SME-in-the-loop: Interaction Preferences when Supervising Bots in Human-AI Communities

Zahra Ashktorab
IBM Research
NY, USA
zahra.ashktorab1@ibm.com

Michael Desmond
IBM Research
NY, USA
mdesmond@us.ibm.com

James Johnson
IBM Research
Cambridge, MA, USA
jmjohnson@us.ibm.com

Qian Pan
IBM Research
Cambridge, MA, USA
qian.pan@us.ibm.com

Casey Dugan
IBM Research
MA, USA
cadugan@us.ibm.com

Michelle Brachman
IBM Research
NY, USA
michelle.brachman@ibm.com

Carolina Spina
IBM Research
Buenos Aires, Argentina

ABSTRACT

Subject matter experts play an important role in customer support communities by responding to user queries. Some communities have adopted chatbots in addition to SMEs to address commonly asked questions. Yet, SME-bot interactions, particularly teaching paradigms between SMEs and bots remain understudied. We investigate human-AI machine teaching interactions in a scenario-based study (n=48). Participants selected their preferred teaching method in simulated community interactions with a consumer, an SME, and an AI Bot. We investigated preferences across three interactions: demonstration (Showing), preference elicitation (Sorting), and labeling (Categorization). Participants preferred the Showing interaction, followed by Sorting and Categorizing. Participants changed their preferences from lower-effort interactions when considering downstream outcomes. Users considered the community's perception of interactions between the bot and the SME, specifically transparency of learning outcome, orientation of the feedback, querying the bot and disruptiveness of the interaction. We discuss implications for our findings for teaching interactions in human-AI communities.

CCS CONCEPTS

• **Human-centered computing → Text input**; **Empirical studies in HCI**.

KEYWORDS

human-AI interaction, human in the loop, interactive machine learning, machine teaching, supervising communities, customer support

ACM Reference Format:
Zahra Ashktorab, Michael Desmond, James Johnson, Qian Pan, Casey Dugan, Michelle Brachman, and Carolina Spina. 2023. SME-in-the-loop: Interaction Preferences when Supervising Bots in Human-AI Communities. In *Designing Interactive Systems Conference (DIS '23), July 10–14, 2023, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 23 pages. https://doi.org/10.1145/3563657.3596100

DIS '23, July 10–14, 2023, Pittsburgh, PA, USA

ACM ISBN 978-1-4503-9893-0/23/07.
https://doi.org/10.1145/3563657.3596100

1 INTRODUCTION

Online communities often involve users with a variety of experience and expertise. Subject matter experts (SMEs) or those with more knowledge in communities, are faced with lofty expected contributions and copious workloads because of the imbalance of distributed knowledge. In this work, we focus on multi-participant human-AI environments in which specific individuals within the community, such as SMEs are tasked with helping these chatbots improve. One solution to alleviate the workload on SMEs is to incorporate AI bots that use NLP and other machine learning technologies to automatically answer common questions within the community [54, 64]. AI chatbots, however, come with their own issues: (1) no matter how much data is labeled and used to train a chatbot, chatbots often fail to understand user intent and perform poorly, and (2) the pre-compiled knowledge encoded within a chatbot cannot be updated as fast as the knowledge within the community itself evolves. AI chatbots become stale and the effort required to mitigate these issues is significant. An ideal chatbot is one that is capable of learning on the job [44], meaning that the bot learns from its interactions with other members of a community [60]. How humans interact with a bot to supervise its behavior and teach it new things becomes an important consideration when bots are capable of learning. Yet, we know little about how people prefer to interact with chatbots in community settings in order to teach and improve them, as the role of chatbots in community settings is still under-explored [55]. Our work is motivated by SME needs within communities such as open-source [23] and workplace support groups. Figures 1a and 1b show example dialogues taken from a workplace benefits channel, and an inner-source development channel.

Open and inner source software development communities often have public, or semi-public, multi-participant chat channels where users and contributors are encouraged to engage with the core development team [4]. Conversations in these channels often follow a Q&A style format, where an individual will ask a question, and others will then provide answers in an open setting [19]. Similar questions reappear, and conversations are typically short and quickly pushed out of recent history [19]. A healthy open source community is onion shaped, with a small group of core developers at the center. The external layers of the community includes active users who contribute by testing new releases and reporting bugs, and passive users who are consumers of the library [21]. The core development group can be quite small relative to other members of the community and open source developers have reported feeling overwhelmed by the demands of new-comers with questions related to documentation and requests for bug fixes [23].

Another context where similar dynamics are observed is in workplace support channels. While in the past, workplace support might have been provided by a chatbot in a one-on-one interaction, it is now common for such support to be provided in multi-participant chat channels. An organization might, for example, host a benefits channel where employees can ask questions regarding workplace benefits and related topics in a public forum. These channels include SMEs, whose role is to support the channel by responding to questions from users. In some cases, users might respond to other users if they know an answer, or have some experience to share, but generally information flows from SME to user. Figure 1a shows an example from within one large, multinational corporation, that has thousands of these community support channels on a variety of HR and IT-related topics in an Enterprise Slack workspace. In this work, we consider scenarios from workplace HR/benefits support and programming Slack channels in order to study SME teaching interactions with a chatbot.

To evaluate SME-Bot teaching preferences in a multi-participant chat context, we utilize a taxonomy of interaction archetypes for human-agent learning developed by Koppol et al. as a framework [38]. The four archetypes from that framework are *Showing* (or learning through demonstration), *Categorizing* (when an SME assigns a label to teach the AI agent), *Sorting* (or preference elicitation), and *Evaluating* (or assigning corrections on a list of feedback before or after task execution). Each archetype is encoded into a hypothetical community dialogue/interaction between a user asking for help, an AI bot helping the user, and an SME supervising the bot. Both the interaction archetype itself and the downstream impact on the bot's behavior based on the SME interaction ("2 weeks later" in our scenarios) is represented. Using this instrument, we investigate how subject matter experts might prefer to supervise bots in interactive communities. Specifically, we explore the following research questions:

RQ1 What are subject matter expert preferences for interaction types for human-teaching in a human-AI interactive community?
RQ2 How do different individual and task-related factors impact preferences for different strategies?
RQ3 Do users change their preferences once seeing downstream effects and how do their preferences change?

We present a scenario-based study (n=48) in which participants select their preferred interaction archetype for supervising a learning chat bot. We present users with a total of three different types of interaction (upstream) with respective downstream conversations, in a choice between two scenarios at a time, that would potentially include SME involvement, in two different contexts (HR benefits and Open Source). We collect user preferences for both upstream and downstream aspects of the conversation and report the findings of our study. Our contributions are the following:

- We find SMEs prefer *Showing* (i.e. demonstrating) to the chatbot in order to correct it, as opposed to other interaction types, in a human-AI interactive community, despite its "disruptiveness" to the community and the conversation.
- We identify user perspectives on each of the interaction paradigms and discuss implications of our findings in in a human-AI interactive community.

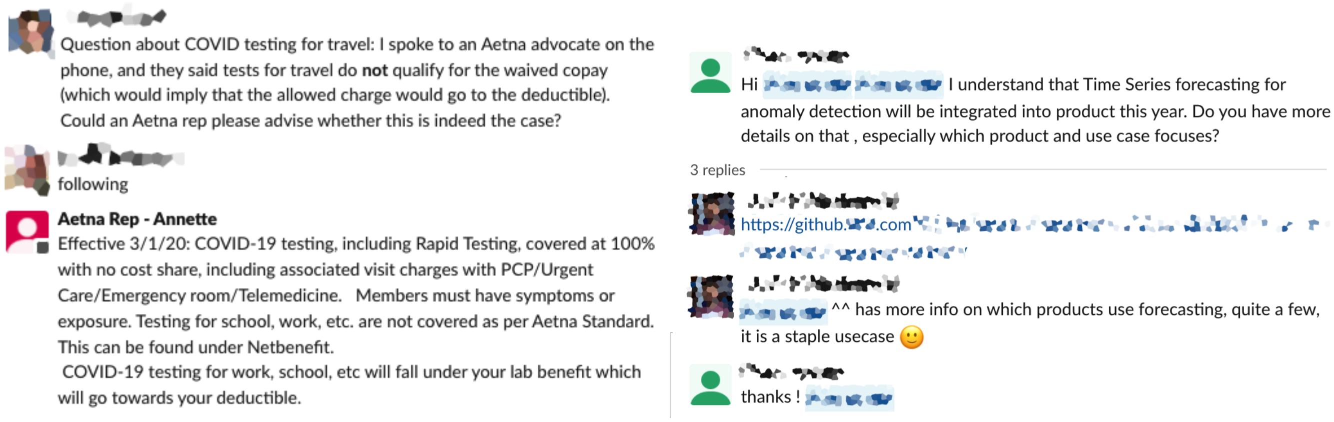
2 RELATED WORK

2.1 Learning Considerations from Human-AI Interaction

In this work, we build on prior work that investigates learning from human-AI interaction, in which machine learning systems improve based on user input. Tools and interaction paradigms have been developed to improve AI agents with the help of users. The traditional machine learning pipeline in the development of an application is a iterative process that includes data collection, feature selection, pre-processing of data, selecting the appropriate model, and assessing the quality of the model. End-users and domain experts are limited to supporting during the data collection process or by providing feedback about the model. Since correct training data is important to develop accurate machine learning models [51], researchers have investigated the best way to collect accurate data to create and improve machine learning models.

Interactive Machine Learning is a common way to improve machine learning models, in which the model is updated based on end-user input. The interaction can be low cost and the update can be minimal, allowing users to interactively steer and improve a machine learning model. A well known example of such systems are recommender systems. Music or movie recommender systems depend on the interactions (like, dislike) that a user inputs into a system. Human-in-the-loop feedback and human-AI interaction provide further ways to collect data to improve these systems. Learning based on human interaction can be traced back to active learning [58], a form of interactive machine teaching involving a human oracle and a machine learner. More broadly, human-in-the-loop optimization is when training feedback, human input, or observed human behavior steer the optimization of an AI agent [18]. In many applications, humans play a key role in evaluating ML performance or providing feedback [45], such as by accepting or rejecting outputs from a model [3]. Human-in-the-loop feedback has been studied in a variety of domains, such as sketching tools [66], classification systems [35], gaming [42], menu designs [12], and chatbots [17, 26].

There have been many inquiries into end-user interactivity and the resulting machine learning model. Fails et al. [24] introduced interactive machine learning to the HCI community when investigating end-user interaction with their Crayon system, which allowed users with very little machine learning expertise to mark pixels as either foreground or background for a image segmentation task.



(a) Insurance benefits example. **(b) Open Source Example**

Figure 1: Two examples from community slack channels in which an SME responds to a consumer question.

The Wekinator was another tool that examined interactive machine learning to create gesture-based instruments. Gestures by the user are interpolated with different sounds by a neural network. The authors found that while students were training and teaching the system through interactivity, the students were also being trained to recognize different noises generated by the system and to steer it in the right direction [28].

As researchers began developing tools to consider user input during the machine learning process, there was an emergence of more research identifying effective teaching strategies and interaction paradigms. In the field of human-robot interaction, researchers identified three types of questions robots can ask humans to learn from them (label, demonstration and feature queries) [15]. Label queries are described as a motion a robot makes to ask whether they have performed a skill correctly. Demonstration queries are when the robot or the agent requests a demonstration from the human teacher. Feature queries are when the robot queries whether a particular feature is important for the target concept. The authors found that humans perceive bots that use feature queries as more intelligent. Koppol et al. built on this work by introducing a taxonomy of interaction types between humans and AI in which humans teach machine learners. They have categorized the interaction types into four categories: *Showing*, *Categorizing*, *Sorting* and *Evaluating* [38]. They measured both usability of these techniques and the amount of subjective effort exerted by users. They found that in the *Evaluation* strategy was more cognitively loading and less usable, while *Categorizing* and *Showing* are least cognitively loading and most usable. In this work, we build on this framework to investigate these learning techniques (*Showing*, *Sorting*, *Categorizing*) in the context of conversation in online communities with SMEs and consumers.

2.2 When to Intervene: Partial Information and Conversational Breakdown

In this paper, we make a distinction between conversational breakdown and a partially correct response delivered by the chatbot, which has a more correct counterpart unbeknownst to the user. Breakdowns in human-AI communication happen often when the bot does not understand a user's utterance or cannot deliver an answer, leading the user to abandon the chatbot altogether or enter a back and forth trial-and-error phase to recover [8]. Breakdowns impact people's satisfaction and trust towards the bot negatively. In our scenarios, the consumer does not know that there is a better more correct response, but the SME is aware. In real world situations, documentation of code may be updated or moved. For example, an insurance/benefit policy may change before a chatbot is updated to reflect the changes. The information delivered by the bot in these situations may be technically correct and not considered a breakdown of the conversation. In this paper we focus on instances in which the bot understands the user's utterance and delivers a response that is considered correct. However, with the help and interaction of the SME, a more correct response can be delivered for future cases. This distinction between conversational breakdowns versus partially correct responses reflects the realities that occur in human-AI communities where one individual (the SME) may possess more information than both the bot and the consumer. Ideally, as SMEs provide the most correct answer, the information they provide can also be instructive for bots, such that the bots improve and reduce the burden on the SME in the future.

