Task-Oriented User Evaluation on Critiquing-Based Recommendation Chatbots

Wanling Cai, Yucheng Jin, and Li Chen

***Abstract*—Dialogue-based conversational recommender systems (DCRSs) have become a new trend in recommender systems (RSs), allowing users to communicate with the system in natural language to facilitate feedback provision and product exploration. However, little work has been done to empirically study user perception of and interaction with such systems and, more importantly, how to best support users in providing feedback on the recommendation they receive. In this article, we aim to develop effective *critiquing* mechanisms for DCRS to improve its feedback elicitation process (i.e., allowing users to *critique* the current recommendation during the dialogue). Specifically, we have implemented three prototype systems featuring three different critiquing techniques, respectively, i.e., *user-initiated critiquing, progressive system-suggested critiquing*, and *cascading system-suggested critiquing*. We have then conducted two task-oriented user studies involving 292 subjects to evaluate the three prototypes. In particular, we consider two typical types of user tasks in RSs: basic recommendation task (BRT, i.e., looking for items according to the user's preferences), and exploration-oriented task (EOT, i.e., exploring different types of items). Results show that EOT stimulates more user interaction, while BRT results in higher user satisfaction. Moreover, when users perform EOT, the type of critiquing techniques is more likely to influence user perception and moderate the relationships between certain interaction metrics and users' perceived serendipity. The findings suggest effective critiquing techniques to enhance the interaction between users and the recommendation chatbot when the system makes recommendations for different purposes.**

***Index Terms*—Chatbot, conversational recommender systems, critiquing, feedback elicitation, user evaluation.**

I. Introduction

In the era of information explosion, recommender systems (RSs) are undoubtedly successful applications of artificial intelligence, providing personalized recommendations (e.g., movies, songs, hotels) for people to make informed decisions. To develop a RS that can better facilitate users' decision-making process, researchers have continued putting efforts in improving the interaction between users and the system, such as building interactive recommenders [1], [2], and making recommendations in conversation [3].

In recent years, there are increasing cases where recommendations are presented to users through dialogues [3]. Such systems, often named dialogue-based conversational recommender systems (DCRSs) [4], [5], enable natural language communication between users and the system, showing great potential for stimulating users' feedback provision and product exploration. However, so far little work has empirically investigated how users perceive and interact with such systems, and, more importantly, how to best support their provision of feedback on the recommendation especially when the recommended item does not satisfy their requirements [5]. Our previous work has studied a critiquing-based recommendation chatbot featuring two critiquing techniques (i.e., *user-initiated critiquing (UC)* and *system-suggested critiquing (SC)* [6]), enabling users to critique the recommendation during the conversation [7]. The results of this prior user study reveals that users tend to find a higher diversity of recommendations when using the system with both UC and SC. Motivated by this observation, we aim to further strengthen critiquing techniques in conversational interaction for DCRS, so as to facilitate its feedback elicitation process, thereby improving user interaction with the system.

Therefore, in this work, we have designed two kinds of system-suggested critiquing (SC) technique: ***Progressive system-suggested critiquing (Progressive SC)*** and ***Cascading system-suggested critiquing (Cascading SC)*** for eliciting users' feedback and facilitating users' exploration of recommendations in two different ways. The former is preference-oriented, which provides critiques (e.g., "*Since you liked the song XXX, would you like to try the song of lower energy of the same genre?*") based on users' current preferences and incremental critiquing feedback [8], while the latter is diversity-oriented, which suggests critiques (e.g., "*Would you like to try another music genre such as country music?*") to steer users into a cascade of diverse types of items by using a strategical approach based on the assumption of the cascading user behavior as inspired by [9]. Then, we have developed a music chatbot with three system variants, which feature *UC* (i.e., users can make critiques on the recommendation by themselves), *Progressive SC*, and *Cascading SC*, respectively.

We have then conducted two task-oriented user studies (involving 292 subjects), which focus on two typical types of user tasks in a RS [10], respectively: **basic recommendation**

Manuscript received March 23, 2021; revised October 2, 2021 and November 14, 2021; accepted November 18, 2021. Date of publication January 7, 2022; date of current version May 17, 2022. This work was supported in part by the Hong Kong Baptist University IRCMS Project under Grant IRCMS/19-20/D05 and in part by the Hong Kong Research Grants Council under Grant RGC/HKBU12201620. This article was recommended by Associate Editor E. Barakova. *(Corresponding author: Wanling Cai.)*

The authors are with the Department of Computer Science, Hong Kong Baptist University, Hong Kong SAR, China (e-mail: cswlcai@comp.hkbu.edu.hk; yuchengjin@hkbu.edu.hk; lichen@comp.hkbu.edu.hk).

