Received: 19 November 2020 | Accepted: 25 January 2021

DOI: 10.1002/aaai.12059

SPECIAL TOPIC ARTICLE

Conversational recommendation: A grand AI challenge

Dietmar Jannach[1] | **Li Chen**[2]

[1]University of Klagenfurt, Klagenfurt, Austria

[2]Hong Kong Baptist University, Hong Kong, China

Correspondence
Li Chen, Hong Kong Baptist University, Hong Kong, China.
Email: lichen@comp.hkbu.edu.hk

Abstract

Animated avatars, which look and talk like humans, are iconic visions of the future of AI-powered systems. Through many sci-fi movies, we are acquainted with the idea of speaking to such virtual personalities as if they were humans. Today, we talk more and more to machines like Apple's Siri, for example, to ask them for the weather forecast. However, when asked for recommendations, for example, for a restaurant to go to, the limitations of such devices quickly become obvious. They do not engage in a conversation to find out what we might prefer, they often do not provide explanations for what they recommend, and they may have difficulties remembering what was said 1 min earlier. Conversational recommender systems (CRS) promise to address these limitations. In this paper, we review existing approaches to building such systems, which developments we observe today, which challenges are still open and why the development of conversational recommenders represents one of the next grand challenges of AI.

INTRODUCTION

We are increasingly used to interact with machines in natural language. We ask Siri to find us a nearby Italian restaurant, we request Alexa to play some music, and we even talk to our cars, instructing them to route us to our destination. Many of us have also interacted with chatbots, which are used by companies as a first contact point for customer service. Being able to interact with machines that are able to converse like humans has long been an iconic vision of the future in sci-fi movies. With the recent advances in natural language processing (NLP), the availability of huge pretrained language models like GPT-3, and the progress of machine learning in general, one may, therefore, think we are already close to achieving this vision.

Looking closer at the above-mentioned interactions, we observe that the devices we talk to are often good at reacting to individual commands. However, these systems sometimes reach their limits in situations when a multistep conversation would be required to achieve a task. A typical example is the reservation of a flight after discussing different options with a travel agent. Another common problem setting, where typically more than a one-shot interaction is needed, is that of recommending something based on individual short-term user preferences. This process is commonly referred to as *conversational recommendation*, cf. (Jannach et al. 2021), and such problems are the focus of this paper.

Conversational recommendation is in several ways different from the conventional, noninteractive presentation of item suggestions as found, for example, on the landing page of Netflix. First, while conventional recommendations rely on *push* communication, conversational recommender systems (CRS) support multi-turn and mixed-initiative interaction patterns. Moreover, in particular, natural language-based CRS combine aspects of *recommendation* and *search*, that is, they allow users to make queries. Generally, CRS in many cases target at problem settings where no long-term user profiles are available and where user requirements and the user's context are interactively acquired.

Interactive and conversational recommendation on the Web has been explored in the academic literature since the mid-1990s, see Hammond, Burke, and Schmitt (1994) or Linden, Hanks, and Lesh (1997) for early *critiquing*-based approaches. In these early systems, an interaction model was implemented where the system made recommendations on which users could apply feedback in the form of critiques with respect to certain item attributes. For example, users could state that they were looking for something "cheaper" or for a laptop of "higher processor speed." While these systems enabled interactive dialogues, the number of supported interaction types was limited and often predefined based on domain knowledge. Later on, the first CRS to support a natural language interface were proposed (Thompson, Göker, and Langley 2004). These early systems were, however, often hampered by the limited capabilities of NLP technology available at that time. Since then, NLP technology has greatly improved and represents one main pillar of academic CRS that are implemented as chatbots (Iovine, Narducci, and Semeraro 2020; Qiu et al. 2017). In the majority of today's real-world chatbot applications, the system's responses are, however, mostly based on predefined templates, and the main tasks where AI technology comes into play, besides language processing, include the detection of the user's intents and the recognition of entities in the user utterances. To avoid the bottleneck that comes with the definition of the templates, recent *end-to-end* learning systems commonly train complex models on large corpora of recorded recommendation dialogues between humans. Table 1 lists a number of conversational recommenders that were successfully applied in various domains over the years.

Given these success stories and the recent technical developments, one might think that building a CRS is a solved problem. Building a CRS also seems much easier than creating a *general* conversational AI system because (i) only one specific task, that is, recommendation, has to be supported and (ii) many field-tested machine learning models exist to determine item recommendations for a given user profile. However, an analysis of two very recent learning-based approaches indicated that today's academic systems are far from being usable in practice (Manzoor and Jannach 2021a). These systems were found to be rather limited in terms of what kind of conversational acts they support or in terms of their ability to maintain the dialogue context. A simulation based on dozens of dialogues ultimately revealed that the system responses were considered to be meaningful by human evaluators in less than two third of the cases, raising questions about their applicability in practice.

Engineered CRS, which are based on predefined templates or explicitly coded dialogue knowledge, are often able to avoid such problems as their behavior is predictable. Ultimately, however, our goal is to avoid the knowledge engineering effort required by such systems and to build systems that are able to learn to converse from data, at least to a certain extent. In this paper, we discuss why building such learning-based CRS, which are able to engage in a recommendation dialogue at the level that we would expect from a human, is one of today's grand challenges and testbeds for AI-powered technologies. We first discuss the various aspects that make building a CRS difficult. We then review limitations of current approaches in particular with respect to how we evaluate systems in academia. Finally, we provide an outlook on possible next steps for building next-generation conversational recommenders.

WHY BUILDING A CRS IS DIFFICULT

A CRS can be characterized as "*a software system that supports its users in achieving recommendation-related goals through a multi-turn dialogue*" (Jannach et al. 2021). A recommendation-related goal in that context can, for example, be to help users find relevant items or, more generally, to make better decisions. However, there can also be more indirect goals like helping users to understand the space of options or explaining to them why a certain option is a good choice for them.

Following this definition, a CRS is a task-oriented system. This differentiates CRS from general conversational AI systems, including the famous ELIZA system from the 1960s. In some ways, building a CRS might, therefore, appear to be an easier task, because the conversation to be supported is usually bounded to a few of predefined tasks and dialogue situations. Moreover, the competence of a particular CRS can furthermore be limited to a certain domain, for example, movies.