Human-in-the-loop ML can occur at different points throughout the ML development and training process, sometimes ending based on stopping conditions. Researchers have worked to understand at what step of the interactive machine learning pipeline it is most effective for an end-user or SME to intervene. In prior work, the stopping condition has been based on the satisfaction of the user, or whether improvement was shown in the ML results [3, 73]. Other studies have investigated human-in-the-loop interaction with ML only at specific times which is determined by the humans at the outset of the study [25, 50, 70]. Lastly, many studies have investigated human-AI interaction when the human interacts with ML once (either in data producing, pre-processing, ML modeling, and/or refinement) [51]. For example, in some ML applications, human involvement during the refinement/evaluation process occurs only once, either at a predetermined point [74] or when the ML method was not able to perform a task [11]. In our work, the intervention is limited to when the SME identifies that only partial information was delivered, unbeknownst to the consumer asking the question.

2.3 Bots in Communities

Chatbots are often used in community Slack channels to facilitate conversation and to respond to consumer requests [61, 62]. They provide information, actively engage users, and moderate correspondence. Bots and intelligent agents have also been applied to various other contexts such as social media platforms and conversational platforms to match friends [31], auto-complete text [29], support collaborative search [9], support collaboration in software development [39] or even match individuals with expertise to the right consumers [68]. Kim et al. [37] explored the feasibility of a discussion facilitator bot ('GroupfeedBot') to improve participation in group chat discussions. They found that incorporation of the bot led to more diverse opinions, and participants perceived the group chat sessions as involving fairer and more effective communication. In a survey of open source projects, bots were found to participate in various tasks such as code reviewing, assigning reviewers to pull requests, merging pull requests, and managing dependencies [71]. The developers surveyed requested smarter bots that could learn from previous interactions. Researchers integrated a chatbot into an online gaming community, finding that it was okay for the bot to not always provide sensible interactions and that the bot was helpful in community moderation as well as clarifying community standards [56]. Researchers introduced a bot on Twitter to persuade people to volunteer, finding that understanding a community is important in order to effectively integrate a persuasive bot into it effectively [53]. Chatbots have been used extensively to support online learning in a variety of ways, in particular by facilitating and responding to questions in online learning communities [27, 63]. Researchers have created a model-driven approach to enable the development of chatbots for social learning communities by non-experts. Chatbots have also been integrated into student support channels [5] [47]. In this work, we build on the study of chatbots in communities, specifically examining preferred interaction paradigms for teaching chatbots in communities in which SMEs are observing interactions and are responsible for addressing consumer queries.

2.4 Continuous Improvement

With the widespread adoption and use of bots and intelligent agents, there is an opportunity for these systems to learn from interaction with users and other parties. Users also want smarter bots, that can learn from previous interactions[71]. Human feedback is crucial for the improvement of AI systems. Christiano et al. demonstrated the effectiveness of a human preferences-based approach in solving complex reinforcement learning tasks without access to the reward function [20]. Similarly, Ouyang et al. showed the potential for fine-tuning language models with human feedback to align with user intent on various tasks. They collected a dataset of rankings of model outputs, which they then used to further fine-tune the model using reinforcement learning [48]. These studies demonstrate the value of collecting human feedback to improve AI systems and provide insight into different approaches for doing so. Additionally, there are many approaches that have been described in prior to work to leverage human interaction to improve the machine learning models that facilitate bot behavior [15, 45, 47]. These techniques require various amounts of effort that, in turn, result in potential for improvement. Hancock et al. [30] developed a self-feeding chatbot capable of extracting new training examples from conversation that it takes part in without explicit user effort. Weston [72] studied dialog based language learning, where supervision is given naturally and implicitly as the bot converses with its users. Blenderbot [60] is described as a 'conversational agent that continually learns to responsibly engage'. When conversing with Blenderbot users can provide feedback using a custom interface. The user can downvote a response, after which the feedback is categorized and they are encouraged to provide a free form description of the failure. In a companion study [7] users were provided with various types of feedback such as thumbs up/down, conversational style feedback, type of failure and improved response per failure. The results indicated that feedback targeting particular system modules was more effective than feedback related to the final response. Textual and binary feedback were useful, but not as much as modular feedback. Our goal is to determine which interaction types users prefer when correcting and supervising an intelligent agent that is capable of learning, and also understand the factors that go into that decision.

3 METHODOLOGY

To answer our research questions, we developed scenarios in which a bot delivers partially correct/incomplete information, when responding to a user request. In each of these scenarios, an SME then uses a different interaction technique to teach the bot to deliver a better, more complete answer in the future. The SME uses one of the interaction archetypes discussed above (*Showing*, *Sorting*, *Categorizing*) each of which have different kinds of downstream improvement associated with them. Each archetype is presented in two different contexts, namely HR benefits and open source. See figure 2 for an example scenario, specifically the *Showing* archetype within the HR benefits context. The full set of scenarios are available in Appendix A. In the HR benefits scenario, the user asks about gym benefits for employees; in the open source scenario the user inquires about coding examples from a time series analysis library. In total, we developed 6 scenarios: 2 (context) × 3 (interaction scenario).

3.1 Upstream Conversation vs Downstream Improvement from Teaching Interactions in Conversation

The purpose of teaching interactions is to improve the chatbot by providing additional data. Different types of interaction techniques require various amounts of effort by the SME during teaching (upstream) which in turn translate to different levels of improvement for the chatbot (downstream). In this paper, we refer to the portion of the conversation in which the teaching interaction occurs as "upstream" and the portion of the conversation where the bot has learned as a result of that teaching interaction the "downstream". "Alice" and "Theo" are example consumers within the communities who are asking similar questions at different points in time. In the study we build on work [38] that proposed a taxonomy of

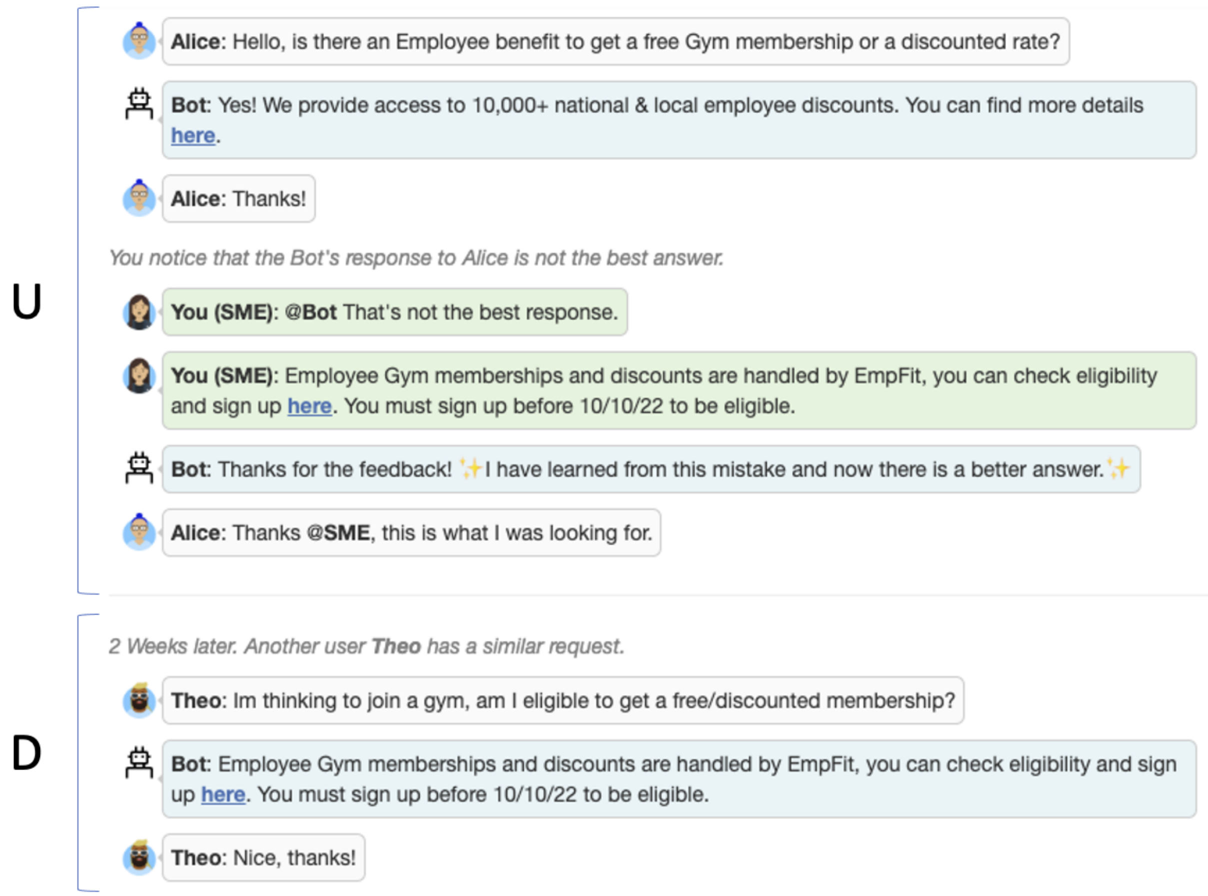


Figure 2: The *Showing* Archetype scenario in the HR benefits context. The scenario is is played out sequentially (one message appears at a time - with this screenshot showing the final view with all messages appearing) to simulate an unfolding chat interaction. The upstream interaction (U) involves user Alice asking the initial question, the bot providing a response, and the SME supervising the bot using a Showing archetype-style interaction. The downstream interaction (D) (2 weeks later) involves a new user Theo, the Bot and the SME. The upstream stage presents the interaction archetype, and the downstream stage presents the learning outcome.

four different types of interaction techniques: categorizing, sorting, showing, and evaluating. In our scenario„ we present our chatbot as a supervised intent-based classifier model since most conversational agents in customer service and other domains are supervised intent based classifers [59]. Three different researchers iterated over scenarios that would accurately reflect interaction types in conversation, given the capabilities of a supervised intent-based classifer, and what the improvement in the bot would look like realistically given the input from the human. We describe the scenarios below.

3.1.1 Categorizing. *Categorizing* is an interaction technique used for regression or classification. It requires the SME to ascribe a label onto the AI generated output as feedback for the AI agent. Prior work that measured the cognitive load of *Categorizing* found that users spent less time on this task than *Showing* and the amount of subjective cognitive load required is less for *Categorizing* than it is for *Showing* [38]. In our scenarios, *Categorizing* is represented upstream as a "thumbs down" label that is assigned to the bot's initial response. The bot then expresses it has learned it is not the best response. It should be apparent that simply by *Categorizing*, while a bot may learn what a wrong answer is, *Categorizing* alone does not teach it the right answer. As a result, in the downstream response, the bot queries the SME for a response. The thumbs down button in the *Categorizing* design was chosen to be simple and easy to use, as one aspect of our inquiry was the amount of effort required by users to provide feedback. It could also be explained that the lack of a partially valid response in the Categorizing design was intentional to test whether users would still prefer this option despite this disadvantage.

3.1.2 Sorting. *Sorting* is an interaction technique that is used for preference elicitation. Ranking-based approaches are precise, and require lower user effort. They are increasingly common approaches for learning reward functions [14, 52]. In our scenarios, the SME invoked the *Sorting* interaction by asking the bot to see other options after the initial request was made by Alice the consumer. The SME then assigned "thumbs up" to one of the other responses presented by the bot, only after the SME asked to see additional options. The bot responds that it has now learned the best response. In the downstream interaction, the bot offers the response that was liked in the upstream conversation to another user who asks a similar question.

3.1.3 Showing. Showing involves demonstrating the correct response [1, 13]. In our scenarios, the SME immediately corrects the chatbot and offers a more correct answer to the user's query in the upstream portion of the conversation. In the downstream portion of the conversation, the bot has now learned the correct answer and offers the corrected response to Theo, who makes a similar request in the downstream portion of the conversation.

3.1.4 Evaluating. Evaluating involves correcting the AI agent by marking good and bad regions during or after the task execution. Because the AI agent in our study was a supervised intent-based classifier, using this interaction technique for conversational agent was not realistic, so we did not include it in our evaluations. In future studies, we can investigate other types of AI agents that are not intent-based, but are generative and driven by foundation models [36] and investigate user interaction when an SME *evaluates* a bot's response.

3.2 Prompt

Upon consenting to participate in the study, users were presented with the prompt that described the lens through which they were instructed to view the scenarios. In order to collect data about SME perspectives and stress that users were not evaluating the bot's intelligence - rather evaluating the entire scenarios from the perspective of a SME, we provided the following prompt at the outset of the experiment.