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Research Ethics Committee of the Hong Kong Baptist University.

Color versions of one or more figures in this article are available at https://doi.org/10.1109/THMS.2021.3131674.

Digital Object Identifier 10.1109/THMS.2021.3131674

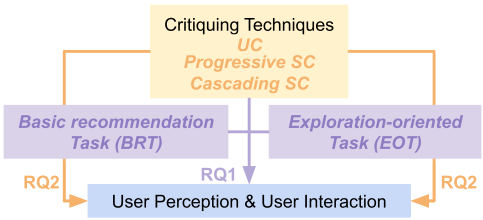


Fig. 1. Our research questions.

task (BRT), i.e., looking for songs according to the user's current preferences; and **exploration-oriented task (EOT)**, i.e., exploring diverse types of songs. The latter was motivated by recent attempts of encouraging users to explore diverse recommendations for mitigating the "filter bubble" issue in RS [11], [12]. The "filter bubble" issue refers to a phenomenon where users become isolated from the items that do not suit their tastes. Personalized recommendations to users are too aligned with their current preferences, which may lead to increasingly narrower exploration space over time [13]. Given that different types of user tasks may influence user interaction with the system (e.g., users tend to spend longer time in difficult tasks than easy tasks [14]) and, hence, impact their experiences [15], [16], we are particularly interested in investigating how the *task type* may take effect on users' perception of and interaction with critiquing-based recommendation chatbots.

Specifically, we aim to address the following two research questions (see Fig. 1).

RQ1: *How do task types influence users' perception of and interaction with the three different critiquing techniques?*

RQ2: *How do critiquing techniques influence user perception and interaction behavior in the basic recommendation task and the exploration-oriented task, respectively?*

The main contributions of our work are four-fold.

1) We have proposed two kinds of SC technique, i.e., *Progressive SC* and *Cascading SC*, to encourage users' feedback provision and recommendation exploration.
2) We have conducted two task-oriented user studies to investigate the influence of task type on users' perception of and interaction with three different critiquing methods. The experimental results show that the *task type* significantly impacts user experience with the critiquing system. EOT encourages more user interaction, while BRT results in more positive user perception like higher satisfaction.
3) We have then investigated the effects of critiquing techniques in BRT and EOT, respectively, and find that, when users perform BRT, three critiquing techniques are perceived at the same level; however, when performing EOT, users perceive higher diversity of recommendations by the system that offers *Cascading SC*, and feel more serendipitous recommendations by the system that offers *Progressive SC*. Also, in EOT, critiquing techniques significantly moderate certain relationships between interaction metrics (such as the number of listened songs) and users' perceived serendipity and satisfaction.
4) We have finally discussed our findings and provided practical implications for designing critiquing-based recommender chatbots for serving users' different purposes.

II. Related Work

A. Conversational Recommender Systems

Conversational Recommender Systems (CRSs) aim to help users seek their desired items through natural language [3], [17], [18]. For instance, ExpertClerk [17] is a conversational agent designed to interact with shoppers by asking questions to obtain their preferences and proposing recommendations to assist users in finding their satisfactory products. Another system, the adaptive place advisor [18], provides personalized recommendations to help users find preferable places for traveling by considering both users' long-term preferences and short-term interests. Several studies have shown the superiority of conversational user interfaces over graphical user interfaces during the process of recommendations [19]–[21].

In the broader area of RSs, critiquing-based RSs have been proposed to elicit users' critiquing feedback to help the system improve the recommendation [6]. In particular, there are two major types of critiquing technique, including user-initiated critiquing (i.e., users construct critiques by themselves) and system-suggested critiquing (i.e., the system generates a set of critique candidates for users to choose). A recent work [7] studied such systems with conversational interaction and found that, while both critiquing techniques enable users to control recommendations in conversational user interfaces, the incorporation of SC leads to users' better perception of recommendation diversity during the interaction with the system. Inspired by this observation, we are interested in investigating in-depth how critiquing techniques could be further improved to facilitate users' feedback provision and enhance user interaction with the recommendation chatbot when users perform different tasks.