However, achieving a certain naturalness of conversational recommendation dialogues can be challenging. For example, in case of a CRS supporting natural language interactions, the virtual CRS agent should probably be able to respond to chit-chat ("phatic") user utterances. Furthermore, a conversation between humans—which a CRS might aim to mimic—is much richer than just answering questions like "*What is a good sci-fi movie?*" In such a conversation, the initiative might also switch between dialogue partners, thus requiring a system that supports both *user-driven*, *system-driven*, or *mixed-initiative* dialogues. Moreover, the system must be able to respond to a variety of possible *user intents*, for example, providing or revising preference information, asking for explanations, or

TABLE 1 Example applications of conversational recommender systems over time

System	Interaction modality	Description
Wasabi Personal Shopper (Burke 1999)	Forms and buttons	The Wasabi Personal Shopper was a database browsing tool based on case-based reasoning and the earlier FindMe systems. It allows users to critique a recommended product so that the system can return a new product based on their feedback.
Advisor Suite/CWAdvisor (Jannach 2004)	Forms and buttons	Advisor Suite was a commercialized software for personalized sales advisory following a knowledge-based approach. Several applications built with the CWAdvisor were deployed in the domains of electronics, investment products, wine, and fine cigars.
MobyRek/STS (Ricci and Nguyen 2007; Braunhofer, Elahi, and Ricci 2014)	Forms and buttons on Mobile	MobyRek was one early CRS designed for mobiles, and it supported critiquing for recommending travel-related items. South Tyrol Suggests (STS) is a more recent smartphone application able to leverage context information for the recommendation of points-of-interest.
AI bot for shopping (Yan et al. 2017)	Natural language	This AI bot is a task-oriented dialogue system built as a shopping assistant for online mobile e-commerce. It is one of the first Chinese AI bots used in online shopping environments with millions of consumers. It leverages various knowledge sources and crowdsourcing to deal with large catalog size and cold-start issues.
AliMe Chat (Qiu et al. 2017)	Natural language	AliMe Chat is an open-domain industrial chatbot relying on a hybrid end-to-end deep learning approach. It was integrated and tested within a real-world intelligent assistant in the e-commerce domain at Alibaba to support customer service, shopping guidance, and life assistance such as flight booking.

rejecting a recommendation. Finally, the CRS must be able to keep track of the ongoing dialogue and possibly even past interactions with the user, as done in Thompson, Göker, and Langley (2004) or Ricci and Nguyen (2007). We discuss these challenges in more detail next after we review the main building blocks of a CRS.

Conceptual architecture of a CRS

Any CRS is an interactive software application, which repeatedly processes inputs of end users and reacts in appropriate ways, for example, by making a recommendation, by asking questions about preferences, or by providing explanations. To support such interactions, the conceptual architecture of a CRS typically comprises the components sketched in Figure 1.

Input and output processing

The interaction with a CRS can be designed in various ways and support different *modalities*. In traditional critiquing systems, users typically interact with the system through predefined forms, either on the web or on the mobile. This is the most simple form of *input processing* as the provided screen elements and their semantics are defined when the system is designed. More recent systems, however, also aim to support conversations in natural language, either in spoken or written form. Implementing such a functionality leads to increased complexity and requires the inclusion of additional system components, for example, for speech-to-text conversion or for named-entity recognition (the latter requiring an underlying knowledge of known entities). In terms of the system *outputs*, the traditional approach is to present the next interaction screen to the user, which may contain various interaction elements (e.g., radio buttons for preference refinement) or a table of item suggestions. In cases where natural language interaction is supported, additional components, for example, for text-to-speech conversion, are required.

Even more complexities can arise when more than one interaction modality should be supported. On the input side, it is, for example, not uncommon in chatbot applications that users in some situations are asked to operate predefined interactive elements such as buttons, and at the same time are allowed to interact with the system by typing free text. One of the difficulties in such a design is to understand which is the most suitable interaction form in a given situation (Iovine, Narducci, and Semeraro 2020). In

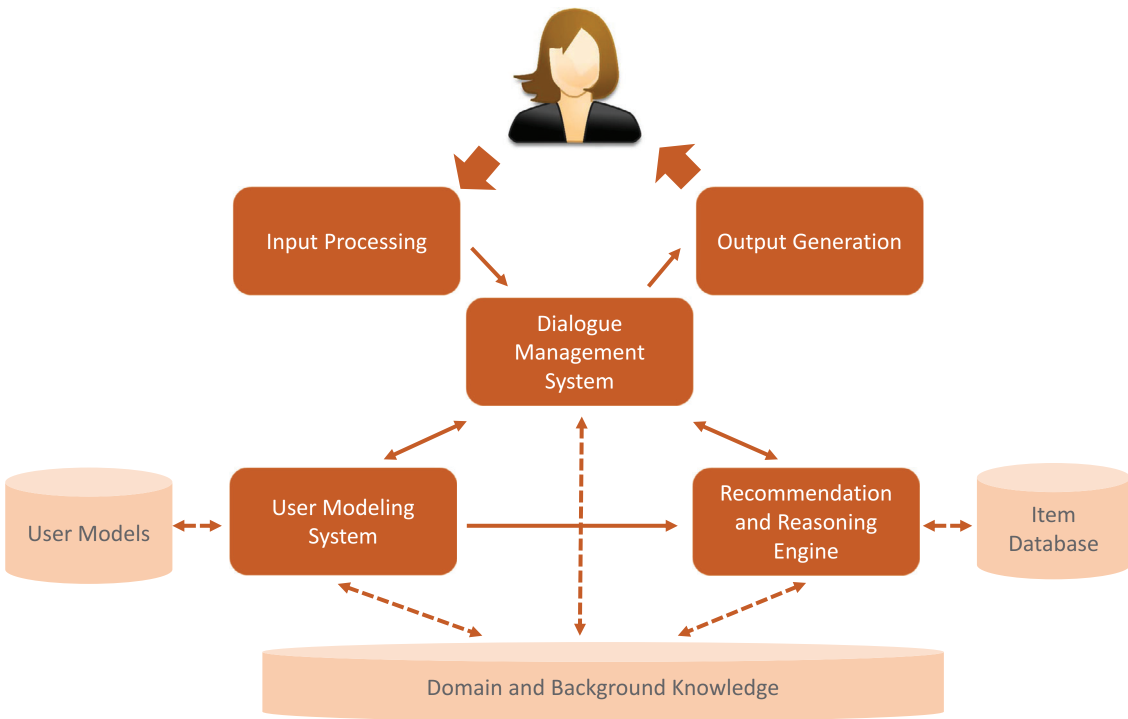

FIGURE 1 Conceptual architecture of a CRS (adapted from Jannach et al. (2021))

other applications, a *hybrid* design of the input and output modalities might be desired, where the input is for example speech-based, but the output is a combination of speech output, structured textual and visual information. Such a hybrid system might in particular be favorable when more than one recommendation should be presented to the user through a multimodal user interface. A CRS on a smartphone, for example, might accept voice input in the preference elicitation phase, and then return the best recommendation via voice output. Finally, several approaches exist that rely on *embodied conversational agents* to provide a more persuasive or emotional user experience and to thereby increase the effectiveness of the system, see, for example, Foster and Oberlander (2010).