> In this study you will assume to role of a Subject Matter Expert (SME) at ACME corporation.
> Part of your job is to monitor a community chat channel and answer questions. You have been spending a lot of time on this task and recently an AI Chatbot was added to the channel to help you out! Sometimes the AI chatbot provides incorrect or partially incomplete answers, but the AI chatbot is capable of learning and you need to teach it to respond so that it answers correctly in the future. There are various ways to teach the AI chatbot. You can correct the bot using dialog, like or dislike the bots messages, or probe the bots knowledge.
> During the study you will see pairs of scenarios in which an individual asks either about a coding problem or gym membership benefits. In each scenario you, the SME, help the AI Chatbot to learn using different interaction techniques.
> You need to choose which scenario you prefer in your role as the SME.
> Note: You are not comparing the bots in the two scenarios. You are comparing the entire interaction and the process through which the SME teaches the bot the be more correct. The bots in both scenarios have the same capabilities but react differently depending on the user interaction.

3.3 Pairwise Experiment

We used a pairwise comparison experiment to collect preferences for learning interaction archetypes. Our experiment consisted of randomly showing participants two of the three interaction techniques, both within the same context (open source/HR benefits) until participants saw all of the conditions for both HR benefits and open source (Sorting vs. Showing, Showing vs. Categorizing, Categorizing vs. Sorting). Ordering was randomized for both how users viewed the pairs and how scenarios were presented within a pair. We asked participants about their preferences at two break points. We first asked before users saw the downstream impact of the learning technique, and we asked again after users saw the downstream impact of the learning strategy. We asked participants to select which scenario they preferred and to describe why they had made their selection. This method allowed participants to see all of the conditions and elicit both quantitative and qualitative responses on the desirable traits of one teaching strategy over another.

Pairwise experiments are commonly used in various fields of research to determine participant judgments [49]. Pairwise comparisons could yield more realistic results than likert scales [22] because they take advantage of simple judgments and prioritize a small set of stimuli to learn people's preferences [8, 12]. We performed rank analysis of our pairwise comparisons using the Bradley-Terry model [67], which has been used in other HCI studies to rank pairwise comparisons [2, 8, 57].

3.4 Participants, Task and Procedure

Participants were recruited on Mechanical Turk. Each participant was exposed to all 6 pairwise comparisons, and given the opportunity to choose their preferred scenario after the *upstream* stage of the scenario, where the SME is observed teaching the bot, and again after the *downstream* stage, where the learning outcome is observable. At each point of preference, the participant was also asked to provide a reason for their choice. For attention checks, we had participants describe the differences between each scenario. Two of the coauthors reviewed all reasons, and users who were unable to describe differences between the two scenarios were not included in the analyses. Participants experienced scenarios on a turn-by-turn (each turn being a chat message) basis, with typing indications that were proportional to the amount of text being written, and an indication that the Bot, SME or user were typing. This was done to simulate a realistic chat experience that played out over time. The ordering of scenarios was randomized both in the way they appeared on the page (left/right) and where they appeared in the sequence of the total 6 scenarios. Users first selected a button that would play out one of the upstream interactions of one scenario. They then selected another button to play out the upstream interaction of the other scenario. After seeing the upstream of both scenarios, users made a selection as to which scenario they preferred, why they preferred it, and the observed differences between the two scenarios. They then could play out the remainder of both

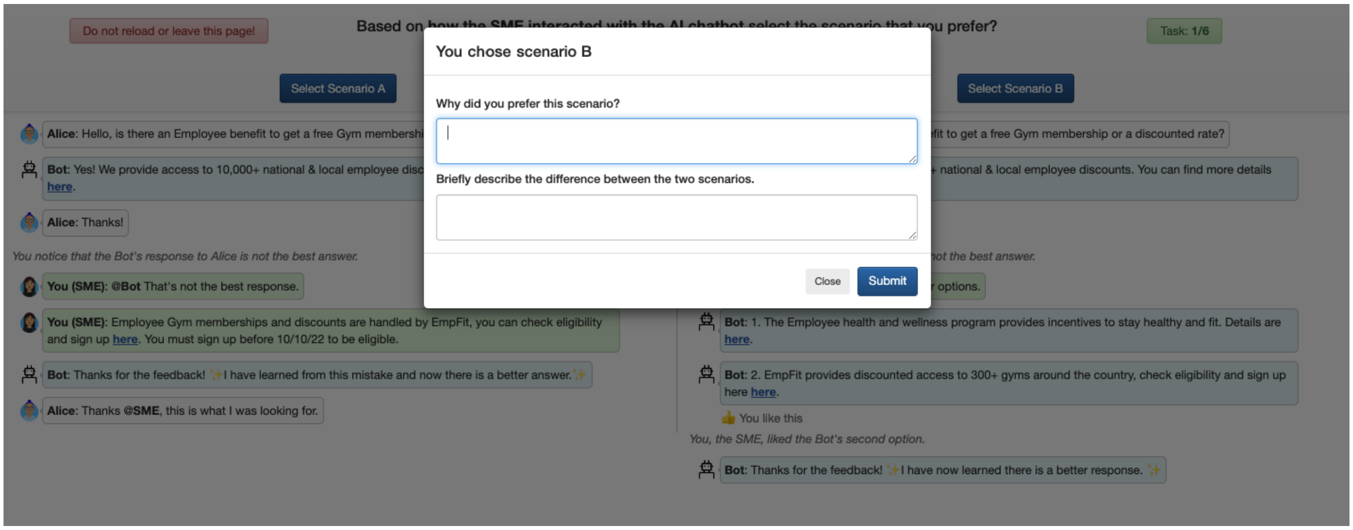


Figure 3: Interface presented to users after selecting a preferred scenario in the upstream phase.

scenarios (one by one) by selecting the appropriate buttons made available on the page. They were again prompted to select their preferred scenario from an SME's perspective, to provide reasoning, and to indicate the differences between each scenario, as seen in Figure 3.

After finishing all comparisons, participants filled out a survey that collected additional demographic data, data on service orientation towards chatbots [8, 34], attitude towards chatbots, and interaction preferences. The overall task took about 30 minutes to complete, and participants were compensated $4.50 for their participation ($9 USD/hr). We deployed a total of 52 tasks on Mechanical Turk and filtered out 4 participants (~8%) who did not pass the attention checks, yielding a final sample of 48 participants. To ensure the quality of the data collected, we also required participants' HIT Approval rate to be greater than 95%, the number of HITS approved to be greater than 500, and to have master qualifications. These qualifications ensured the quality of data being collected.

3.5 Individual Factors Survey

We are interested in how the following individual factors impact preferences for learning strategies: social orientation, prior experience with AI, and long term orientation vs. short term orientation.

3.5.1 Service Orientation. We asked participants questions to determine whether participants viewed their interactions with chatbots as a transaction or a social interaction (service orientation). Prior work makes a distinction between utilitarian orientation and relational orientation [41]. We adapted these questions for our study similar to [8]. An example of one of the likert questions asked to capture a utilitarian service orientation was "Efficient customer service is important to me". We also probed into relational orientation by asking the the following likert question "I think 'small talk' with a chatbot is enjoyable".

3.5.2 Effort Preferences: We also wanted to capture preferences around effort exertion. Long term orientation means the "valuing both past and the future rather than deeming actions important only in their effects in the here and now or the short term" whereas short term orientation emphasizes quick results [32]. We asked participants about their exertion of effort preferences and preferred outcome to capture user effort preferences. We adapted questions from prior work [33] to capture effort preferences, including: "As the subject matter expert, I would rather exert more effort earlier in a conversation if it means that the AI agent/chatbot will learn from my interactions" and "As the subject matter expert, I would rather exert minimum effort in my interactions with the AI agent/chatbot".

3.5.3 Prior exposure to AI and Chatbots. We also wanted to examine how prior exposure to AI agents impacts user preferences of interaction techniques. We asked participants about their prior exposure to AI technologies and prior experience and interactions with chatbots.

4 RESULTS

In this section, we describe participants' preferences for learning technique strategies and the underlying reasons (**RQ1**), We also explore how individual and task-related factors impact these preferences (**RQ2**) and look at how archetype preferences changed between upstream and downstream stages of the scenarios (**RQ3**). Our results include data from both scenarios (HR benefits and Open Source) since interaction preference rankings for both scenarios yielded the following: *Showing* > *Sorting* > *Categorizing*. These results are promising since they show that our findings would generalize across different domains.

4.1 Preferences of Learning Techniques and Reasons

We used the Bradley Terry Model [67] to compare preferences, as it allows ranking of a set of paired object comparisons by estimating "ability scores". Prior HCI studies have used the Bradley Terry Model [2, 8, 57]. For each interaction type, the model generates a p-value

Conversation Phase	Preferred vs. Rejected Learning Interaction	Open Source p-value	Benefts p-value	All Data p-value
Upstream	Showing vs. Sorting	0.090	0.633	0.500
	Showing vs. Categorizing	0.000**	0.000**	0.000**
	Sorting vs. Categorizing	0.000**	0.000**	0.000**
Downstream	Showing vs. Sorting	0.000**	0.000**	0.000**
	Showing vs. Categorizing	0.000**	0.000**	0.000**
	Sorting vs. Categorizing	0.000**	0.007*	0.000**

Table 1: Significant values, after Bonferroni adjustment ($p < 0.05/3$), are noted with *. $p < 0.005/3$ noted with **

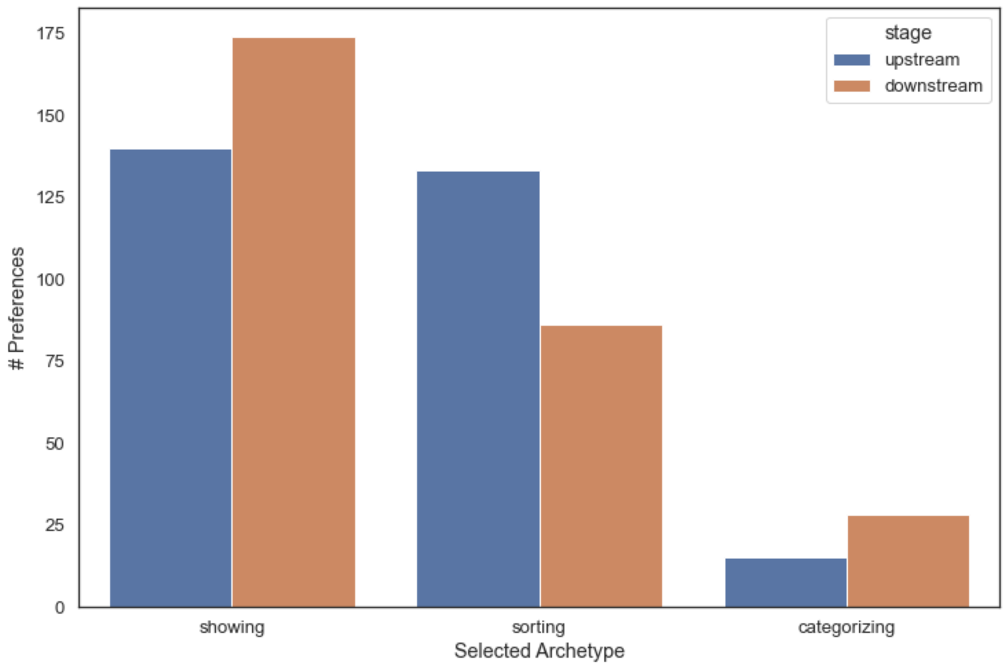


Figure 4: Archetype preferences grouped by scenario stage.

for a pairwise comparison. We used a Bonferroni correction to take into account the number of comparisons. The results for the pairwise comparisons are shown in Table 1.

We found that users rated the *Showing* interaction most highly when they saw the downstream impact. In the Upstream conversations, there was no significant difference between *Showing* and *Sorting* while in the Downstream conversation participants preferred the *Showing* strategy significantly more than the *Sorting* strategy. *Categorizing* was selected significantly less in both upstream and downstream preferences compared to both *Sorting* and *Showing*. Figure 4 shows the overall count of preferences for each of the teaching archetypes grouped by stage.

4.2 Reasons for Preferences

For each preference, we asked users to provide a reason for their preference as well as identify the difference between the two scenarios as an attention check. Two authors independently coded the responses to identify strengths and weakness associated with each learning archetype, for both upstream and downstream interactions. After two iterations, a final code book was decided, which is presented in Table 2 and Table 3. Emergent themes are summarized in the next section. Theme names are bolded for reference.

4.2.1 Upstream Preference: Categorizing. In terms of strengths, participants indicated that *Categorizing*, which only involved a thumbs up/down from the SME upstream, exhibited **ease of use** from the SMEs point of view (based on the simplicity of the interaction), and was **not invasive** to the interaction between the user and the bot. Some participants expressed that the approach simply **prevents bad behavior**, which was a reasonable outcome given the effort involved.