Different from [7], in this work, we introduce two kinds of system-suggested critiquing (SC): *Progressive SC* that is preference-oriented (generating critiques considering both users' current preferences and incremental critiquing feedback [8]); and *Cascading SC* that is diversity-oriented (suggesting critiques in a strategic approach with the assumption of the cascading user behavior as inspired by [9]). In addition, we consider the chatbot's proactivity in our designed system (i.e., the ability of proactively offering SC to encourage users to find music), since the robot's proactivity may help people get rich information and reduce the decision space [22].

B. Effects of Task Type

During the interaction with RS, users may have various tasks (e.g., finding relevant items, exploring the decision space) [10], [23]. Previous studies concerning the evaluation of RS have suggested that users' choice goal may exert the influence on their perception of and interaction with the system [24], but most studies evaluate their proposed systems with a basic recommendation task (BRT), e.g., "help the user find relevant/good items" [10]. To

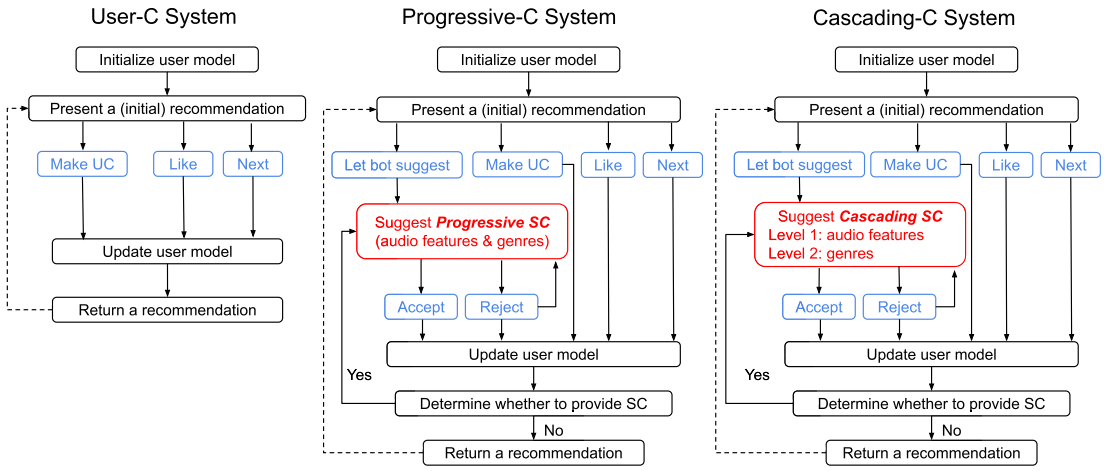


Fig. 2. Three system variants' behavior policies during the conversation.

the best of our knowledge, few studies consider converting different users' choice goals into different types of user tasks when evaluating RS.

On the other hand, some studies in the information retrieval (IR) domain have demonstrated that task characteristics (such as task type, task complexity, and task difficulty) are important factors that can influence user experience with the IR system [16], [25]. Their results show that users tend to make more efforts in difficult tasks than in easy tasks, and that different types of tasks lead to different searching behavior [15].

Inspired by these works, we are interested in investigating the effects of *task type* on users' perception of and interaction with critiquing-based recommendation chatbots. As researchers have recently paid more attention to supporting user exploration to minimize the "filter bubble" issue of personalized recommendations [11], [13], we particularly consider the exploration-oriented task (EOT), i.e., exploring diverse types of songs, together with the basic recommendation task (BRT), i.e., looking for items according to the user's current preferences, as two typical types of user tasks in our studies.

III. System Design

Following the workflow of an existing music chatbot [7], we have developed a music chatbot by using a popular NLU platform, DialogFlow,[1] and a widely used music service, Spotify API.[2] The system supports both user-initiated critiquing (UC) and system-suggested critiquing (SC) [7]. In particular, we have devised two kinds of SC in the newest version [26]: *Progressive SC* that guides users to different recommendations based on their current preferences and incremental critiquing feedback; and *Cascading SC* that motivates users to explore a cascade of different types of music. Specifically, we implemented three variants of the critiquing system as follows.

User-initiated Critiquing System (User-C): The system only supports UC. Users can post self-initiated critiques to the current recommendation based on music-related attributes such as genres, tempo, and danceability.

Progressive Critiquing System (Progressive-C): The system is a hybrid critiquing system that supports both UC and SC. Users can either post UC or ask the system to provide *Progressive SC* to help them access more recommendations.