User modeling and recommendation reasoning components

On the *backend* side of a CRS, multiple components are usually needed. These include, as a central element, a component that generates recommendations that match the user's preferences. These preferences are maintained with the help of a user-modeling component. This component may either consider only the user's preferences stated in the ongoing dialogue, or it may be able to also make use of long-term preference profiles (Ricci and Nguyen 2007; Thompson, Göker, and Langley 2004). How to combine long- and short-term preferences—which may be changing and thus conflicting across sessions—in the best way, is so far a largely unexplored question.

From a technical perspective, various approaches are possible to generate recommendations. One can, for example, rely on explicit recommendation rules and constraints, as done in critiquing or constraint-based systems (Chen and Pu 2012; Felfernig et al. 2015; Jannach 2004). Other approaches train machine learning models based on collaborative information (e.g., rating datasets), which is then combined with preference information from the ongoing dialogue (Christakopoulou, Radlinski, and Hofmann 2016) and maybe structured external data. Besides the generation of recommendations, the backend components might furthermore support additional reasoning tasks, for example, the computation of a user-oriented explanation for the recommendation. Also, it might implement complex reasoning functionality, for example, in constraint-based systems, to determine recommendations in case the customer's requirements cannot be fulfilled by any recommendable item.

Background knowledge

To accomplish these tasks, the backend uses different types of knowledge. Besides an optional database containing long-term preference information, a CRS has at least explicit knowledge about the items that can be recommended, sometimes including meta-data. Various other types of knowledge can be integrated as well. This may include explicit knowledge about possible dialogue states and supported user intents as well as different forms of "background knowledge." The background knowledge can include both additional item-related knowledge and structured world knowledge (e.g., from DBPedia), as well as unstructured textual sources such as logs of recorded recommendation dialogues between humans.

This latter class of background information, that is, logs of human-to-human conversations, often serves as a main basis for end-to-end learning approaches, as in Chen et al. (2019) and Li et al. (2018). In the last few years, a number of such dialogue corpora were published, such as the ReDial dataset (Li et al. 2018) that consists of over 10,000 dialogues created with the help of crowdworkers.

An alternative way of collecting such recommendation dialogue corpora with crowdworkers is the use of Wizard-of-Oz studies. In such studies, the participants interact with a human agent without knowing that it is not a chatbot. Radlinski et al. (2019) used such a study to build a dataset focused on the preference elicitation process. Furthermore, a CRS might also rely on more *general* dialogue corpora, that is, ones that are not necessarily centered around a recommendation task. Facebook, for example, released different datasets in the context of the *bAbI* project, designed to serve as a basis to train end-to-end learning systems for different tasks, for example, for restaurant bookings. Finally, recent works explore the use of very general pretrained language models like BERT or GPT-3 in the context of CRS. Penha and Hauff (2020) explore what such a general model like BERT already knows about movies, for example, about their genres, and discuss where such models currently succeed and where they fail.

Dialogue management

Today's voice controlled digital assistants often fail to maintain the context of an ongoing conversation and, for example, do not remember or take into account what was said in previous interaction cycles. Keeping track of the conversation is, however, a central feature in a CRS, for example, when the system tries to interactively acquire the user's needs or preferences.

To keep track of the current situation, CRS typically rely on a predefined set of *dialogue states*. These states can either be defined explicitly, for example, in the form of a state transition graph (Jannach 2004; Ricci and Nguyen 2007), or implicitly. Figure 2 shows a dialogue graph that determines not only the predefined states, but also the possible transitions, that is, the *conversational moves*. Encoding the knowledge in this way is common in critiquing and "slot-filling" preference elicitation approaches, where the system asks questions to the user about her preferences for a predefined set of item attributes. Note that while the set of states is typically predefined and static, the choice of the next conversational move, for example, whether to ask more questions or show a recommendation, can be determined dynamically, for example, based on reinforcement learning (Christakopoulou, Radlinski, and Hofmann 2016; Ricci and Nguyen 2007).

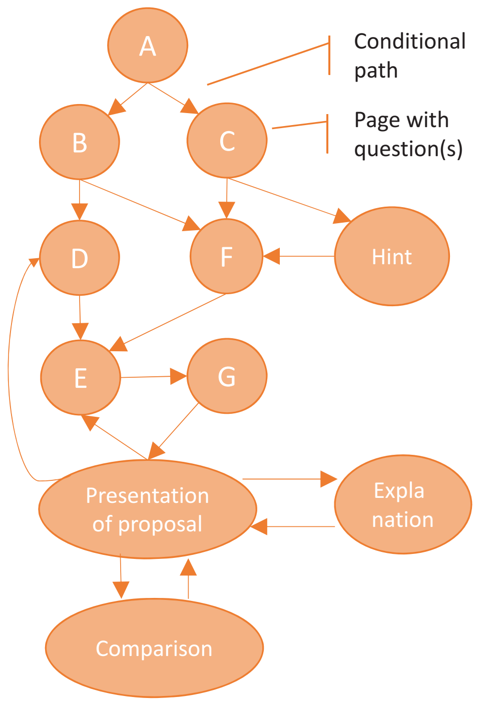

FIGURE 2 State transition graph (adapted from Jannach et al. (2021))

In contrast, a typical form of representing dialogue knowledge in a more implicit way is to model the set of supported *user intents*. Such an approach is common in today's chatbot applications and tools like Google's *DialogFlow*. The main task of such dialogue engines is to guess the most probable intent from a user utterance given in natural language. The responses are then commonly constructed by filling templates that are predefined for each intent.

End-to-end learning approaches, finally, seek to avoid the knowledge-intensive process of defining explicit dialogue states. These systems aim to learn from data as much as possible how to react appropriately given a current user utterance and the history of the ongoing dialogue. For example, when observing a greeting from the user, the learned model will, after training, respond with a greeting as well.

Discussion

Our discussions show that the development of CRS comes with many challenges that are not present in "one-shot" recommender systems found online or in digital assistants. Determining suitable recommendations given a set of known user preferences, for example, in the form of previously liked items or explicitly stated preferences regarding item attributes, is probably one of the easier parts. Many existing algorithms for item ranking can in principle be applied. There are, however, many design choices that can be made for the other components. It can, for example, be less than clear which mix of interaction modalities is the best one for a given application. Some users might prefer voice input, others might prefer to type text, and yet other users might have a preference for predefined input controls such as radio buttons. Likewise, on the output side, a mix of modalities, for example, voice and visual outputs, might be desirable by some users, but not all. Moreover, while providing a human-like avatar or the consideration of additional input signals like facial expressions or gestures might help increase the naturalness of a conversation, such features also add to the complexity of the system.