However *Categorizing* was the least preferred learning technique and the following themes emerged as weaknesses of the approach during upstream. Participants expressed concern that there was **no answer** provided to the user who first asked the question, and that the teaching interaction exhibited a **negative orientation**, characterized as 'passive aggressive' and possibly leaving a 'bad impression' on the asker. Note that in the *Categorizing* scenario the SME corrects the bot using only a dislike, and does not provide a corrected answer in the upstream interaction. In addition participants noted the **limited teaching scope** and an **unclear learning outcome** indicating a lack of confidence that the bot would learn from the interaction with the SME.

4.2.2 Upstream Preference: Sorting. Starting with strengths, in the upstream interaction participants selected *Sorting*, which included asking the bot for multiple options and choosing a favorite among them using a thumbs up/down, as their preferred teaching archetype due to the SMEs' **ability to query** the bots knowledge, and the subsequent **positive orientation** of the teaching interaction. Participants noted that querying or probing the bot allowed the SME to understand the scope of the bots capabilities, and exposed the asker to additional information. The teaching interaction was described as 'friendly', 'interrogative' and 'encouraging'. In a more general sense, participants considered the **interactive teaching** paradigm to be a strength of the *Sorting* archetype, alluding to interaction between the SME and the bot during the supervision task. Participants described that the *Sorting* archetype was **not invasive** and noted that there was an **answer provided** to the asker. Finally, some participants considered the **shared effort** between SME and bot as a strength, describing that both the SME and the Bot play a role in the learning interaction.

Participants who rejected the *Sorting* archetype did so primarily because the learning interaction was **limited to existing knowledge**, meaning that the bot could only learn from the results of the probing interaction. This limitation was designed into the *Sorting* scenario. To a lesser extent, participants noted that there was an **unclear learning outcome**, indicating that participants were not confident about what the bot had learned during its interaction with the SME.

4.2.3 Upstream Preference: Showing. *Showing* was the most preferred learning technique throughout the study and participants highlighted various strengths. Many participants mentioned that the *Showing* interaction provided an **explicit answer** to the user asking the original question, in addition to supervising the bot. Moreover the interaction was deemed to be **more precise** and required **fewer steps** to complete. In a broader sense participants preferred *Showing* because it was an **effective teaching** paradigm and exhibited a **clear learning outcome** that was 'easier to follow'. Participants were confident that the interaction would result in the bot answering correctly in the future. Some participants also preferred the **dialog interaction** implemented in the *Showing* scenario.

In those cases where *Showing* was rejected during the upstream stage, participants cited that it was **invasive** and exhibited a **negative orientation**. The SME was described as 'scolding', 'lecturing' and 'less friendly more condescending' to the bot. It was also noted that the *Showing* interaction **required more effort**.

4.2.4 Downstream Preference: Categorizing. A significant subset of participants noted that **asking for help** was a strength of the *Categorizing* archetype during downstream. This group preferred that the bot 'knew its limitations' and asked for assistance when needed rather than the SME having to actively monitor user-bot interactions.

Categorizing was the also the least preferred strategy in downstream and the following weaknesses emerged. Most notably, participants indicated that *Categorizing* required **more effort** on behalf of the SME, as it was necessary to respond to the bots request for assistance. The bot was also **perceived as less intelligent** and **failed to learn the correct response** based on its interaction with the SME.

4.2.5 Downstream Preference: Sorting. Participants preferred *Sorting* at downstream due to the **shared effort** between the bot and the SME when addressing the user. The bot 'did half the work' which then required only 'a little input from the human'. On a related note, it was indicated that the **split response** (the bot provided one answer, and the SME provided another) was easier to read. In general, participants preferred the **user experience** of the *Sorting* archetype, describing 'better conversation' and dialog that 'flows much better'. In terms of weaknesses, participants were concerned with that the SME had to exert **more effort** to fully address the users request.

4.2.6 Downstream Preference: Showing. *Showing* was also the most preferred archetype at downstream and we were unable to identify any weaknesses during the thematic analysis. In terms of strengths participants focused on effort, indicating that *Showing* required **less effort later** in the bots life-cycle as it was supervised in a more **comprehensive** manner earlier in the scenario. Participants also noted that the downstream interaction was **less invasive** based on the observation that the bot was able to respond to the user without assistance.

4.3 Individual Factors

In addition to collecting user preferences, we collected individual factors that captured users service orientation and effort preferences (**RQ2**). The statistically significant values can be seen in Table 4. We describe the correlations for each Interaction Archetype below.

4.3.1 Sorting. We found that users who preferred the *Sorting* learning technique were more familiar to chat technology. The preference elicitation technique that we represented to users was similar to how chatbots behave when they provide options to users [8]. Prior work also found that users who are more familiar with chatbot technologies prefer to be given options, so it is unsurprising that those who were familiar with chatbot technologies preferred sorting. There was also a positive correlation between those who had a relational orientation and those who preferred *Sorting* $p < 0.05$, as well a marginally significant correlation between those who preferred *Sorting* and individuals who preferred the probing strategy (i.e. "show me more options").

4.3.2 Showing. Participants who preferred *Showing* as a teaching strategy did not want to exert minimum effort when interacting with the bot $p < 0.005$. Conversely, there is a significant positive correlation between preferring *Showing* and wanting to exert effort early to teach the bot $p < 0.05$. There are marginally significant ($p < 0.1$) negative correlations between the preference of *Showing* to

Interaction	Archetype	Strength	Example Reason
Upstream	Categorizing	Ease of use	Its much easier to just thumbs down the bot and let it figure out the answer
		Not invasive	There's less interaction from the SME. It seems intrusive so the less of it the better for the user.
		Prevents bad behavior	I actively correct the bot by using dislike which should stop it from answering with this response in the future.
	Sorting	Ability to query	the bot was prompted and provided additional information that was directed more towards the poster's original request.
		Positive Orientation	the bot has been encouraged by the user to provide more answers so that it can learn quickly
		Interactive Teaching	Chat bot has details and ready to share with the customer and updates the feedback for future purpose. It is more interactive.
		Answer Provided	It gives the bot the correct answer for future use.
		Shared effort	Less effort. The bot came up with the examples and all i had to do was thumbs up the correct response
		Not Invasive	It interrupted less into the conversation between Alice and the bot.
	Showing	Effective teaching	I'd rather provide the most information possible to the user as well as teach the bot more instead of just approving of another bot response
		Explicit answer	Better information to the customer with this training.
		Fewer Steps	required less steps to complete the task with the help of SME.
		More Precise	SME provided a more detailed response and I feel it learned from it on Scenario A.
		Clear learning outcome	I like A since I was able to give the examples both for the customer and for the AI. On B just the dislike button doesn't give me confidence that it is learning what I want.
		Dialog Interaction	I prefer the verbal interaction to the simple use of the "dislike" function.
Downstream	Categorizing	Asking for help	The bot knew its limitations and asked for help.
	Sorting	Shared effort	the bot did half the work and only needed the other example to be provided which serve two purposes to inform the user as well as to teach the bot to provide multiple examples when possible.
		User Experience	It flows much better. The user would have a better experience.
		Split response	The examples are split and more easy to read.
	Showing	Comprehensive	has learned from it's mistake from last time. It did not require someone to interfere in the process.
		Less effort later	I did more work to assist the bot in the beginning in Scenario B. I had to help the bot later on in Scenario A because I wasn't as helpful from the start.
		Less invasive	In the first scenario SME asks for some explanations but in the second the chatbot gives the perfect answer without the interference of SME.

Table 2: Emergent Themes from reasons why individuals selected learning techniques.

Interaction	Archetype	Weakness	Example Reason
Upstream	Categorizing	No Answer	Scenario A (Categorizing) is too passive aggressive and Alice never ends up with the right response.
		Negative orientation	The "dislike" button Scenario B with the thumbs down icon turns me off and I think it makes a bad impression on Alice the customer
		Unclear learning outcome	I think the bot in Scenario A was taught the specific correct answer and am unsure what it learns from the dislike button in the other scenario.
		Limited teaching scope	It isn't given any additional information regarding how it should answer the question. In Scenario B the bot is pushed to provide more information and is able to come up with a good answer that the SME tells it to use.
	Sorting	Limited to existing knowledge	I'd rather provide the most information possible to the user as well as teach the bot more instead of just approving of another bot response.
		Unclear learning outcome	In Scenario A (Showing) the bot is given information that it will be able to use in the future. In Scenario B it's prompted to find out the information and then suggested which parts to use in the future. This is less clear.
	Showing	More effort	it took more effort on the part of the SME
		Negative orientation	looks like you're scolding the bot
		Invasive	In a scenario SME provided the information for the bot and ended up acting in the place of the bot interjecting multiple lines into the conversation.
Downstream	Categorizing	Bot perceived as less intelligent	The Bot has provided partial response that was accurate and did not sound as incompetent.
		More effort	The bot gave half of the correct response in Scenario B I needed to provide the entire answer in Scenario A.
		Failed to learn the correct response	the bot didn't even try to help the user and instead totally depended on the SME to give the best response and didn't reply that it learned anything
	Sorting	More effort	The SME didn't have to give feedback 2 weeks later because the bot already learned the right response previously

Table 3: Emergent Themes from reasons why individuals did not select learning techniques.

Interaction Archetype	Individual Factor	Correlation
Sorting	Familiar to Chat Technology	0.31^{**}
	Relational Orientation	0.29^{**}
	Probing	0.26^{*}
Showing	Effort Minimum Preferred	-0.42^{***}
	Effort Early to Yield Long Term Results	0.32^{**}
	Relational Orientation	-0.25^{*}
	Familiar to Chat Technologies	-0.25^{*}
	Frequently use chat Technologies	-0.27^{*}
Categorizing	Effort Minimum Preferred	0.40^{**}
	Effort Early to Yield Long Term Results	-0.34^{**}

Table 4: Spearman correlations between user preferences for interaction archetype and individual factors collected, $p < 0.005^{***}$, $p < 0.05^{**}$, $p < 0.1^{*}$.

social orientation, emojis, and familiarity and frequency of exposure to chat technology.

4.3.3 Categorizing. Participants who preferred *Categorizing* as a teaching strategy were likely to want to exert minimum effort when interacting with the bot $p < 0.005$. Conversely there is a significant negative correlation between preferring *Categorizing* and wanting to exert effort early to teach the bot $p < 0.05$.

These individual factor correlations support the notion that those individuals who preferred to exert more effort earlier, to reap rewards later, preferred the *Showing* interaction, whereas those who did not prefer to exert effort to yield long term results preferred the *Categorizing* interaction. We also find that users who selected *Categorizing* prefer to exert the least amount of effort. Our other findings suggest that those who have a more relational orientation with chatbots and are familiar with chatbot technologies prefer the *Sorting* interaction.

4.4 Changing Preferences Upstream to Downstream

The study design made it possible for participants to choose a preferred teaching archetype after an initial *upstream* stage of the scenario, where the SME is observed teaching the bot, and again after a *downstream* stage, where learning outcomes are observable in a later interaction. As such, it was possible to analyze those cases where participants changed their preference upon observing downstream outcomes **(RQ3)**.

Figure 5 provides a map depicting the occurrence of upstream archetype selections with corresponding downstream selections. When participants chose *Showing* in the upstream stage, they also chose *Showing* in the downstream stage in 92% of cases, indicating a consistent preference for this archetype across both stages. When participants did switch preference from *Showing*, they were more likely to choose the *Sorting* archetype over *Categorizing*.

Preferences across the remaining two archetypes, *Sorting* and *Categorizing*, exhibited greater variation between upstream and downstream. When choosing the *Sorting* archetype after the upstream stage, participants kept that same choice in just 56% of cases at downstream, choosing instead *Showing* in 30% of cases and *Categorizing* the remaining 14%. Participants were less consistent when *Categorizing* was chosen during upstream interactions, only keeping the same preference in 40% of cases at downstream, choosing instead *Showing* or *Sorting* at downstream with a preference for showing.

Overall, preference for the *Showing* and *Categorizing* archetypes increased in the downstream stage, when compared to upstream preference counts, while preference for the *Sorting* archetype decreased across stages. This information is depicted in figure 4.

Upon analysis of participant-provided reasons for choosing a particular downstream archetype, in those cases where the upstream preference differed, some themes were evident. The tendency to select *Showing* downstream, changed from *Categorizing* upstream, was primarily related to the learning outcome. Only the bot was involved in the downstream interaction with the user. A number of participants who chose *Categorizing* downstream, changing from *Sorting* upstream, commented that the bot asking the SME for help was a factor in the decision.