Cascading Critiquing System (Cascading-C): Similar to the Progressive-C system, the system also supports both UC and SC, but provides *Cascading SC* when the SC is triggered.

A. Behavior Policies and Algorithms

Based on the typical recommendation process introduced in [6], we design the associated behavior policies for these three types of system, as shown in Fig. 2.

Initiation: Before initiating the conversation, the system obtains users' initial preferences for three attributes, i.e., songs, artists, and music genres, so as to initialize the user model. Of note, the music data (including metadata and song attributes) in our system were obtained from the Spotify platform. Our system gets users' preference data from their profiles in Spotify, or creates preference data for non-regular Spotify users by asking them their favorite songs and artists. Then, the system calls Spotify recommendation API to obtain 150 recommendations for generating a ranked playlist based on the initial user model. As we can only collect a single user's preference data after s/he login into the system in our case, we adopted the multi-attribute utility theory (MAUT) [27] to estimate her/his preferences over songs. Formally speaking, MAUT estimates the user (denoted as u)'s preference over each song (denoted as i) as $r^M_{u,i} = \sum_{a \in \mathcal{A}} w_{u,a} \times v(u,i,a)$, where $\mathcal{A}$ denotes all concerned music-related attributes, and $w_{u,a}$ is the relative importance (i.e., the user u's preference weight) of the attribute a. $v(u,i,a)$ represents the user u's preference over the song i regarding the attribute a, which is measured as $p(k_{a,i}|\mathcal{I}^{\text{liked}}_u)$, i.e., the probability that the attribute a's values appearing in the user u's previously favorite songs ($\mathcal{I}^{\text{liked}}_u$) fall into the value bin[3] of the attribute a of the currently considered song i (denoted as $k_{a,i}$). The initial weights of all attributes are the same and will be gradually adjusted based on the user's subsequent critiques on the attributes.

[1][Online]. Available: https://dialogflow.com

[2][Online]. Available: https://developer.spotify.com/documentation/web-api

[3]We divide the value range of each attribute into 10 or 15 bins for numerical attributes. For categorical attributes, each value refers to one value bin.

User-initiated critiquing (UC): After receiving a recommendation, the user may make self-initiated critique on its audio attributes (i.e., energy, danceability, speechiness, tempo, and valence), music categories, or artists, e.g., saying *"I want higher tempo."* The system then updates the user model and returns a new recommendation.

System-suggested critiquing (SC): In the two hybrid critiquing systems, the user can ask for the system's suggested critiques (i.e., *Progressive SC* or *Cascading SC*) by clicking the button *"Let bot suggest"*. In response, the system provides the suggested critique, e.g., *"Compared to the last played song, do you like the song of lower tempo?"* User feedback to the suggested critique ("Accept" by clicking the button "*Yes*" or "Reject" via the button "*No*") will then be used to update the user model and make subsequent recommendations.

There are two major differences between *Progressive SC* and *Cascading SC*. First, the critique selection of *Progressive SC* mainly considers the user's preferences over songs and critiquing feedback as captured from the previous interactions, while *Cascading SC* focuses more on the diversity of recommended songs. Second, *Cascading SC* contains two levels of critiquing: At Level 1, the suggested critiques are on audio features, which keep the user within the currently listened music genre; at Level 2, critiques are on music genres, which encourage the user to try songs in a different genre. *Progressive SC*, however, does not make a distinction between audio attributes and genres.

Specifically, the generation of these two kinds of system-suggested critique consists of the following four major steps.