In the following, we discuss two additional challenges. First, we argue that real-world conversations about recommendations can be rich and multi-faceted, and that a CRS, to be effective, should be able to support such conversations that go beyond simple question–answering (Q&A) patterns. Second, while the use of learning-based techniques in CRS has the promise to reduce or avoid manual knowledge engineering efforts, the use of machine learning can leave it difficult to make quality guarantees regarding the system's behavior.

Recommendation dialogues: More than question–answering

Traditionally, the literature differentiates between *system-driven*, *user-driven*, and *mixed-initiative* conversations. In an entirely system-driven approach, the system leads the dialogue, usually by asking questions, for example, about the user's preferences regarding individual items or item attributes, until it is confident to make a recommendation. Such a guided-dialogue approach is common in knowledge- and critiquing-based systems. Similar "system ask–user respond" dialogue patterns are also found in recent learning-based approaches, for example, Zhang et al. (2018). The other extreme would be an entirely user-driven dialogue, where the CRS only behaves reactively. Voice-controlled smart assistants are an example of such systems that only respond to users, but usually do not take the initiative, for example, by asking questions back to the users.

Real conversations between humans can, however, be more varied, for example, in terms of who takes the initiative and how the dialogue develops. To build a CRS that feels "natural," it is, therefore, important to understand how humans would converse in a given domain. Christakopoulou, Radlinski, and Hofmann (2016) conducted such an analysis in the restaurant recommendation domain as a starting point for their research to understand which questions should be asked to users.

Similar analyses in Cai and Chen (2020) reveal that, in reality, human recommendation dialogues can be quite rich in terms of what kinds of information are exchanged. Moreover, such information exchanges are often not limited to recommendation-related aspects, such as providing or revising preferences, asking for an explanation, or requesting additional information. In particular, when natural language interactions are supported, we often observe social communication acts such as greetings or phatic expressions. Moreover, besides such human-to-human interactions, there can be specific information exchanges that only happen in human–machine interactions, for example, when the user wants to restart a broken conversation.

A catalog of user intents and system actions

Several previous studies analyzed the users' intents behind their utterances when they interact with a conversational system in natural language. For instance, Yan et al. (2017) identified the four most-frequent user intents in a shopping chatbot: *recommendation*, *comparison*, *ask opinion*, and *Q&A*; and three session-aware intents: *add filter condition*, *see-more*, and *negation*. Kang et al. (2017) collected initial and follow-up queries issued by users when they ask for recommendations via speech- or text-based dialogues. They classified these initial queries into *objective*, *subjective*, and *navigation* goals, and the follow-up queries into the categories *refine*, *reformulate*, and *start over*.

Most recently, Cai and Chen (2020) established a catalog of user intents based on an analysis of human–human dialogues centered around movie recommendations. Specifically, they annotated a subset of conversations from the ReDial dataset after performing a cleaning process. The final catalog of user intents was then created using an iterative *grounded theory* procedure. In the catalog, the intents are classified into three top-level intents, that is, *Ask for Recommendation*, *Add Details*, and *Give Feedback*, and 15 sub-intents (see Table 2). For instance, there are four subintents under *Ask for Recommendation*, including *"Initial Query," "Continue"* (continuing to seek for more suggestions), *"Start Over"* (when the recommendation seeker starts a new query), and *"Reformulate"* (when the seeker wants to revise a previous query).

Moreover, Cai and Chen (2020) classified the actions of the human recommenders into four top-level categories, that is, *Request*, *Respond*, *Recommend*, and *Explain*, and nine sub-actions (see Table 3). An analysis regarding the action distribution shows that the sub-actions under *Explain* more frequently occur in satisfactory dialogues,

TABLE 2 Catalog of user intents

Intent	Description	Example
Ask for Recommendation		
Initial Query	Seeker asks for a recommendation in the first query.	"*I like comedy do you know of any good ones?*"
Continue	Seeker asks for more recommendations in the subsequent query.	"*Do you have any other suggestions?*"
Reformulate	Seeker restates her/his query with or without clarification/further constraints.	"*Maybe I am not being clear. I want something that is in the theater now.*"
Start Over	Seeker starts a new query to ask for recommendations.	"*Anything that I can watch with my kids under 10.*"
Add Details		
Provide Preference	Seeker provides specific preference for the item s/he is looking for.	"*I usually enjoy movies with Seth Rogen and Jonah Hill.*"
Answer	Seeker answers the question issued by the recommender.	"*Maybe something with more action.*" (Q: "*What kind of fun movie you look for?*")
Ask Opinion	Seeker asks the recommender's personal opinions.	"*I really like Reese Witherspoon. How about you?*"
Give Feedback		
Seen	Seeker has seen the recommended item before.	"*I have seen that one and enjoyed it.*"
Accept	Seeker likes the recommended item.	"*Awesome, I will check it out.*"
Reject	Seeker dislikes the recommended item.	"*I hated that movie. I did not even crack a smile once.*"
Inquire	Seeker wants to know more about the recommended item.	"*I haven't seen that one yet. What's it about?*"
Critique-Feature	Seeker makes critiques on specific features of the current recommendation.	"*That's a bit too scary for me.*"
Critique-Add	Seeker adds further constraints on top of the current recommendation.	"*I would like something more recent.*"
Critique-Similar	Seeker requests something similar to the current recommendation.	"*Den of Thieves (2018) sounds amazing. Any others like that?*"
Neutral Response	Seeker does not indicate her/his preference for the current recommendation.	"*I have actually never seen that one.*"
Others	Greetings, gratitude expression, or chit-chat utterances.	"*Sorry about the weird typing.*"

that is, in dialogues where at least one recommended item was liked by the seeker. This may imply that providing explanations for the recommendation is likely to increase user acceptance.

Discussion

While logged dialogues between humans represent a valuable resource for understanding how humans interact, crowdsourced datasets like ReDial have their limitations. When the ReDial data were collected, the crowdworkers were required to mention at least four movies during their conversation. This might have led to dialogues that mostly focus on the instance level. As a result, preferences are often obtained by asking for the seeker's general tastes or opinions on specific movies. Possible other types of preference elicitation—such as asking for pairwise preference feedback on two or more items (Rana and Bridge 2020)—rarely occur.

Regarding the developed catalog of user intents and system actions, note that the analysis was done for only one domain (i.e., movies), and that additional user intents might be relevant in other domains. Also, investigating chit-chat intents in more depth seems advisable, for example, in terms of *in which condition* users might have such intent, and *what kind of chit-chat* they may expect to have. In a recent work, Liu et al. (2020) developed a CRS to cover multiple dialogue types such as *Q&A*, *chit-chat*, and *recommendation* in a primarily system-driven approach, where the system proactively leads the conversation by following a planned goal sequence. However, it seems that their collected dialogue dataset (called DuRecDial) was limited to predefined user profiles and task templates, which might not fully reflect the characteristics of natural conversations.