> *The bot knew it didn't have a clear answer so asked me to provide it.*
> -Participant #97an5j3p, *Sorting* vs. *Categorizing* (*Sorting* selected upstream, *Categorizing* selected downstream)

5 LIMITATIONS

As with all Mechanical Turk studies, there are limitations to this work. First, the study was conducted on Mechanical Turk and is limited to the subjective preferences of Mechanical Turkers, which are not actual SME's. We did however explicitly provide cues (SME (you)) and context to encourage participants to put themselves in the place of an SME. While this is a limitation, we believe our required qualifications for recruitment yielded higher quality responses. Additionally, previous research has shown that crowd workers can perform well and achieve accuracy levels similar to experts in product reviewing [69]. In a study by Wang et al., they investigated the impact of introducing monetary payments on review quality and found no significant differences in quality between paid and unpaid reviews. Additionally, Mason et al. provided guidance on ensuring high-quality work for behavioral research, which we followed [46]. Our participants were required to have a HIT Approval rate of over 95%, have approved more than 500 HITS, and possess master qualifications, ensuring high-quality data collection. Furthermore, the prompts we used did not require any specific subject matter expertise. Furthermore, while we were exploring downstream impact of interaction paradigms, we acknowledge that in the real world, the value of teaching a bot would play out over a longer period of time. However, we believe the way in which we designed our scenarios still show the potential downstream improvements to the participants and allow them to opine and express their preferences in the given scenarios. We only investigate scenarios in which the interaction paradigms yield a successful outcome in the downstream. We acknowledge that preferences may not be the same in instances where the interaction paradigms were unsuccessful in teaching the AI agent the correct response. We also make clear in our study and wording (through provided prompt) throughout the study that we are asking users to consider the perspective of the SME. Of course, the presence of the customer (Theo/Alice) should be considered since we are investigating SME preferences in the context of a community setting. Prior work shows that the presence of an audience when interacting with a conversation agent impacts user experience [16], so we cannot consider the SME's preferences in a community without considering Alice or Theo as the audience. While some of the responses consider Theo and Alice's impressions, all participants included in the study followed directions and participated as the SME as instructed in the prompts.

Lastly, we based our designs of interaction paradigms on heuristics discussed in previous literature [38], and spent significant effort translating them to the conversational context through numerous design iterations. While we believe our designs effectively represent the chatbot-SME interaction for *Categorizing*, *Sorting*, and *Showing*, we acknowledge that alternative representations and designs could also be effective and impact how users provide feedback. Therefore, future work can explore various interaction paradigms and designs beyond the ones explored in this study.

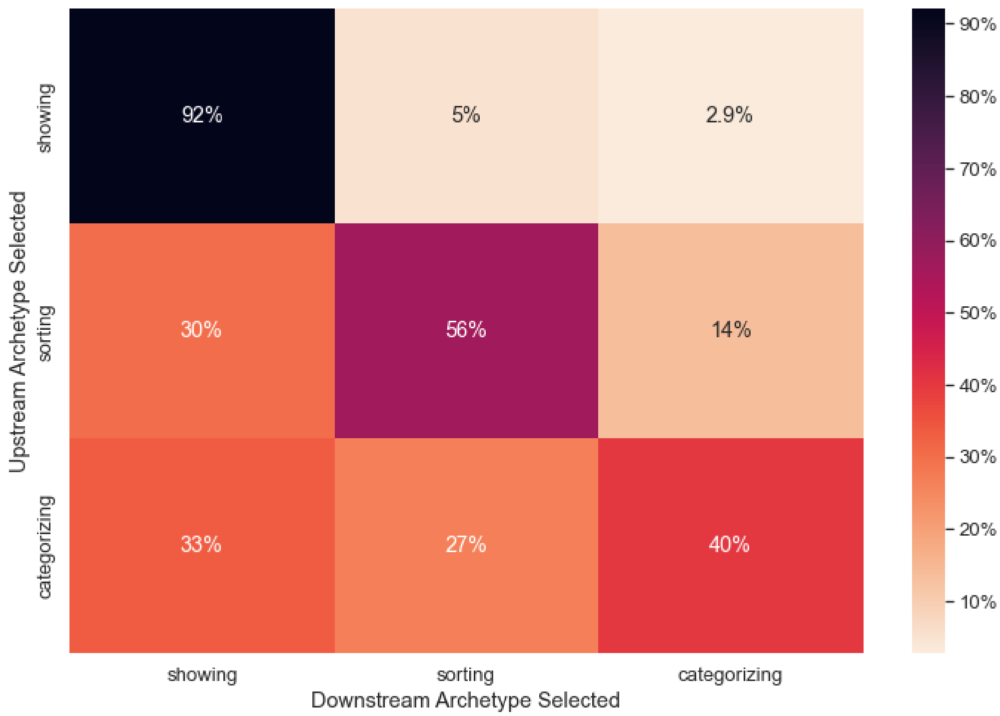


Figure 5: A heat-map depicting how participants changed their interaction archetype preferences when considering downstream outcomes.

6 DISCUSSION

In **RQ1**, we were interested in investigating SME preferences in communities for interaction paradigms. We found that users preferred *Showing* followed by *Sorting* and *Categorizing*. Beyond capturing preferences, we were able to capture reasons why users made their preferences. In this section, we further discuss the reasons for user's selections and implications of human-AI interaction with AI agents which are capable of learning within online communities. We contextualize our findings within prior work on human-AI guidelines [6] and how that applies to human-AI teaching paradigms within communities as well as prior findings on human-robot interaction [65].

6.1 Transparency and Clear Learning Outcomes for Human-Agent Teaching Techniques

"Making clear what the system can do" and "how well the system can do what it can do" are important human-AI interaction guidelines [6]. Beyond making clear what the AI is capable of, our findings show it is important to make clear what an AI agent can learn and how well it can learn. In a study of human-bot interaction in open source projects Wessel et al. [71] identified a 'lack of information on how to interact with the bot' as a recurrent challenge.

The clarity of the learning outcome emerged as a broad and recurring theme that impacted user preferences across all archetypes. Many participants (14) expressed that *Categorizing*, particularly in the upstream conversation had an unclear learning outcome. These participants were uncertain what if anything the Bot would learn from the categorizing interaction (disliking the Bot's response), and consequently they preferred the more explicit archetypes, primarily *Showing*. One participant described the general problem as "A learns only that its answer is not the best but not why." Participants strongly preferred *Showing* because of the effective teaching interaction (19) and the clear learning outcome (8). In this archetype the SME corrected the Bot explicitly with a more accurate answer than was initially provided. One participant who preferred the *Showing* archetype said that "I feel as though the bot has learned for it's next conversation appropriately". Feedback captured during the study also indicated that the audience effect was intertwined with the desire for a more explicit teaching interaction. Participants preferred archetypes with clearer learning outcomes because the audience, Alice or Theo, also benefited from this clarity. One participant stated, on this topic, "Alice got the answer without hassle. The bot learned better". Overall our results indicate that when designing human-AI teaching interactions, its desirable that users are allowed to express feedback in a way that minimizes uncertainty in the learning outcome. Furthermore the interaction style of the Bot should also be designed to support predictable learning. A bot's responses and feedback should be explicit with what it has learned after an interaction. One participant complained that " there's no real proof that they learned anything" after experiencing the upstream *Categorizing* scenario, another said "Scenario B was vague and the bot learned nothing". Instead of responding, "Thank you I have now learned the best response", a better designed bot might specifically itemize and list what it has now learned and how it will

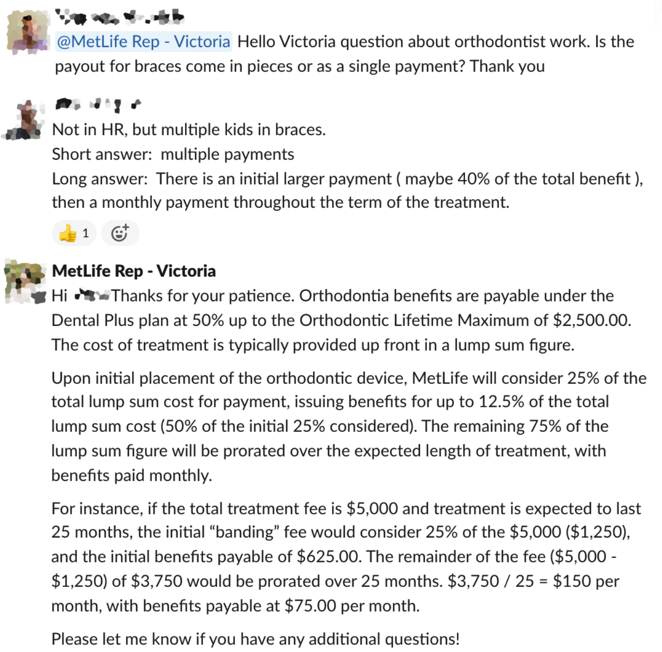


Figure 6: HR Benefits example of community member who is not an SME but just another member of the community responding to another user's query. Future designs can consider learning not just from SMEs but other active users that provide valuable information.

answer the question in the future making the learning outcome more clear to the SME. This type of feedback would make interaction paradigms like *Categorizing* and *Sorting* more attractive to users who rejected them for having an unclear learning outcome.

6.2 Negative Feedback in Learning Interactions

A recurrent theme that emerged from participant feedback was the orientation of the teaching interaction, whether it be positive or negative. The scenarios presented to the participants required that the SME correct the Bot and participants exhibited a range of feedback on the topic of perceived orientation. Some participants indicated that *Categorizing* exhibited a negative orientation, describing the interaction as "passive agressive" and "makes a bad impression on Alice the customer". In some cases the *Showing* archetype was also described negatively. Participants noted that the SME was "lecturing the bot", "scolding the bot" and the interaction "seemed less friendly more condescending" as compared to the *Sorting* interaction. Interestingly the *Sorting* archetype received mixed feedback, with most participants expressing a positive orientation. When considering a *Sorting* scenario participants described that "There was positive reinforcement which makes me think the bot will improve faster using this method" and "the bot thinks on its own". When describing the interactive learning interaction (the SME probes the bot and provides a like on a better response), participants highlighted that the teaching "seemed friendly" and "the bot was encouraged". Alternatively one participant noted that "from the user's perspective it may seem like an argument".

Prior work has found that in question-and-answer chatbot interaction users have a tendency to anthropomorphize the agent and engage in chit-chat [43], indicating that the user has recognized the agent as a 'subject'. Feedback from our participants supports the finding even though participants were considering bot interaction in a passive manner. It was clear that participants anthropomorphized the bot and sympathized with its attempts to learn from its mistakes, preferring when the SME interacted positively, and calling out instances of negative orientation. Thomaz & Breazeal [65] found that humans prefer to give positive feedback and motivation to agents because they fall into a "natural teaching interaction" and treat the AI agent as a social entity, believing that positive feedback leads to an agent learning better. Lee et al. [40] also found that when interacting with a chatbot designed to give and receive care, users exhibited socially desirable reactions, such as politeness towards machines as social actors, as well as emotional empathy, i.e., the ability to 'feel for' the chatbot. Our results seem to suggest this same preference exists in the context of interactive human supervision of AI agents, at least for some subset of user personalities. From a design point of view it may be preferable to design teaching and supervision interaction through a lens of positive reinforcement.

6.3 Querying the Bot

A primary reason cited for users' preferences, when comparing interaction techniques with *Sorting*, was the fact that the SME had the ability to probe the chatbot for additional responses. One participant sumamrized the strength as "I liked that the bot was able to give more information when asked". Six participants mentioned probing as an upstream strength of the *Sorting* archetype, and One highlighted it as a strength at downstream. However, a similar theme was also expressed as the bot being a "better conversationalist" and the SME having a "more interactive" session with the bot. Existing work [8] exploring end-user repair strategies for chatbot mistakes found that users found chatbots more desirable when they provided more options if the initial answer is not correct because users had a better understanding of what the AI was capable of. In a similar vein Blenderbot [60] provides users with a 'look inside' feature that explains how the current dialog response was generated based on memory, search query and search responses.

Our findings indicate that participants prefer when the SME interacted and queried the bot for more information when teaching. Participants also noted that observing the SME interactively probing the bots knowledge might be helpful for the user asking the question, as well as for the SME as a gauge of the bots performance/knowledge. One of the human-AI interaction guidelines recommended by Amershi el al. [6] is to allow scoping services when in doubt in cases where the AI is incorrect.

6.4 Disruptiveness

Disruptiveness was often mentioned by users when providing reasons for their learning interaction preferences. For example, when *Showing* was the preferred learning technique, users mentioned the SME and the bot being at odds with one another in their interaction. One participant described the SME "ended up acting in the place of the bot interjecting multiple lines into the conversation" and another said "there's too much back and forth between the bot and SME". Prior work has described the phenomenon of Disruptiveness as "For some phenomenon A to be considered disruptive for another phenomenon B, A's effects on B, direct or indirect, must be such that B is not able to continue to exist and to transition smoothly to another state, given the rules governing B's nature" [10].