1) The system first constructs a critique pattern vector for each candidate song in the current playlist (e.g., *{(genre, pop), (valence, higher),..., (danceability, lower)}*) by comparing it with the currently recommended song in terms of music-related attributes. Each critique pattern (e.g., *(genre, pop)*) denotes a critique that contains one attribute, which is also called unit critique [6].
2) The system filters out the critiques rejected by the user in her/his previous interactions, as well as the critiques rarely occurring in all critique pattern vectors (frequency lower than 10%). Then, for each remaining critique, the songs in the current playlist that satisfy this critique are grouped together as its contained songs.
3) The system selects *Progressive SC* by calculating the utility of each remaining critique (denoted as c) [28] as $U_u(c) = w_{u,a_c} \times f_c \times \frac{1}{\mathcal{I}_c} \sum_{i \in \mathcal{I}_c} (r^M_{u,i} + r^C_{u,i})$, where w_{u,a_c} denotes the user u' preference for c's contained attribute, f_c denotes the relative frequency of c among all critique pattern vectors, and $\mathcal{I}_c$ denotes the set of songs that satisfy c. $\frac{1}{\mathcal{I}_c} \sum_{i \in \mathcal{I}_c} (r^M_{u,i} + r^C_{u,i})$ represents u's preference over c's contained songs, which considers u's preference over the song i (estimated as $r^M_{u,i}$ based on MAUT), as well as the compatibility of i with the critiques previously made by u (PC_u) [8] (calculated as $r^C_{u,i} = \frac{1}{|\text{PC}_u|} \sum_{c' \in \text{PC}_u} \text{satisfies}(c,' i)$, where $\text{satisfies}(c,' i)$ is an indicator function used to check whether the song i satisfies c' or not).
   For *Cascading SC*, the system calculates the overall diversity of the critique's contained songs and the songs the user has listened to in the previous interactions. As Shannon's entropy [29], [30], a popularly used diversity metric, can measure both the difference between the recommendation candidates and users' listened songs and the degree of recommendation novelty (i.e., items with high entropy are likely to be novel to the user), in our design, we calculate the diversity by the average Shannon's entropy across all music-related attributes [31]: $D_u(c) = \sum_{a \in \mathcal{A}} H_a(c)$, where $H_a(c) = -\sum_{k \in K_a} p(k|\mathcal{I}_c \cup \mathcal{I}_L) \log p(k|\mathcal{I}_c \cup \mathcal{I}_L)$ measures the entropy[4] of the attribute a, $k \in K_a$ denotes one value bin k in all value bins K_a of the attribute a, $\mathcal{I}_L$ denotes the listened songs by the user, $\mathcal{I}_c \cup \mathcal{I}_L$ represents the resulting set of songs when the user accepts c, and $p(k|\mathcal{I}_c \cup \mathcal{I}_L)$ refers to the probability that the attribute a's values of the resulting set of songs fall into the value bin k. Motivated by observations of our pilot study,[5] we determine *Cascading SC* to be switched from Level 1 to Level 2 when the user likes more than four songs or skips more than three songs within the currently listened genre.
4) The system finally shows the critique of the highest utility $U(c)$ in Progressive-C, or diversity $D(c)$ in Cascading-C.

Inspired by recent studies about the chatbot's proactivity [22], each hybrid critiquing system (i.e., Progressive-C or Cascading-C) is designed to provide SC in two different manners: Reactive SC that suggests critiques to users when they make an explicit request (i.e., clicking the button "*Let bot suggest*" during the conversation; see the interface of our music chatbot in Fig. 3); and Proactive SC that proactively offers critiques for stimulating users to find music that they may like, which will further be described in the dialogue management.

User modeling: User model contains two parts: 1) user preference model stores the user's preferred value range and preference weight for the critiqued attribute, i.e., a music genre or an audio feature, which will be adjusted based on the user's feedback on the recommended item (i.e., clicking *"Like"* for accepting or *"Next"* for skipping) and the critique made by the user; 2) user critiquing history tracks all occurred critiques in the current dialogue.

Dialogue management: All the three systems are designed to respond to the user's inputs after detecting her/his intents (i.e., the user's feedback intents such as skipping the song or making critiques), but they may respond differently to the detected intents. For the User-C system, the system proceeds to the next recommendation based on the user's intent, while the two hybrid systems will determine whether it is time to recommend a song or show a system-suggested critique based on the user's interaction behavior. We find it is reasonable to let the system proactively offer critique if the user has consecutively skipped three recommended songs or listened to five songs according to our observation in the pilot study.

Recommendation: With the refined user model, we rerank the current playlist by calculating the sum of the MAUT-based

[4] A higher entropy of an attribute indicates that the resulting set contains songs with higher diversity in terms of that attribute.

[5] We conducted a lab-controlled pilot study (with 3 volunteers) in order to test adequacy of our system and the experimental procedure.

