized by a computer (Nass and Lee 2001). Therefore, it is an important first step for design - even before leveraging user data - to define the role the assistant should convey and a few, core personality traits. Whether the role is to be a representative of a brand or an individual with their own opinions and values will affect how interactions need to be designed. Identifying the target user helps create a user-centered plan for design. E.g., a music companion for teenagers will require a different approach than an assistant for medical support for the elderly.

Domain-specific behavior and expectations

With the role and the target user group in mind, we can create a better understanding of the domain. This helps to anticipate the voice input the assistant will receive and provides insights on expectations. This includes not just expectations for functionality, but also for the tone of voice and the behavior of the assistant. What questions help to guide this process and what data can be leveraged?

How do people talk in the domain that needs to be modeled for a domain-specific assistant?

Examining existing data from **other systems in the same domain** is a useful, though often times not comprehensive, method of understanding what kinds of voice requests the system is likely to receive. For example, text search interfaces often compel users to search for named entities. However, voice requests can often be nonspecific or generic, such as asking a TV assistant "Play a dark and gritty documentary" or "Show me something my friends will like." Understanding the way that people talk about the target domain is a necessary first step to predict and prepare for the types of voice-specific utterances the system will need to be able to process. Similarly, back and forth dialogues that are crucial in domains such as customer service, cannot per se be derived from non-dialogue, search-type data.

Voice assistants often take on roles that are inspired by existing human roles or tasks, including trying to replicate their domain-specific knowledge, e.g. a travel, or customer service, agent. Both **content analysis**, as well as qualitative design research methods, like **interviewing domain experts** can provide a more comprehensive picture of what utterances to expect or what functionalities to include. E.g., asking experts which questions they are asked by their audience or which questions they would like to ask from individual users, but cannot scale. Dialogues are crucial to understand in fields like customer service, in which case in-depth content analysis of existing interactions is also vital. Creating this understanding of the role the human assistant takes on can help to identify interaction flows. This approach has also been successfully applied for years in the context of information retrieval, e.g., to identify information seeking behavior at a library (Taylor 1968). **Crowdsourcing** can be used in multiple ways, and is a common element of voice projects. First, it is a useful method to elicit large amounts of data to bootstrap natural language understanding systems (Callison-Burch and Dredze 2010). Data can be collected from a variety of crowdworkers from various geographies and domain-related skillsets to increase diversity of training data. Second, it can be used to collect speech data from a diverse population so that a broadly applicable ASR system can be trained and developed (Pavlick et al. 2014). Finally, crowdworkers can label data for supervised machine learning methods and therefore improve existing models. A better understanding of the domain will also help to put user utterances into context. E.g., certain user utterances that seem offensive might be sincere requests in the context of music and entertainment. Content like the song "F*** you" by CeeLo Green or the TV series "I love D***" illustrates this potential ambiguity well. Culturally specific references carry the potential for this ambiguity, too. If new entities are regularly added to content catalogues where popularity fluctuations are frequent, this becomes even more challenging.

What behavior do people expect?

People might have built up expectations from experiences with people in the roles that the assistant is intended to take on, including what the assistant should be capable of, the tone of its interactions, its demeanor, or even how it looks. E.g., consider the stereotypical differences in how people think a travel agent, a bank teller, or a DJ might behave or appear. Of course, the previously mentioned interviews with domain experts provide insights into this, too.

Another way to elicit understanding via qualitative design research is to ask participants to **role play**. Role-playing through Wizard-of-Oz set-ups can identify whether a scripted dialogue works and a more open-ended setup can help identify potential functional challenges. Pretend-users might come up with requests that the Wizard-of-Oz prototype may not be able to solve. Take for the hypothetical example of a restaurant recommendation assistant. The pretend-user might want to "Send that restaurant to my friend Frank". This could lead to the realization there is no script for sending recommendations to unknown friends. Maybe, the pretend-user would want to "Order me a pizza". Potentially, being outside the originally envisioned functionality, that might point to a missed opportunity and/or required features, like having to have credit card

information on file, and the need to integrate a secure payment partner.

Identifying domain-specific challenges

While people might only expect domain-specific services from a human expert assistant, we cannot necessarily make that assumption for domain-specific voice assistants. One might expect such an assistant to be able to navigate, find music, or even order pizza. Therefore, creating a good understanding what the user expects within a specific domain might provide an initial assumption on what variety of utterances to expect and then to define how you want to deal with the functional limitations of your system.

Functions and knowledge

Deciding how to limit the scope of the NLU/NLG system is a particular problem for domain-specific assistants. For example, while playful questions such as "Are you married?" happen in general assistant contexts, they are less expected for most domain-specific voice assistants. Domain-specific assistants require design and engineering decisions about how and where to limit conversation and how to distinguish erroneous and out of range requests, both of which are potentially unsupported by the machine learning model underlying the assistant. There are several options. The assistant can respond in a way that shapes expectations moving forward ("Sorry, I can't help you with that.") at the risk of being perceived as incomplete or less competent. The assistant can use an unsupported utterance as an opportunity to educate the user about what it can do ("No can do. But I can sure be of assistance if you want to book a flight.").