Our study feedback indicated a prevailing theme that the learning interaction was disruptive to the original conversation between the user and the bot. Participants also mentioned not wanting the user to have to wait while the SME corrected the bot, which highlights the consideration of the presence of an "audience".

Prior work has found that the existance of an audience when interacting with an AI agents has significant effects on a user's self-reported experience [16]. Similarly, we find that participants consider the "audience" when interacting with the agent, being conscious when the SME appears to be "scolding" or 'at odds with" the AI agent, or specifically wasting the users time when teaching.

During interaction with an AI agent, showing contextually relevant information to the user is important [6]. The additional interaction between the SME and the bot is irrelevant to the user. A potential solution is to provide a private interface only visible to the SME when interacting with the agent. The end user may only be interested in receiving an answer, while the SME is motivated to provide detailed comprehensive feedback which would be less disruptive in a private interface between the SME and the AI system. While we cannot say this will definitively improve the user-experience, future work can consider user preferences both from the perspective of the customer and the SME in human-AI interactions.

6.5 Future Opportunities for The Community

One of the main distinctions of our work from prior work is that we are investigating teaching paradigms within the context of community and examining perceptions of the SME. "Invasiveness" or the degree to which the teaching paradigm was disruptive, or the appearance that the SME and the bot were "at odds" with one another impacted users' perceptions of the interaction. While a teaching paradigm might be appropriate for a dyadic human-AI interactive interaction, it may not be appropriate within the community context. We believe our design recommendations take into account this distinction, particularly the recommendation to provide a second interface for SME to bot interaction. In our study we make the distinction between two different types of users, the SME and the consumer. In reality, however, many communities include active users who could also potentially contribute to the teaching the AI agent based on their domain knowledge as opposed to newcomers or more passive users. In those instances, a secondary interface might not make sense as the teaching paradigms would be obfuscated from the consumers who also possess the domain knowledge to contribute. Furthermore, while the *Categorizing* interaction was the least preferred by the SME, it might be the most appropriate kind of interactions for more passive users in the community who want to contribute, but not exert effort. *Categorizing* was appreciated as a low effort way to teach the bot in the community. While SMEs are specifically charged with helping community in core ways, it is possible that lower effort teaching interaction styles could still be used by other members of the community who are not as incentivized to be of assistance as SMEs. There is an opportunity for future research to investigate types of consumers and their perceptions/strategies of navigating the interactions within such communities.

7 CONCLUSION

Through a scenario based study, we evaluated user preferences across three interaction archetypes (Showing, Sorting, and Categorizing) designed for supervising AI bots in a community context. The Evaluation archetype from [38] was not included in the study because the interaction paradigm was not plausible for a supervised intent based classifier which is what we represent in our scenarios. Given the tendency towards generative AI, providing fine grained feedback through *evaluation* is an element of future work we hope to explore. The scenarios incorporated both an upstream learning interaction, and a downstream interaction where the learning outcome could be observed. We found that users preferred the *Showing* archetype, meaning that correct behavior is explicitly demonstrated to the bot, despite the additional effort involved when compared to alternative archetypes. We also gathered user opinions and identified a set of strengths and weaknesses associated with each archetype. Based on our findings, we discuss interaction paradigms in a human-AI interactive community. Our research shows the importance of studying human-AI interaction within the community context, in addition to those done in a dyadic context.

REFERENCES

[1] Pieter Abbeel and Andrew Y Ng. 2004. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the twenty-first international conference on Machine learning*. 1.

[2] Ahmed Al Maimani and Anne Roudaut. 2017. Frozen Suit: Designing a Changeable Stiffness Suit and Its Application to Haptic Games. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 2440–2448. https://doi.org/10.1145/3025453.3025655

[3] Saeed Alahmari, Dmitry Goldgof, Lawrence Hall, Palak Dave, Hady Ahmady Phoulady, and Peter Mouton. 2018. Iterative deep learning based unbiased stereology with human-in-the-loop. In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 665–670.

[4] Adam Alami and Andrzej Wąsowski. 2019. Affiliated participation in open source communities. In *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*. IEEE, 1–11.

[5] Sara Alqaidi, Waad Alharbi, and Omaima Almatrafi. 2021. A support system for formal college students: A case study of a community-based app augmented with a chatbot. In *2021 19th International Conference on Information Technology Based Higher Education and Training (ITHET)*. IEEE, 01–05.

[6] Saleema Amershi, Dan Weld, Mihaela Vorvoreanu, Adam Fourney, Besmira Nushi, Penny Collisson, Jina Suh, Shamsi Iqbal, Paul N Bennett, Kori Inkpen, et al. 2019. Guidelines for human-AI interaction. In *Proceedings of the 2019 chi conference on human factors in computing systems*. 1–13.

[7] Kushal Arora, Kurt Shuster, Sainbayar Sukhbaatar, and Jason Weston. 2022. Director: Generator-classifiers for supervised language modeling. *arXiv preprint arXiv:2206.07694* (2022).

[8] Zahra Ashktorab, Mohit Jain, Q Vera Liao, and Justin D Weisz. 2019. Resilient chatbots: Repair strategy preferences for conversational breakdowns. In *Proceedings of the 2019 CHI conference on human factors in computing systems*. 1–12.

[9] Sandeep Avula, Gordon Chadwick, Jaime Arguello, and Robert Capra. 2018. Searchbots: User engagement with chatbots during collaborative search. In *Proceedings of the 2018 conference on human information interaction & retrieval*. 52–61.

[10] Dina Babushkina and Athanasios Votsis. 2022. Disruption, technology and the question of (artificial) identity. *AI and Ethics* 2, 4 (2022), 611–622.

[11] Mehdi Bahrami and Wei-Peng Chen. 2019. WATAPI: composing web API specification from API documentations through an intelligent and interactive annotation tool. In *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 4573–4578.

[12] Gilles Bailly, Antti Oulasvirta, Timo Kötzing, and Sabrina Hoppe. 2013. Menuoptimizer: Interactive optimization of menu systems. In *Proceedings of the 26th annual ACM symposium on User interface software and technology*. 331–342.

[13] Michael Bain and Claude Sammut. 1995. A Framework for Behavioural Cloning.. In *Machine Intelligence 15*. 103–129.

[14] Erdem Bıyık, Malayandi Palan, Nicholas C Landolfi, Dylan P Losey, and Dorsa Sadigh. 2019. Asking easy questions: A user-friendly approach to active reward learning. *arXiv preprint arXiv:1910.04365* (2019).

[15] Maya Cakmak and Andrea L Thomaz. 2012. Designing robot learners that ask good questions. In *2012 7th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 17–24.

[16] Heloisa Candello, Claudio Pinhanez, Mauro Pichiliani, Paulo Cavalin, Flavio Figueiredo, Marisa Vasconcelos, and Haylla Do Carmo. 2019. The effect of audiences on the user experience with conversational interfaces in physical spaces. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.

[17] Heloisa Candello, Mairieli Wessel, and Claudio Pinhanez. 2020. Unveiling Practices and Challenges of Machine Teachers of Customer Service Conversational Systems. In *Proceedings of the 2nd Conference on Conversational User Interfaces* (Bilbao, Spain) *(CUI '20)*. Association for Computing Machinery, New York, NY, USA, Article 32, 2 pages. https://doi.org/10.1145/3405755.3406173

[18] Liwei Chan, Yi-Chi Liao, George B Mo, John J Dudley, Chun-Lien Cheng, Per Ola Kristensson, and Antti Oulasvirta. 2022. Investigating Positive and Negative Qualities of Human-in-the-Loop Optimization for Designing Interaction Techniques. In *CHI Conference on Human Factors in Computing Systems*. 1–14.

[19] Preetha Chatterjee, Kostadin Damevski, Lori Pollock, Vinay Augustine, and Nicholas A Kraft. 2019. Exploratory study of slack q&a chats as a mining source for software engineering tools. In *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*. IEEE, 490–501.

[20] Paul F Christiano, Jan Leike, Tom Brown, Miljan Martic, Shane Legg, and Dario Amodei. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems* 30 (2017).

[21] Kevin Crowston and James Howison. 2006. Assessing the health of open source communities. *Computer* 39, 5 (2006), 89–91.

[22] Regina Dittrich, Brian Francis, Reinhold Hatzinger, and Walter Katzenbeisser. 2007. A paired comparison approach for the analysis of sets of Likert-scale responses. *Statistical Modelling* 7, 1 (2007), 3–28.

[23] Nicolas Ducheneaut. 2005. Socialization in an open source software community: A socio-technical analysis. *Computer Supported Cooperative Work (CSCW)* 14, 4 (2005), 323–368.

[24] Jerry Alan Fails and D Olsen. 2003. *Image processing with crayons*. Ph.D. Dissertation. Brigham Young University. Department of Computer Science.

[25] Amir Feder, Danny Vainstein, Roni Rosenfeld, Tzvika Hartman, Avinatan Hassidim, and Yossi Matias. 2020. Active deep learning to detect demographic traits in free-form clinical notes. *Journal of Biomedical Informatics* 107 (2020), 103436.

[26] Jasper Feine, Stefan Morana, and Alexander Maedche. 2020. Designing Interactive Chatbot Development Systems.. In *ICIS*.

[27] Donghui Feng, Erin Shaw, Jihie Kim, and Eduard Hovy. 2006. An intelligent discussion-bot for answering student queries in threaded discussions. In *Proceedings of the 11th international conference on Intelligent user interfaces*. 171–177.

[28] Rebecca Fiebrink and Perry R Cook. 2010. The Wekinator: a system for real-time, interactive machine learning in music. In *Proceedings of The Eleventh International Society for Music Information Retrieval Conference (ISMIR 2010)(Utrecht)*, Vol. 3.

[29] Spence Green, Jeffrey Heer, and Christopher D Manning. 2015. Natural language translation at the intersection of AI and HCI. *Commun. ACM* 58, 9 (2015), 46–53.

[30] Braden Hancock, Antoine Bordes, Pierre-Emmanuel Mazare, and Jason Weston. 2019. Learning from dialogue after deployment: Feed yourself, chatbot! *arXiv preprint arXiv:1901.05415* (2019).

[31] Jeffrey T Hancock, Mor Naaman, and Karen Levy. 2020. AI-mediated communication: Definition, research agenda, and ethical considerations. *Journal of Computer-Mediated Communication* 25, 1 (2020), 89–100.

[32] Geert Hofstede. 2009. Geert Hofstede cultural dimensions. (2009).

[33] Gert Jan Hofstede, Catholijn M Jonker, and Tim Verwaart. 2008. Long-term orientation in trade. In *Complexity and Artificial Markets*. Springer, 107–119.

[34] Marie-Claire Jenkins, Richard Churchill, Stephen Cox, and Dan Smith. 2007. Analysis of user interaction with service oriented chatbot systems. In *International conference on human-computer interaction*. Springer, 76–83.

[35] Ashish Kapoor, Bongshin Lee, Desney Tan, and Eric Horvitz. 2012. Performance and preferences: Interactive refinement of machine learning procedures. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*.

[36] Beomsu Kim, Seokjun Seo, Seungju Han, Enkhbayar Erdenee, and Buru Chang. 2021. Distilling the knowledge of large-scale generative models into retrieval models for efficient open-domain conversation. *arXiv preprint arXiv:2108.12582* (2021).

[37] Soomin Kim, Jinsu Eun, Changhoon Oh, Bongwon Suh, and Joonhwan Lee. 2020. Bot in the bunch: Facilitating group chat discussion by improving efficiency and participation with a chatbot. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[38] Pallavi Koppol, Henny Admoni, and Reid G Simmons. 2021. Interaction Considerations in Learning from Humans.. In *IJCAI*. 283–291.

[39] Carlene Lebeuf, Margaret-Anne Storey, and Alexey Zagalsky. 2017. How software developers mitigate collaboration friction with chatbots. *arXiv preprint arXiv:1702.07011* (2017).

[40] Minha Lee, Sander Ackermans, Nena Van As, Hanwen Chang, Enzo Lucas, and Wijnand IJsselsteijn. 2019. Caring for Vincent: a chatbot for self-compassion. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.

[41] Min Kyung Lee, Sara Kiesler, Jodi Forlizzi, Siddhartha Srinivasa, and Paul Rybski. 2010. Gracefully mitigating breakdowns in robotic services. In *Human-Robot Interaction (HRI), 2010 5th ACM/IEEE International Conference on*. IEEE, 203–210.

[42] Adam Eric Leeper, Kaijen Hsiao, Matei Ciocarlie, Leila Takayama, and David Gossow. 2012. Strategies for human-in-the-loop robotic grasping. In *Proceedings of the seventh annual ACM/IEEE international conference on Human-Robot Interaction*. 1–8.

[43] Q Vera Liao, Muhammed Mas-ud Hussain, Praveen Chandar, Matthew Davis, Yasaman Khazaeni, Marco Patricio Crasso, Dakuo Wang, Michael Muller, N Sadat Shami, and Werner Geyer. 2018. All work and no play?. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. 1–13.