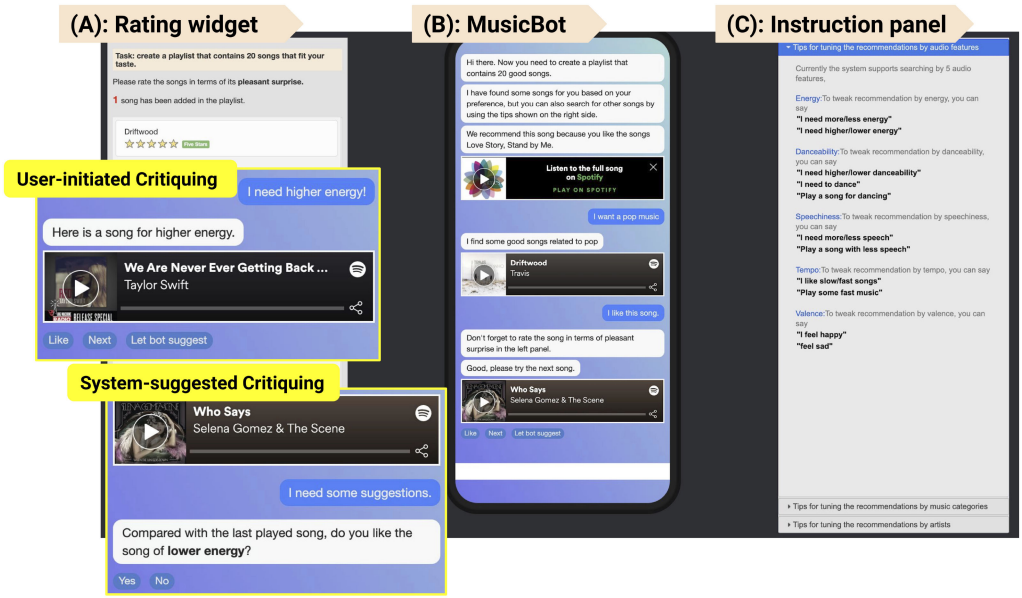


Fig. 3. User interface of our music chatbot. Note that the user interface is the same as that of [7], but the underlying algorithms used to generate the two kinds of system-suggested critique (i.e., *Progressive SC* and *Cascading SC*) are different.

estimated user preference and the compatibility with the user's critiquing feedback for each candidate song.

B. User Interface Design

The user interface of the music chatbot consists of three parts: a rating widget, a dialogue window, and an instruction panel. Specifically, the dialogue window [see Fig. 3(B)] shows the dialogue between the user and the bot. The recommended song is shown on a card with a set of buttons under the card for the user to give feedback. When the user clicks the "*Like*" button, the current song will be added to the playlist where the user can rate the song [see Fig. 3(A)]. The "*Next*" button allows the user to skip the current song, and the "*Let bot suggest*" button is to activate a system-suggested critique on the song. If the user would like to critique the recommended song on her/him own, s/he can send a message to tune the recommendation, e.g., by audio features, or music genres [see Fig. 3(C) explains the supported features with some examples]. Two dialogue examples illustrate how the user can make UC and SC, respectively.

TABLE I
NUMBERS OF PARTICIPANTS IN THE THREE ECS FOR TWO STUDIES

	User-C	Progressive-C	Cascading-C	Total
Study 1	32	45	35	112
Study 2	35	36	36	107

TABLE II
DEMOGRAPHICS OF 219 PARTICIPANTS IN OUR STUDIES

	Item	Study 1	Study 2
Age	19-25	59	40
	26-30	21	19
	31-40	26	25
	41-50	2	13
	>50	4	10
Gender	Male	65	53
	Female	46	52
	Other	1	2
Nationality	United States of America	35	12
	United Kingdom	20	35
	Portugal	8	16
	Poland	7	11
	Others	42	33

IV. EVALUATION

In order to investigate the effects of *task type* and *critiquing technique* on user perception of and interaction with critiquing-based recommendation chatbots, we conducted two task-oriented user studies, which focus on two typical types of user tasks as supported by a RS [10], respectively: basic recommendation task (BRT) and exploration-oriented task (EOT), with a between-subjects design (N=112 in Study 1 for BRT, N=107 in Study 2 for EOT). In each study, we randomly assigned participants to one of the three experimental conditions (ECs): User-C, Progressive-C, and Cascading-C (see Table I).