Generally, traditional CRS such as critiquing approaches are mostly limited to form-based preference elicitation and in many cases based on a predefined set of attributes. Today's CRS often only support a smaller subset of user intents. Some of end-to-end learning-based systems, on the other hand, sometimes only support a limited set of

TABLE 3 Catalog of recommender actions

Action	Description	Example
Request		
Request Information	Recommender requests for the seeker's preference or feedback.	"*What kind of movies do you like?*"
Clarify Question	Recommender asks a clarifying question for more details.	"*What kind of animated movie are you thinking of?*"
Respond		
Answer	Recommender answers the question asked by the seeker.	"*Steve Martin and John Candy*." (Q: "*Who is in that?*")
Respond-Feedback	Recommender responds to other feedback from the seeker.	"*That's my favourite Christmas movie too!* " (U: "*My absolute favourite!!*")
Recommend		
Recommend-Show	Recommender provides recommendation by showing it directly.	"*The Invitation (2015) is a movie kids like*."
Recommend-Explore	Recommender provides recommendation by inquiring about the seeker's preference.	"*Have you seen Cult of Chucky (2017) that one as pretty scary*."
Explain		
Explain-Introduction	Recommender explains recommendation with non-personalized introduction.	"*What about Sleepless in Seattle (1993)? Hanks and Ryan?*"
Explain-Preference	Recommender explains recommendation based on the seeker's past preference.	"*Will Ferrell is also very good in Elf (2003) if you're in need of another comedy*"
Explain-Suggestion	Recommender explains recommendation in a suggestive way.	"*If you like gory then I would suggest The Last House on the Left (2009)*."
Others	Greetings, gratitude expression, or chit-chat utterances.	"*Have a good night*."

actions like asking for a preference and recommending an item, and they are unable to provide explanations or consider user preferences about item attributes. Overall, more research is, therefore, required to better understand how humans interact in recommendation dialogues, in particular in terms of the social communication acts that may help to increase the acceptance of recommendations (Hayati et al. 2020).

The need for predictability and quality guarantees

According to our discussions, there are two extreme approaches of how to build the core of a CRS. One extreme is to explicitly encode the different types of knowledge needed for dialogue management and item recommendation, leading to an entirely deterministic system. Such an approach is desirable or even required when the quality of the responses must be guaranteed, for example, when the system makes recommendation in the health or finance domains. The other extreme is to try to learn everything from observed dialogues between humans, that is, from mostly unstructured data. Various existing approaches are placed somewhere between these two extremes (Christakopoulou, Radlinski, and Hofmann 2016; Ricci and Nguyen 2007).

Explicitly modeling the dialogue and recommendation knowledge, for example, in the form of state automata and recommendation rules, can require substantial knowledge engineering. Special-purpose modeling environments may help to reduce these efforts (Jannach 2004). However, the problem remains that the knowledge bases have to be manually updated, for example, when new items become available that require the change of the recommendation logic or the revision of the dialogue flow. Also, these systems are by design restricted in terms of which interactions they support. For example, free-text inputs in natural language are usually not supported.

One ultimate vision of a pure end-to-end learning system, as the other extreme, is that the behavior of the CRS (e.g., the choice of dialogue moves and the recommendations) is entirely learnt from data. In such an approach, no knowledge engineering is required and the system automatically updates its models whenever new data arrive. At the same time, such learning-based systems are not limited to a predefined set of dialogue situations and may be able to react to a rich set of user intents that were observed previously in the data.

One main challenge of learning-based systems, however, is that there are usually no quality guarantees regarding the system responses, as they are not entirely predictable. One main problem is that there are several places in a typical CRS architecture where a learning-based system can fail. During natural language input processing, for example, such systems may encounter difficulties in speech-to-text conversion, named entity recognition, or intent detection. Given these uncertainties, it might, therefore, be desirable to constrain the possible outputs of a learning-based CRS to assure a certain level of predictability and guaranteed output quality. This may start with constraints at the grammatical level when generating sentences with the aim to avoid ungrammatical system responses. At the dialogue level, one may furthermore constrain the system to a predefined set of user intents and system actions. Finally, one could also restrict the final outputs to be based on predefined and quality-assured response templates, which turns response generation into a retrieval task. All these measures may ultimately help to increase the predictability (and quality) of the system output. However, these measures also add several knowledge-based components to the overall architecture, leading to challenges regarding knowledge modeling and maintenance.

To exemplify the challenges of today's learning-based systems, consider the case of the approaches by Li et al. (2018) (termed DeepCRS) and Chen et al. (2019) (termed KBRD), which were presented in recent years at leading conferences on NLP and neural networks. Both approaches are based on the ReDial dataset, and KBRD also relies on structured external knowledge. Evaluations of these systems in Manzoor and Jannach (2021a; 2021b), which also compared them with a retrieval-based approach, indicated that both systems seem to make too many mistakes when generating responses. For both systems, at least 30% of the system responses were considered to be not meaningful dialogue continuations according to the authors. Typical problems include that the system did not properly react to a user question, repeatedly made the same recommendation, or abruptly ended the dialogue. These findings raise questions regarding the usefulness of such unconstrained academic approaches in practice. An interesting side observation in the mentioned evaluation was that both end-to-end systems actually did not generate new sentences and almost all of the returned responses appeared in similar or identical form in the training data.

Clearly, our general goal when building CRS of the future is not to rely on pre-implemented dialogue flows and knowledge-based recommendation systems, but to design "intelligent," learning-based solutions. More work is thus needed to understand how useful our solutions are in practice and how we can build systems that guarantee a certain quality level. Moreover, better methodological approaches seem to be needed to assess the quality of a CRS in the lab and how to compare different conversational systems. We discuss such methodological questions in the next section.

WHY EVALUATING A CRS IS CHALLENGING

CRS are multicomponent adaptive interactive systems

Modern CRS are in general complex, multicomponent software systems, which have to support nontrivial interactions with their users. Consequently, the quality perception and the adoption of such systems by users can depend on a variety of factors, and in case the system is not as successful as expected, many components may be the culprits. The system might, for example, have failed to understand the user utterance or to extract the correct intent. Or, it might have not been able to respond appropriately to question, or it might just have made poor recommendations. Finally, there might have been issues with the user interface or the supported interaction modality. Overall, any evaluation of a CRS, therefore, must aim to isolate these potential factors to see what works well and what does not.