[44] Bing Liu and Sahisnu Mazumder. 2021. Lifelong and continual learning dialogue systems: learning during conversation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 15058–15063.

[45] Mansoureh Maadi, Hadi Akbarzadeh Khorshidi, and Uwe Aickelin. 2021. A review on human–AI interaction in machine learning and insights for medical applications. *International journal of environmental research and public health* 18, 4 (2021), 2121.

[46] Winter Mason and Siddharth Suri. 2012. Conducting behavioral research on Amazon's Mechanical Turk. *Behavior research methods* 44, 1 (2012), 1–23.

[47] Alexander Tobias Neumann, Peter de Lange, and Ralf Klamma. 2019. Collaborative creation and training of social bots in learning communities. In *2019 IEEE 5th International Conference on Collaboration and Internet Computing (CIC)*. IEEE, 11–19.

[48] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems* 35 (2022), 27730–27744.

[49] Nataliya Pankratova and Nadezhda Nedashkovskaya. 2015. Methods of evaluation and improvement of consistency of expert pairwise comparison judgements. *International Journal «Information Theories and Applications* 22, 3 (2015), 203–223.

[50] Brandon J Perry, Yanlin Guo, Rebecca Atadero, and John W van de Lindt. 2020. Streamlined bridge inspection system utilizing unmanned aerial vehicles (UAVs) and machine learning. *Measurement* 164 (2020), 108048.

[51] Marco Roccetti, Giovanni Delnevo, Luca Casini, and Paola Salomoni. 2020. A cautionary tale for machine learning design: why we still need human-assisted big data analysis. *Mobile Networks and Applications* 25, 3 (2020), 1075–1083.

[52] Dorsa Sadigh, Anca D Dragan, Shankar Sastry, and Sanjit A Seshia. 2017. *Active preference-based learning of reward functions*.

[53] Saiph Savage, Andres Monroy-Hernandez, and Tobias Höllerer. 2016. Botivist: Calling volunteers to action using online bots. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing*. 813–822.

[54] Joseph Seering, Juan Pablo Flores, Saiph Savage, and Jessica Hammer. 2018. The social roles of bots: evaluating impact of bots on discussions in online communities. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–29.

[55] Joseph Seering, Michal Luria, Geoff Kaufman, and Jessica Hammer. 2019. Beyond dyadic interactions: Considering chatbots as community members. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. 1–13.

[56] Joseph Seering, Michal Luria, Connie Ye, Geoff Kaufman, and Jessica Hammer. 2020. It takes a village: integrating an adaptive chatbot into an online gaming community. In *Proceedings of the 2020 chi conference on human factors in computing systems*. 1–13.

[57] Marcos Serrano, Anne Roudaut, and Pourang Irani. 2017. Visual Composition of Graphical Elements on Non-Rectangular Displays. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) *(CHI '17)*. Association for Computing Machinery, New York, NY, USA, 4405–4416. https://doi.org/10.1145/3025453.3025677

[58] Burr Settles. 2009. Active learning literature survey. (2009).

[59] Muhammad Yusril Helmi Setyawan, Rolly Maulana Awangga, and Safif Rafi Efendi. 2018. Comparison of multinomial naive bayes algorithm and logistic regression for intent classification in chatbot. In *2018 International Conference on Applied Engineering (ICAE)*. IEEE, 1–5.

[60] Kurt Shuster, Jing Xu, Mojtaba Komeili, Da Ju, Eric Michael Smith, Stephen Roller, Megan Ung, Moya Chen, Kushal Arora, Joshua Lane, et al. 2022. Blenderbot 3: a deployed conversational agent that continually learns to responsibly engage. *arXiv preprint arXiv:2208.03188* (2022).
[61] Emanuel Stoeckli, Christian Dremel, Falk Uebernickel, and Walter Brenner. 2020. How affordances of chatbots cross the chasm between social and traditional enterprise systems. *Electronic Markets* 30, 2 (2020), 369–403.
[62] Emanuel Stoeckli, Falk Uebernickel, and Walter Brenner. 2018. Exploring affordances of slack integrations and their actualization within enterprises-towards an understanding of how chatbots create value. (2018).
[63] Sylvie Studente and Stephen Ellis. 2020. Enhancing the Online Student Experience Through Creating Learning Communities—The Benefits of Chatbots in Higher Education. In *Tertiary Online Teaching and Learning*. Springer, 25–33.
[64] Ed Summers and Ricardo Punzalan. 2017. Bots, seeds and people: Web archives as infrastructure. In *Proceedings of the 2017 ACM conference on computer supported cooperative work and social computing*. 821–834.
[65] Andrea L Thomaz and Cynthia Breazeal. 2008. Teachable robots: Understanding human teaching behavior to build more effective robot learners. *Artificial Intelligence* 172, 6-7 (2008), 716–737.
[66] Kashyap Todi, Daryl Weir, and Antti Oulasvirta. 2016. Sketchplore: Sketch and explore with a layout optimiser. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems*. 543–555.
[67] Heather Turner and David Firth. 2012. Bradley-Terry models in R: the BradleyTerry2 package. *Journal of Statistical Software* 48 (2012), 1–21.
[68] R Vedapradha, Ravi Hariharan, Rajan Shivakami, et al. 2019. Artificial intelligence: A technological prototype in recruitment. *Journal of Service Science and Management* 12, 03 (2019), 382.
[69] Jing Wang, Anindya Ghose, and Panos Ipeirotis. 2012. Bonus, disclosure, and choice: what motivates the creation of high-quality paid reviews?. In *ICIS*. Citeseer.
[70] Xupeng Wen, Yifeng Niu, Chang Wang, Lizhen Wu, Yuting Zhu, and Dong Yin. 2018. Human-in-the-loop multi-task tracking improved by interactive learning. In *2018 Chinese Automation Congress (CAC)*. IEEE, 2289–2294.
[71] Mairieli Wessel, Bruno Mendes De Souza, Igor Steinmacher, Igor S Wiese, Ivanilton Polato, Ana Paula Chaves, and Marco A Gerosa. 2018. The power of bots: Characterizing and understanding bots in oss projects. *Proceedings of the ACM on Human-Computer Interaction* 2, CSCW (2018), 1–19.
[72] Jason E Weston. 2016. Dialog-based language learning. *Advances in Neural Information Processing Systems* 29 (2016).
[73] Yiwei Yang, Eser Kandogan, Yunyao Li, Prithviraj Sen, and Walter S Lasecki. 2019. A study on interaction in human-in-the-loop machine learning for text analytics. In *IUI Workshops*.
[74] Beste F Yuksel, Soo Jung Kim, Seung Jung Jin, Joshua Junhee Lee, Pooyan Fazli, Umang Mathur, Vaishali Bisht, Ilmi Yoon, Yue-Ting Siu, and Joshua A Miele. 2020. Increasing video accessibility for visually impaired users with human-in-the-loop machine learning. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–9.

A SCENARIOS

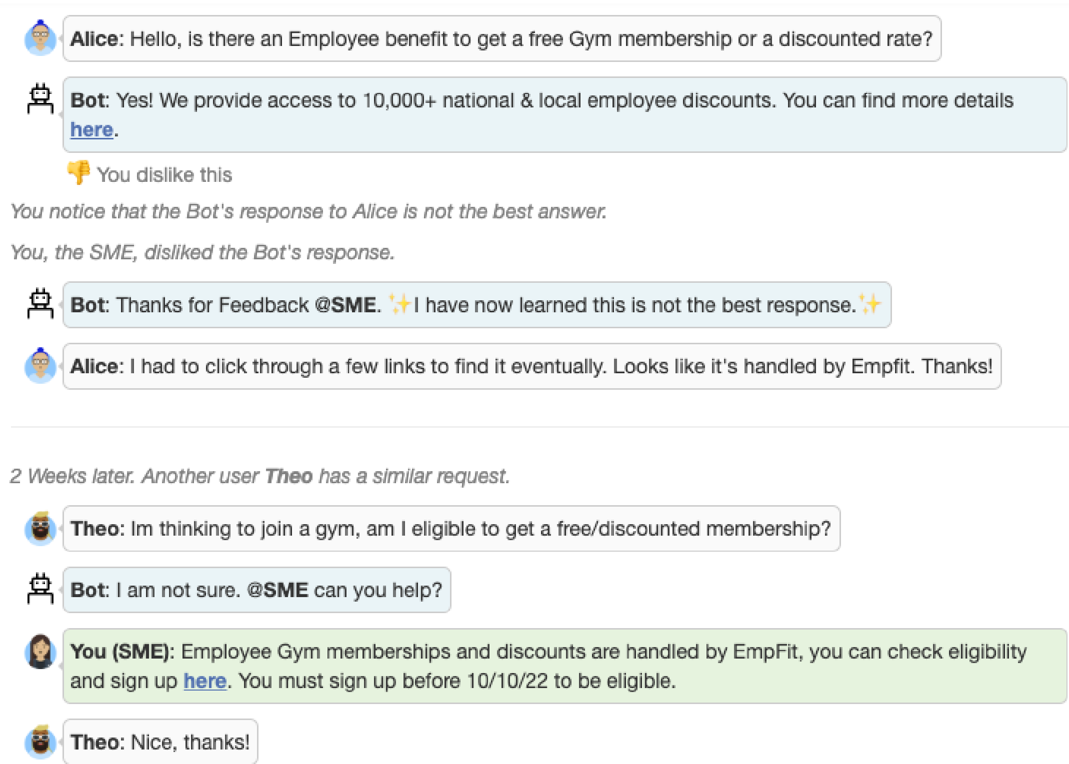


Figure 7: Benefits *Categorizing* scenario.

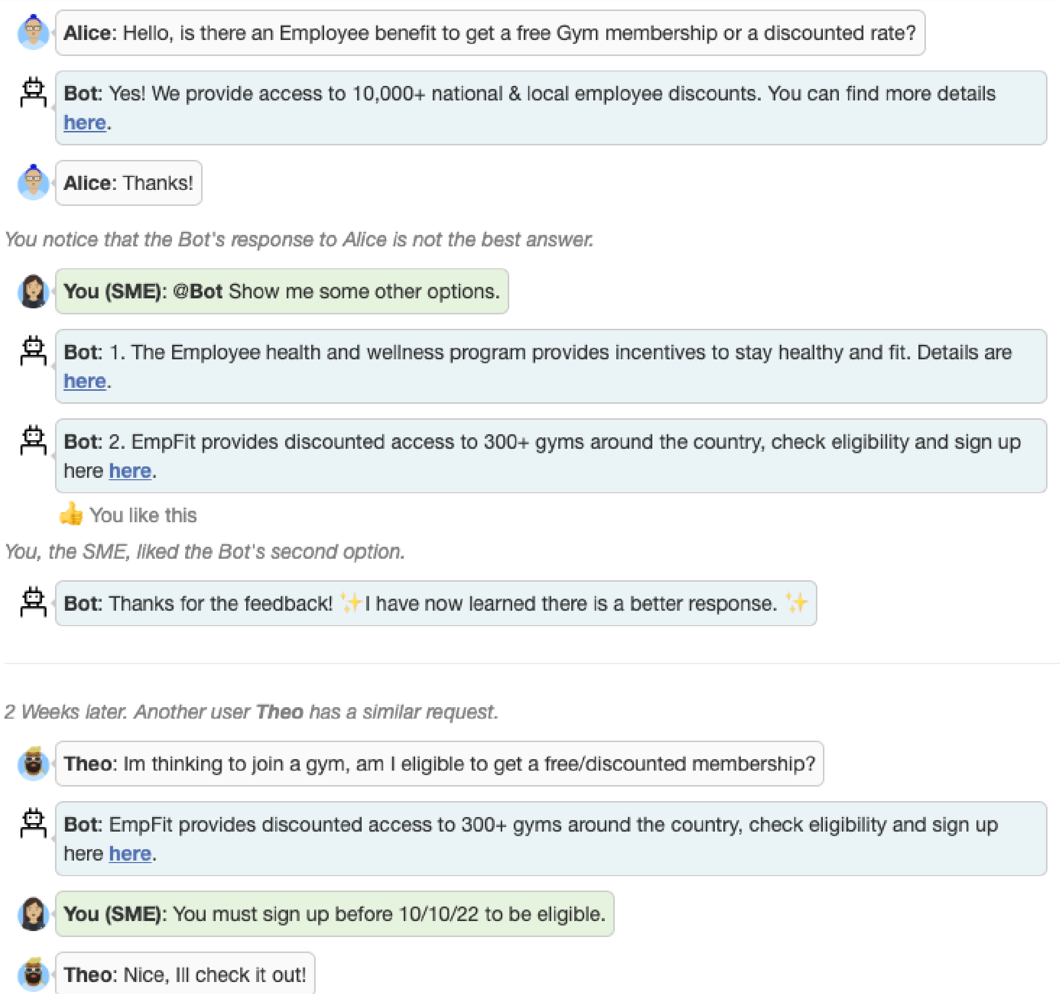


Figure 8: Benefits *Sorting* scenario.