The design decisions above also open up broader questions about the nature of conversations. For example, if the semantic processing component of an NLP system depends on a knowledge graph, it needs to be decided how to limit its scope. Similar questions arise when building out dialogue management systems, regarding what facets of conversation the assistant should and should not support, such as multiple turn question answering sessions.

Pronunciation

Since users might expect domain-specific assistants to have deep expertise in that domain, unique and little-known terms that do not occur that frequently in general natural language corpora will need to be modeled. Unique terms and pronunciations are not always easily covered by off-the-shelf lexicons. Some domains include entities for which full names are not originally intended to be pronounced, such as emoji in text and email messages. For instance, music systems will have to handle a diverse catalog with unique artist and track names. User-generated music playlists can have names that consist of emojis (Spotify Blog 2017) or make use of character substitutions ($ for S) that might not correspond to obvious pronunciations. Non-obvious and ambiguous pronunciations pose a challenge for ASR systems, and their detection may require dedicated new techniques.

In the music domain, code-switching between languages occurs when users ask to listen to music in multiple languages in addition to their primary language (e.g. "Play *Me gustas tu*"). This poses another challenge for ASR. In the travel domain, an assistant that supports international travel will likely have to train its ASR on more than one pronunciation for international destinations (e.g., the English and Spanish pronunciation of cities in Latin America), and understand that multiple names, across multiple languages, refer to the same location.

Privacy

Hands-free voice assistants also face particular design challenges surrounding confidential information, especially if the assistant is developed for a domain where privacy is highly prioritized, like banking or healthcare. Password controls are challenges for all assistants, but financial assistants may face greater challenges surrounding information such as bank account numbers and sensitive social information such as account balances.

Default behavior

Different assistants also trigger assumptions of a default action on the part of the user. While many utterances contain a verb, some utterances are simply the name of an entity the user would like to search for, similar to text searches. For example, instead of saying "Play David Bowie," they may just say "David Bowie." If a user is interacting with a music assistant, these utterances should probably result in a David Bowie album being played. However, on a movie assistant, this utterance may result in the user watching the movie *Labyrinth.* This is different from what a general voice assistant would return; all current general voice assistants that have been brought to market support a general search as their default action. The fact that a user might reasonably expect identical utterances to result in distinct content across different types of assistants poses challenges for assistant design and user research. This also influences the type of linguistic utterance data the model should be trained on and expect to receive.

Prototyping tangible experiences

Depending on the domain and the domain-specific challenges there are easy ways to prototype early on to inform further iterations and refinements.

A quick way to test an envisioned interaction is working with writers who are experienced in writing dialogue and then ask participants to provide feedback. However, this method put participants into a passive position where they act more as an observer, rather than being immersed themselves. **Scripted and pre-recorded dialogues** can alleviate this to some extent. Asking participants to read out the requests or questions that have been identified as common for the domain and then present them with pre-recorded audio will create at least some level of immersion.

Off-the-shelf conversational prototyping tools, such as Alexa Skills or Google Actions, are simple software toolkits for the commercially available hardware Amazon Echo and Google Home. They provide a lightweight way to prototype a dialogue experience for a wide general audience. The main benefit of this type of prototype is a relatively easy setup. However, these platforms are not fully customizable and will not allow designers to model the depth needed for realistic interactions with a domain-specific assistant. They also do not allow full access to the user utterances and speech data that is collected by the hardware which might be required for the prototyping of the envisioned functionalities.

Custom-prototyping tools such as **Wizard-of-Oz tools** for rapid prototyping and testing are widely known in research, but quick-and-easy tools are not yet easily accessible to industry product teams. Oftentimes, a lot of custom work is required to implement such prototypes. Active research is, for example, ongoing in developing in-car voice interfaces (Martelaro and Ju 2017). The recent attention to the 'fake autonomous car' (Solon 2017) in which prototyping involved someone dressing up as a car seat inspired by the Stanford Ghost Rider set-up (Rothenbücher et al. 2016) is a testament to the offbeat creativity still necessary in testing people's reactions to new applications.

Integrating Machine Learning into Design Prototyping Tools

Making the compelling collaboration between careful interface design and the capabilities of machine learning tangible for user testing is quite challenging. Rapid iteration and prototyping of experiences is a vital process of the design process for voice assistants, but design prototyping tools often do not include the functionalities enabled by machine learning, e.g., personalized or context-aware context. When users are instead presented with data prepared ahead of time or if the interaction lacks personalization, it reduces how representative these studies are for the envisioned experience. The value of experiencing a prototype which includes the models being worked on is therefore significant for end-user testing and to inform iterative design.