In that context, note that the evaluation of the quality of the underlying *recommendation algorithms* alone can be challenging in its own. Today, the research community largely relies on offline evaluation procedures using historical datasets to assess how good an algorithm works. Such an approach can be suited to assess how good an algorithm is at predicting which item a user will rate highly or click on. However, these evaluation procedures cannot inform us about the quality perceptions by users, for instance, if the recommendations will help users discover new items, or if the recommendations will lead to higher business value. Such aspects can only be assessed through user studies or field tests. In fact, in CRS, the majority of the aspects that may contribute to the success or failure of the system cannot be reasonably assessed without involving humans users. In contrast, both in earlier critiquing-based CRS and in recent chatbot applications, a very common approach is to assess the number of required interaction turns until the user finds a suitable recommendation through simulations. Such simulations, however, rely on a number of assumptions like rational user behavior or that the user preferences are stable and pre-existing. Moreover, in practice, longer interaction sessions with a CRS not necessarily mean that the system does not work well. Instead, longer interactions might in contrast indicate higher engagement and more exploration by users (Cai, Jin, and Chen 2021; 2022).

The need for multi-faceted evaluations

Generally, we can identify three main dimensions in which CRS are evaluated: (i) *Effectiveness of task support*, (ii) *Efficiency of task support*, and (iii) *Quality of the conversation and usability*. Ultimately, all of these aspects can contribute to the success of a CRS in practice.

Effectiveness of task support: These measurements relate to the ability of the CRS to support a recommendation-related task, for example, helping users to make a decision or to find an item of interest. In the literature, researchers assess such aspects with the help of evaluations with users and/or through offline experiments. In user studies, often both *objective* and *subjective* measures are applied. Objectively, one can, for example, determine how often users actually found an interesting item during the study. This can, for example, be measured by counting add-to-cart actions or *task completion rates*. Common subjective (self-reported) measures in user studies include *decision confidence*, *perceived recommendation quality*, and *purchase/return intentions*. A more general level, *user satisfaction*—usually with the system as a whole—is frequently used in the literature as well.

In offline studies on effectiveness, common proxies from non-conversational systems are typically applied, including all types of accuracy measures like Precision, Recall, and RMSE. Often, such measures are taken in a simulation-based approach, where artificial users with predefined preferences interact with the CRS. Besides accuracy, some studies also measure the *success rate* and *rejection rate* in simulations. Since any CRS is a multicomponent system, researchers sometimes focus on individual components in their evaluation. A typical problem, for example, is to assess the performance of the entity and intent recognition modules as in Liao et al. (2019) or Narducci et al. (2018).

Efficiency of task support: Measurements in this category assess how quickly users make a decision or find something suitable. As discussed above, one underlying assumption is that shorter dialogues are preferable. Traditionally, the number of *required interaction cycles* in simulated user–machine conversations is measured. An alternative objective measure is *task completion time*. Iovine, Narducci, and Semeraro (2020), for example, compared the use of different interaction modalities of a CRS—language-based, button-based, mixed—in a user study. They found that natural language interfaces can lead to a less efficient recommendation process, which is partially due to the challenge of understanding the natural language input. The use of voice-based interactions was studied in Yang et al. (2018). Here, the authors found that users of a podcast recommender were slower and explored fewer options when interacting through voice. Combined, these findings indicate that natural language-based interaction modalities may make the interaction process less efficient than when buttons and forms are used.

Quality of the conversation and usability: Commonly, these aspects are evaluated through subjective measures, where users are asked about their quality perceptions. In terms of general usability, the *ease-of-use* of the system or the *task ease* are often in the focus. Looking at the quality of the recommendation process itself, researchers furthermore addressed questions of *transparency* or the perceived level of *user control*. At the dialogue level, researchers furthermore applied ideas that were previously applied for general spoken dialogue systems, and measured how quickly a system adapts to the user's preferences, how intuitive and natural the dialogue feels, and if the dialogue is entertaining. Moreover, Pecune et al. (2019) considered *coordination*, *mutual attentiveness*, *positivity*, and *rapport* as factors that might affect the perceived dialogue quality. In other approaches, a variety of additional subjective measures were used, including *consistency*, *engagingness*, *informativeness*, or *relevance*.

Recently, various attempts were also made to *objectively* measure linguistic aspects of the system's generated utterances. One idea is to compare these utterances with ground-truth utterances by humans in a given dialogue situation, using, for example, the BLEU or NIST scores, which are commonly used in machine-translation tasks. Alternative linguistic measures include the lexical diversity or perplexity (fluency), for example, in Chen et al. (2019).

Discussion

The recommender systems research community has developed broadly accepted standards regarding the evaluation of such systems. In the predominant area of algorithms research, offline experimentation and the use of metrics from information retrieval and machine learning are common. On the other hand, for user-centric research, different general evaluation frameworks were proposed, for example, by Pu, Chen, and Hu (2011) or Knijnenburg et al. (2012). Both types of evaluation approaches can be applied for CRS as well. The general limitations of offline evaluations, however, remain, that is, that it is not always clear if offline results are representative of the user-perceived qualities of the recommendations. The user-centric frameworks, on the other hand, are by design focusing on general aspects of recommender system acceptance such as the perceived recommendation quality, usability aspects, or the intention to use the system in the future.

In the context of CRS, given their interactive nature, offline evaluations might only be informative for very specific subtasks, such as the recognition rate of entities in user utterances, which are often not very specific to CRS. As a result, almost all research efforts in some way might require the involvement of humans in the evaluation process. Recent works on end-to-end learning approaches, as those mentioned above by Li et al. (2018) (DeepCRS) and Chen et al. (2019) (KBRD), therefore, apply a combined approach in their evaluations, which is based both on objective measures in offline evaluations and on human assessments of the dialogue quality.

There are, however, a number of potential pitfalls to be considered. For example, using metrics like the BLEU score in offline experiments to compare a system-generated response with a ground-truth response from a recorded dialogue may be problematic in different ways. First, the commonly used BLEU score, according to Liu et al. (2016), not in all cases seems to correlate with human perceptions. Second, an evaluation procedure where the system response is compared to a ground-truth statement might be too limited, because many alternative generated statements might be appropriate as well in a given dialogue situation. Moreover, remember that in some works, metrics are used to measure the fluency of the system responses. However, in cases where the end-to-end learning system is actually not generating new responses, but only returns utterances that appear in the training data, the application of fluency metrics might not be too meaningful.

User-centric research can also have pitfalls. When evaluating the DeepCRS and KBRD system, the authors for example rely on human annotators to rank the responses by different systems in a given dialogue situation, for example,, in terms of the consistency with what was said before in the dialogue. However, in some evaluations, the task of the annotators was to assess which system response is better. Given only such *relative* judgments, it unfortunately remains unclear if the systems generate high-quality responses on average on an *absolute* scale. Moreover, for the evaluation of some specific aspects of CRS, for example, regarding the naturalness or engagingness of the conversation, very specific experimental setups are required. Unfortunately, no commonly accepted evaluation standards for such questions exist so far.