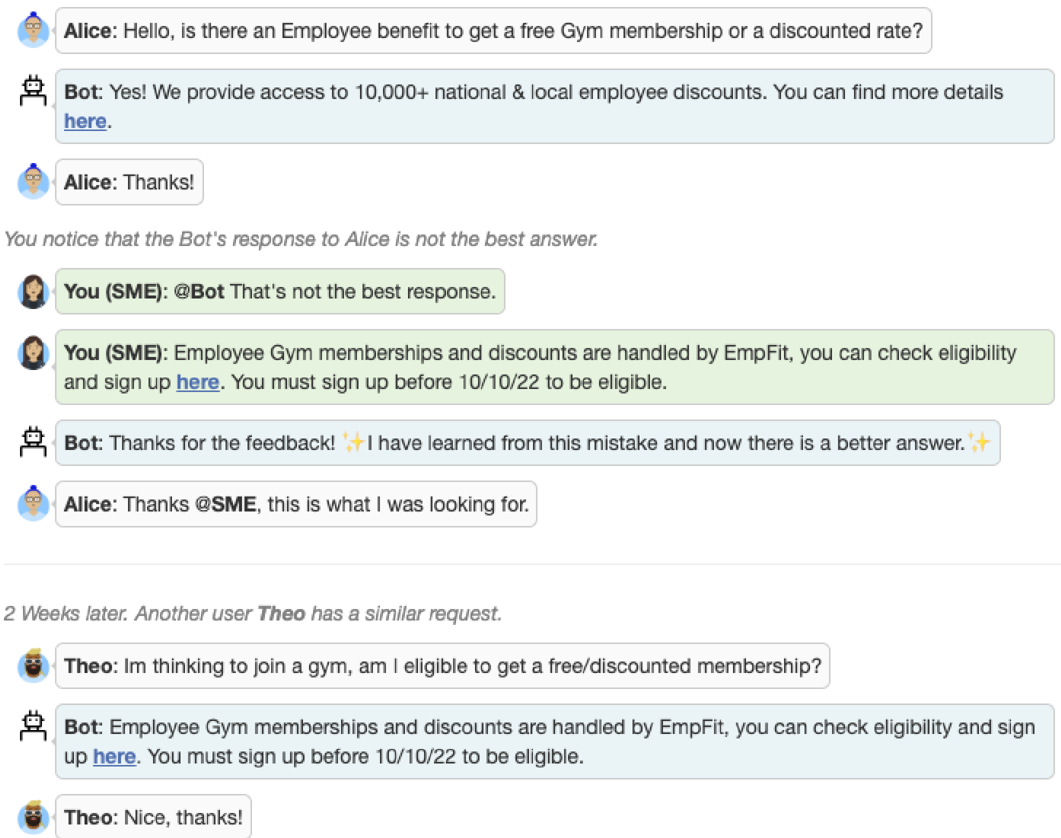


Figure 9: Benefits *Showing* scenario.

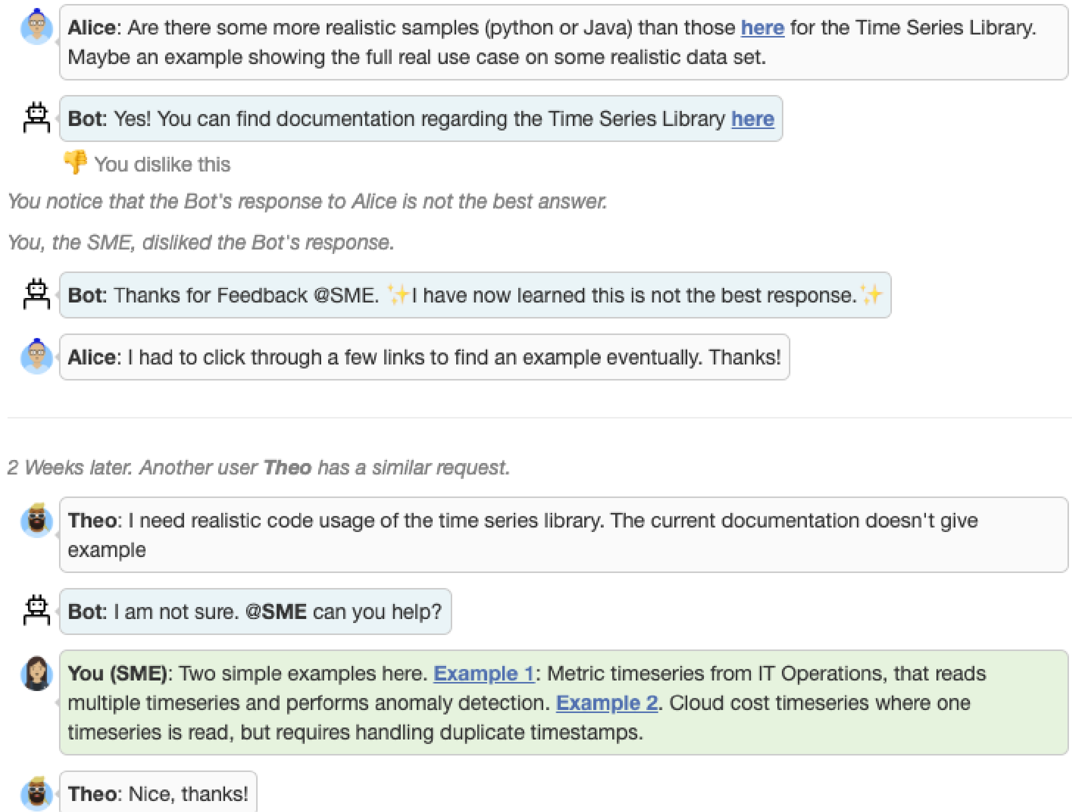


Figure 10: Open Source *Categorizing* scenario.

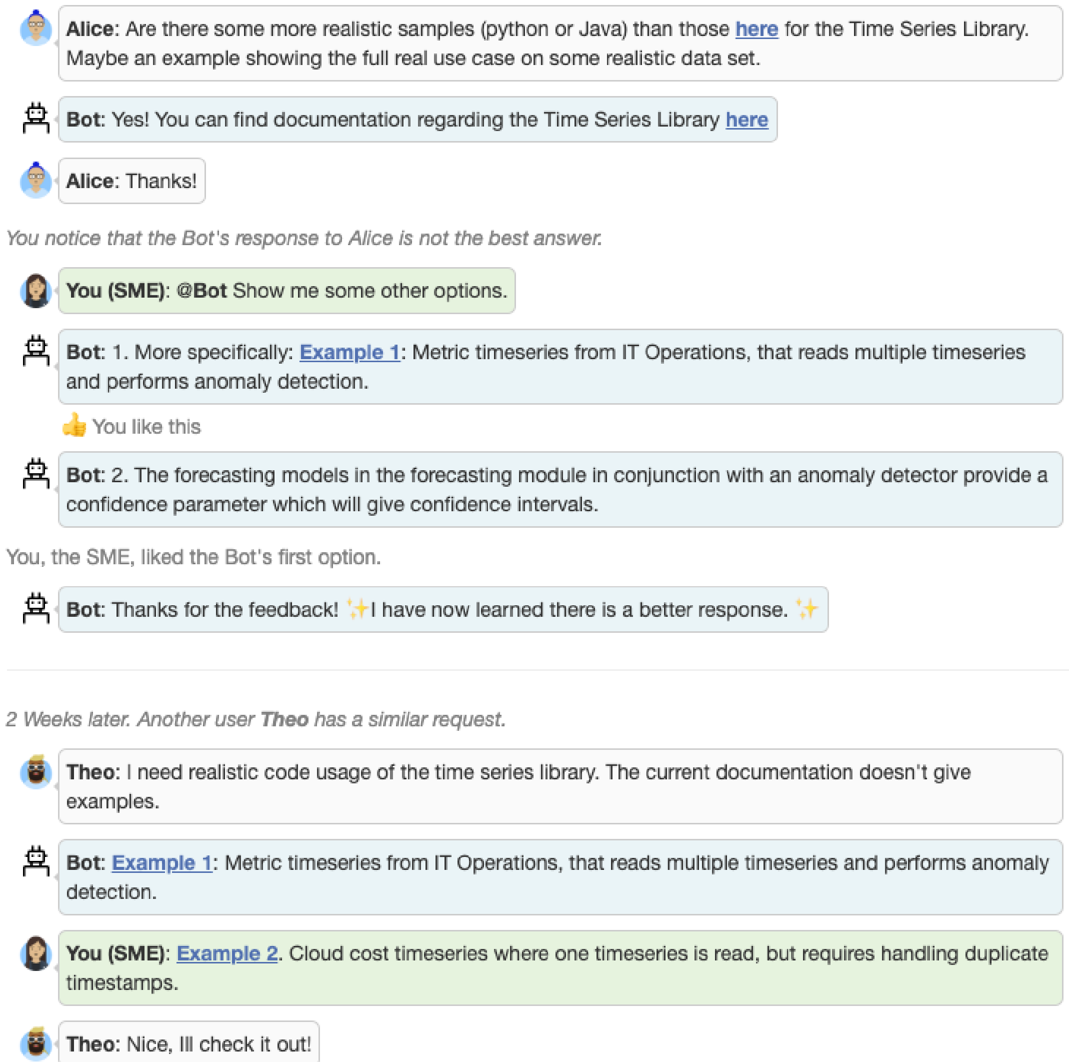


Figure 11: Open Source *Sorting* scenario.

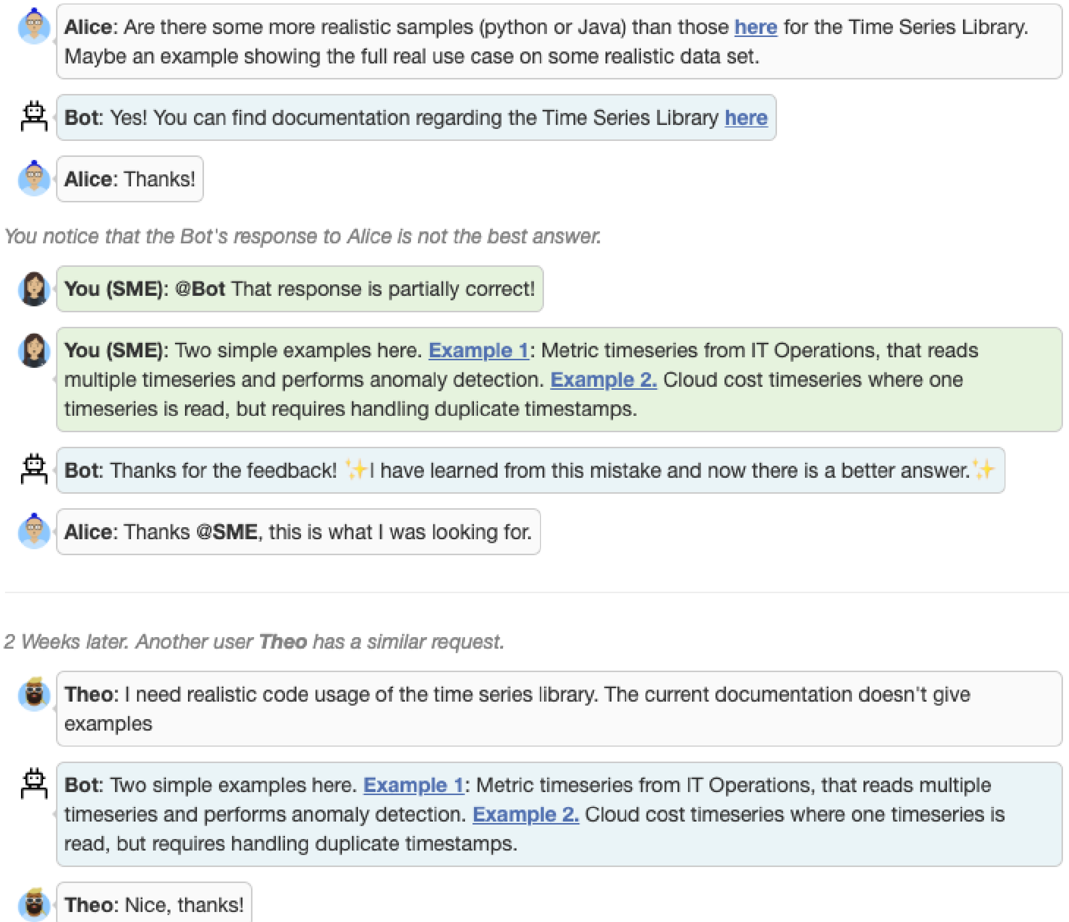


Figure 12: Open Source showing scenario