Integrated prototyping also allows for a better understanding how design decision result in technical implications. If machine learning-based functionality is a part of the design prototypes, it provides an opportunity to learn about potential errors and edge cases early. User studies with such integrated tools can provide insights into possible limitations that might occur when being used in a real or slightly different context than the one for which the models have been trained for.

In Short

When designing for domain-specific voice assistants, there are many ways to learn from how users and people in roles similar to the assistant naturally speak within that domain and what their expectations are.

Domain-specific behavior and expectations
- Analyze existing behavior data
- Crowdsource data selection
- Interview domain experts
- Use role play in user studies

Domain-specific challenges
- Functions and knowledge of assistant
- Pronunciation of domain vocabulary
- Privacy of application behavior
- Expected default behavior

Prototyping tangible experiences
- Scripted and pre-recorded dialogues
- Off-the-shelf conversational prototyping tools
- Wizard-of-Oz tools

This will help to identify unique challenges that affect what the assistants should be capable of early on and allow for informed design decisions to deal with functional limitations. By including the functionalities enabled by machine learning in prototypes early on will allow collecting more representative user data while also informing and testing machine learning models.

Biographies

Sarah Mennicken

I am a research scientist as Spotify focusing on the design of voice output and novel voice experiences. I work at the intersection of user research, design, and technology helping to translate insights and designs into prototypes that allow studying tangible experiences.

Prior, I was a senior UX scientist/strategist at a startup and a visiting researcher at Microsoft Research focusing on interactive technologies based on real-time computer vision, applied machine learning, and sensing. My academic background and longstanding interest lie in user-centric experiences and interaction design for automated systems and agents, especially in the domestic context.

Ruth Brillman

I'm a research scientist at Spotify focusing on the human side of machine learning with a particular focus on voice systems, natural language processing and the relationships between NLP systems and the linguistic data they rely on for training. I graduated from MIT with a PhD in Linguistics in 2017, and have also done research and development work at Amazon and Akamai.

Jennifer Thom

I am a sr. research scientist at Spotify also focusing on the human side of machine learning and in particular, the social and collaborative aspects of the work conducted by those who label and collect the data that underlie these systems. I'm also interested in conversational interfaces and the social aspects of dialogue between humans and machines.

Previously, I was a research scientist at Amazon where I used various crowdsourcing techniques to provide data to improve the machine learning models that power the Alexa assistant and investigated informal question-answer behavior while a research scientist at IBM Research.

Henriette Cramer

I'm a research lead at Spotify, where I focus on the dialogue between people, data and machines. Prior, I researched user engagement at Yahoo and was a researcher at the Mobile Life Centre in Stockholm where I led projects on human-robot interaction and location-based services. My academic background is in people's responses to adaptive and autonomous systems.

References

Callison-Burch, C. and Dredze, M., 2010, June. Creating speech and language data with Amazon's Mechanical Turk. In Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk (pp. 1-12). Association for Computational Linguistics.

Nuance Press Release., 'Nuance Introduces Nina for Amazon Alexa, First Enterprise Virtual Assistant for the Smart Home', June 1, 2017, https://www.nuance.com/about-us/newsroom/press-releases/nuance-nina-for-amazon-alexa.html

Martelaro, N. and Ju, W., 2017. DJ Bot: Needfinding Machines for Improved Music Recommendations. *AAAI Spring Symposium '17, UX of ML workshop.*

Nass, C. and Lee, K.M., 2001. Does computer-synthesized speech manifest personality? Experimental tests of recognition, similarity-attraction, and consistency-attraction. *Journal of experimental psychology: applied*, *7*(3), p.171.

Pavlick, E., Post, M., Irvine, A., Kachaev, D. and Callison-Burch, C., 2014. The language demographics of amazon mechanical turk. *Transactions of the Association for Computational Linguistics*, *2*, pp.79-92.

Rothenbücher, D., Li, J., Sirkin, D., Mok, B. and Ju, W., 2016, August. Ghost driver: A field study investigating the interaction between pedestrians and driverless vehicles. In *Robot and Human Interactive Communication (RO-MAN), 2016 25th IEEE International Symposium* on (pp. 795-802). IEEE. Vancouver

Spotify Blog, 'Iconic Playlists: What Emoji Say About Music', May 2, 2017, https://insights.spotify.com/us/2017/05/02/spotify-emoji-music/

Solon, Olivia, 'Why did Ford build a 'fake driverless car' using a man dressed as a seat?, September 15, 2017, https://www.theguardian.com/technology/2017/sep/15/self-driving-car-fake-ford-virginia-tech-man-in-seat

Taylor, R. S. (1968). Question-Negotiation and Information Seeking in Libraries. College & Research Libraries, 29(3), 178-194.