MOVING FORWARD

Improving our research methodology

A direct consequence of our discussions is that we should aim at further improving our evaluation methodology. Such improvements could, for example, be achieved through the development of a unified user-centric evaluation framework for CRS, which (i) considers and relates the various factors that may influence the quality perceptions of users, and (ii) provides standardized questionnaire for the different model constructs; see also Jin et al. (2021). Moreover, in case of offline evaluations, the current metrics that we use to automatically assess the linguistic quality of system-generated responses have to be re-assessed and validated.

Generally, it seems advisable that we rely on a richer methodological repertoire, which allows us to conduct multimodal and multi-metric evaluations to obtain a more comprehensive picture of the quality of the different aspects of a CRS. In that context, also more *exploratory* research is needed, for example, to understand how humans interact in conversations, what users expect from a computerized advisor, or how tolerant they are with respect to problems in such a conversation.

Combining learning- and knowledge-based systems

Our analysis of recent end-to-end learning system indicates that building a CRS solely based on learning from logged interactions still has a number of limitations. One reason for this could lie in the current lack of datasets that contain richer interactions between humans. The ReDial dataset, as mentioned above, in some ways may appear unnatural. This seems to have led to the effect that mentions of desired movie attributes like genres or actors are under-represented, which makes learning difficult.

Therefore, for the time being, hybrid AI-based solutions seem to be the method of choice, where certain parts of the needed knowledge are learned from past data, certain parts are represented in structured form and taken from external sources such as DBPedia, and certain parts are manually engineered based on domain expertise. How to combine learned and explicit knowledge in the best way is a very active research area in AI. Moreover, as discussed above, more research is required to understand how to provide certain guiding rails for learning-based systems in order to ensure predictable, high-quality responses.

Supporting novel interaction forms and application domains

In their traditional form, CRS were mostly web-based applications with a predefined form for eliciting user feedback on a set of fixed attributes. Due to recent technological advances, natural language input, either in written or spoken form, has become mainstream. In the future, a more frequent use of additional or alternative forms of interactions can be envisioned. In the literature, we, for example, find approaches that consider *nonverbal* communication acts like body postures, gestures, and facial expressions as inputs. But also new forms of output and feedback are possible, in particular by using novel forms of *Embodied Conversational Agents* or application-specific 3D visualizations.

Future CRS might also not be limited in terms of the application environment. Today, CRS are mostly implemented as desktop or mobile applications. Increasingly, we also see CRS functionality implemented on smart speakers like Amazon Echo. Very differently from that, future CRS might also be part of other physical environments, for example, in the form of an interactive wall installed in a real shop, a service robot in a restaurant, or an in-car system.

Towards social CRS

An analysis of the recorded human–human conversations in the ReDial dataset shows that a substantial fraction of the dialogue represents *phatic* utterances like chit-chat. To further increase the adoption of chatbot-like CRS, it might, therefore, be highly important to further enhance such a functionality so that the system is able to establish a social connection to the users. In Shum, He, and Li (2018), for example, the authors report that users of a social chatbot by Microsoft, which is able to detect the users' emotional needs and personalities, even possess a certain feeling of "social belonging."

While today's end-to-end learning systems seem able to engage in chit-chat conversations, limited work exists so far on understanding the users' current emotions or to react on them. One reason for the lack of such capabilities may lie in the characteristics of many of today's datasets, which do not reflect the full spectrum of how humans would engage in real-world conversations. Recently, a new dataset called INSPIRED was released by Hayati et al. (2020), and the authors show that an end-to-end learning model trained on the data annotated with social strategies can be beneficial. Future work might build on these results and the released datasets to build next-generation social CRS. For instance, as summarized by Thomas et al. (2020), a number of human–human conversation strategies (such as social norms, structures, affect, prosody, and style) might be leveraged to enable the system to be more engaging, persuasive, and trustworthy.

SUMMARY

Being able to conduct natural conversations with a human-like computerized system is a long-standing vision of AI. We have reviewed existing approaches to building interactive advice-giving systems in the form of CRS. Our discussions indicate that today often a trade-off exists between mostly engineered and entirely learning-based solutions. While engineered solutions may excel in terms of predictability and quality guarantees, learning-based systems have the promise to support much more natural and flexible conversations.

CONFLICT OF INTEREST

The authors have no conflicts of interest to report.

ORCID

Dietmar Jannach https://orcid.org/0000-0002-4698-8507
Li Chen https://orcid.org/0000-0002-5842-838X

REFERENCES

Braunhofer, M., M. Elahi, and F. Ricci. 2014. "Usability Assessment of a Context-aware and Personality-based Mobile Recommender System." In *Proceedings of the EC-Web'14*, 77–88.

Burke, R. 1999. "The Wasabi Personal Shopper: A Case-based Recommender System." In *Proceedings of the AAAI'99/IAAI'99*, 844–9.

Cai, W., and L. Chen. 2020. "Predicting User Intents and Satisfaction with Dialogue-based Conversational Recommendations." In *Proceedings of the UMAP'20*, 33–42.

Cai, W., Y. Jin, and L. Chen. 2021. "Critiquing for Music Exploration in Conversational Recommender Systems." In *Proceedings of the IUI'21*.

Cai, W., Y. Jin, and L. Chen. 2022. "Task-oriented User Evaluation on Critiquing-based Recommendation Chatbots." *IEEE Transactions on Human–Machine Systems* 52: 1–13.

Chen, L., and P. Pu. 2012. "Critiquing-based Recommenders: Survey and Emerging Trends." *UMUAI* 22(1-2): 125–50.

Chen, Q., J. Lin, Y. Zhang, M. Ding, Y. Cen, H. Yang, and J. Tang. 2019. "Towards Knowledge-based Recommender Dialog System." In *Proceedings of the EMNLP-IJCNLP'19*, 1803–13.

Christakopoulou, K., F. Radlinski, and K. Hofmann. 2016. "Towards Conversational Recommender Systems." In *Proceedings of the KDD'16*, 815–24.

Felfernig, A., G. Friedrich, D. Jannach, and M. Zanker. 2015. "Constraint-based Recommender Systems." In *Recommender Systems Handbook*, 161–90. Boston, MA: Springer.

Foster, M. E., and J. Oberlander. 2010. "User Preferences can Drive Facial Expressions: Evaluating an Embodied Conversational Agent in a Recommender Dialogue System." *UMUAI* 20(4): 341–81.

Hammond, K. J., R. Burke, and K. Schmitt. 1994. "Case-based Approach to Knowledge Navigation." In *Proceedings of the AAAI'94*.

Hayati, S. A., D. Kang, Q. Zhu, W. Shi, and Z. Yu. 2020. "Inspired: Toward Sociable Recommendation Dialog Systems." In *Proceedings of the EMNNLP'20*.

Iovine, A., F. Narducci, and G. Semeraro. 2020. "Conversational Recommender Systems and Natural Language: A Study through the ConveRSE Framework." *Decision Support Systems* 131: 113250–60.

Jannach, D. 2004. "ADVISOR SUITE—A Knowledge-based Sales Advisory System." In *Proceedings of the ECAI'04*, 720–4.

Jannach, D., A. Manzoor, W. Cai, and L. Chen. 2021. "A Survey on Conversational Recommender Systems." *ACM Computing Surveys* 54(5): 1–26.

Jin, Y., L. Chen, W. Cai, and P. Pu. 2021. "Key Qualities of Conversational Recommender Systems: From Users' Perspective." In *Proceedings of the HAI'21*, 93–102.

Kang, J., K. Condiff, S. Chang, J. A. Konstan, L. Terveen, and F. M. Harper. 2017. "Understanding How People Use Natural Language to Ask for Recommendations." In *Proceedings of the RecSys'17*, 229–37.

Knijnenburg, B., M. Willemsen, Z. Gantner, H. Soncu, and C. Newell. 2012. "Explaining the User Experience of Recommender Systems." *UMUAI* 22(4): 441–504.

Li, R., S. E. Kahou, H. Schulz, V. Michalski, L. Charlin, and C. Pal. 2018. "Towards Deep Conversational Recommendations." In *Proceedings of the NIPS'18*, 9725–35.

Liao, L., R. Takanobu, Y. Ma, X. Yang, M. Huang, and T.-S. Chua. 2019. "Deep Conversational Recommender in Travel." https://arxiv.org/abs/1907.00710.

Linden, G., S. Hanks, and N. Lesh. 1997. "Interactive Assessment of User Preference Models: The Automated Travel Assistant." In *Proceedings of the UM'97*, 67–78.

Liu, C.-W., R. Lowe, I. Serban, M. Noseworthy, L. Charlin, and J. Pineau. 2016. "How NOT to Evaluate your Dialogue System: An Empirical Study of Unsupervised Evaluation Metrics for Dialogue Response Generation." In *Proceedings of the EMNLP'16*, 2122–32.

Liu, Z., H. Wang, Z.-Y. Niu, H. Wu, W. Che, and T. Liu. 2020, July. "Towards Conversational Recommendation over Multi-type Dialogs." In *Proceedings of the ACL'20*, 1036–49.

Manzoor, A., and D. Jannach. 2021a. "Conversational Recommendation based on End-to-end Learning: How Far are We?." *Computers in Human Behavior Reports* 4: 100139.

Manzoor, A., and D. Jannach. 2021b. "Generation-based vs. Retrieval-based Conversational Recommendation: A User-centric Comparison." In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys'21)*.

Narducci, F., M. de Gemmis, P. Lops, and G. Semeraro. 2018. "Improving the User Experience with a Conversational Recommender System." In *Proceedings of the AI*IA'18*, 528–38.

Pecune, F., S. Murali, V. Tsai, Y. Matsuyama, and J. Cassell. 2019. "A Model of Social Explanations for a Conversational Movie Recommendation System." In *Proceedings of the HAI'19*, pp. 135–43.

Penha, G., and C. Hauff. 2020. "What Does BERT Know about Books, Movies and Music? Probing BERT for Conversational Recommendation." In *Proceedings of the RecSys'20*, 388–97.

Pu, P., L. Chen, and R. Hu. 2011. "A User-centric Evaluation Framework for Recommender Systems." In *Proceedings of the RecSys'11*, 157–64.

Qiu, M., F.-L. Li, S. Wang, X. Gao, Y. Chen, W. Zhao, H. Chen, J. Huang, and W. Chu. 2017. "AliMe Chat: A Sequence to Sequence and Rerank based Chatbot Engine." In *Proceedings of the ACL'17*, 498–503.

Radlinski, F., K. Balog, B. Byrne, and K. Krishnamoorthi. 2019. "Coached Conversational Preference Elicitation: A Case Study in Understanding Movie Preferences." In *Proceedings of the SIGDIAL'19*.

Rana, A., and D. Bridge. 2020. "Navigation-by-preference: A New Conversational Recommender with Preference-based Feedback." In *Proceedings of the IUI'20*, 155–65.

Ricci, F., and Q. N. Nguyen. 2007. "Acquiring and Revising Preferences in a Critique-based Mobile Recommender System." *Intelligent Systems* 22(3): 22–9.

Shum, H., X. He, and D. Li. 2018. "From Eliza to XiaoIce: Challenges and Opportunities with Social Chatbots." *Frontiers of Information Technology & Electronic Engineering* 19: 10–26.

Thomas, P., M. Czerwinski, D. McDuff, and N. Craswell. 2020. "Theories of Conversation for Conversational IR." In *Proceedings of the CAIR'20 Workshop*.

Thompson, C. A., M. H. Göker, and P. Langley. 2004. "A personalized system for conversational recommendations." *JAIR* 21(1): 393–428.

Yan, Z., N. Duan, P. Chen, M. Zhou, J. Zhou, and Z. Li. 2017. "Building Task-oriented Dialogue Systems for Online Shopping." In *Proceedings of the AAAI'17*, 4618–26.

Yang, L., M. Sobolev, C. Tsangouri, and D. Estrin. 2018. "Understanding User Interactions with Podcast Recommendations Delivered via Voice." In *Proceedings of the RecSys'18*, 190–4.

Zhang, Y., X. Chen, Q. Ai, L. Yang, and W. B. Croft. 2018. "Towards Conversational Search and Recommendation: System Ask, User Respond." In *Proceedings of the CIKM'18*, 177–86.

AUTHOR BIOGRAPHIES

Dietmar Jannach is a Professor of computer science at the University of Klagenfurt, Austria. His research is focused on practical applications of artificial intelligence, with a focus on recommender systems. He is also the Leading Author of the first textbook on recommender systems published with Cambridge University Press.

Li Chen is an Associate Professor in the Department of Computer Science at Hong Kong Baptist University, China. Her research focuses are mainly on human-centered AI, web personalization, recommender systems, and intelligent user interfaces.

How to cite this article: Jannach, D., and L. Chen. 2022. "Conversational recommendation: A grand AI challenge." *AI Magazine* 43: 151–63. https://doi.org/10.1002/aaai.12059