A. Participants

Participants were recruited from the Prolific platform,[6] which is popularly used for academic surveys [32]. To ensure the quality of the experiment, we pre-screened users in Prolific using the following criteria: 1) participants should be fluent in English; 2) the number of her/his previous submissions should be more than 100; 3) the approval rate should be greater than 95%. In Study 1, the experiment took 15 mins on average and each participant was compensated £2.0 if s/he successfully completed

[6][Online]. Available: https://www.prolific.co/

TABLE III
POST-STUDY QUESTIONNAIRE FOR MEASURING USERS' PERCEPTION OF THE MUSIC CHATBOT

Metric	Statement (each is rated on a 7-point Likert scale)
Interest	Q1. The songs recommended to me matched my interests.
Novelty	Q2. The songs recommended to me were novel.
Music discovery	Q3. The music chatbot helped me discover new songs.
Diversity	Q4. The songs recommended to me were diverse.
Serendipity	Q5. The music chatbot provided me with recommendations that I had not considered in the first place but turned out to be a positive and surprising discovery.
Interaction adequacy	Q6. I found it easy to inform the music chatbot if I dislike/like the recommended song.
Ease of use	Q7. I easily found the songs I was looking for.
Transparency	Q8. I understood why the songs were recommended to me.
Control	Q9. I felt in control of modifying my taste using this music chatbot.
Trust	Q10. This music chatbot can be trusted.
Confidence	Q11. I am confident I will like the songs recommended to me.
Satisfaction	Q12. Overall, I am satisfied with this music chatbot.

the experiment. In Study 2, the experiment took 25 min on average and each participant was compensated £2.4.

A total of 292 users (145 users for Study 1 and 147 users for Study 2) participated in our studies, which is within our estimated sample size[7]. With the 1.5 times interquartile range (1.5×IQR) rule, we identified 40 outliers that have an extremely long duration in the experiment (i.e., longer than 32 min in Study 1 and 50 min in Study 2). To avoid the disproportionate effect of outliers on statistical results, we removed their responses in our analysis. We also filtered out 33 participants due to their failure to pass the attention check questions.[8] We finally kept the data of 219 participants: 112 for Study 1 and 107 for Study 2. See Table II for demographics of those participants.

B. Procedure

First, participants need to accept the general data protection regulation consent form before signing into our system with their Spotify accounts. After reading the instructions of the user study, participants are asked to fill out a pre-study questionnaire. To ensure that they understand the study task and the use of our chatbot, we ask them to read a tutorial about interacting with music recommendations in the chatbot and then try the assigned chatbot for two minutes. Once they are ready, they are asked to complete the experimental task. In Study 1, the task BRT is to interact with our chatbot to find five pieces of songs that suit the user's preference, and rate each song in terms of its pleasant surprise. In Study 2, the task EOT contains two steps: First, use our chatbot to discover songs in different music types as much as possible, and create a playlist that contains 20 pieces of music that fit the user's taste, and then rate each song in terms of its pleasant surprise. Second, select the top-5 most preferred songs from the created playlist. The two-step design of the task EOT was inspired by the previous exploration-related studies and the results of user interviews in our pilot study. In the previous related studies, to engage users in the exploration task, they required participants to rate 20 songs and spend at least ten minutes [2], [34] or explore at least five genres [35] during the interaction. Moreover, all the three participants in our lab-controlled pilot study expressed that if they were only allowed to add five songs to their playlist, they would feel having less chance to explore different types of music. To engage users in exploring diverse music, we decided to allow them to first add 20 songs to their playlist, and then select the top-5 preferred songs. In this way, we can also measure if users can find songs they feel more pleasantly surprised through performing EOT.

After finishing the task, participants fill out a post-study questionnaire regarding their experiences with the music chatbot (see the following section).

C. Measurement

The post-study questionnaire contains 12 statements (see Table III) that measure user perception of music recommendations when using the chatbot: Q1-Q4 and Q6-Q12 are adapted from ResQue (a widely used user-centric evaluation framework for RSs) [36]. The statement of Q5 measures user perceived serendipity according to [37]. All measures are self-reported using 7-point Likert scale from "*Strongly disagree*" to "*Strongly agree*".

V. ANALYSIS AND RESULTS

A. Effects of Task Type

To investigate how the task type (i.e., BRT and EOT) influences user perception of and interaction with recommendations in the three critiquing systems, we ran two-way ANOVA (2×3) to analyze its effect and that of experimental condition (EC) on a particular dependent variable.

1) User Perception: The results of two-way ANOVA show significant main effects of task type on five user perception metrics, but no interaction effect between task type and EC. As shown in Table IV, when users performed BRT, their perceived interest matching, interaction adequacy, transparency, control, and satisfaction under all the three conditions are significantly higher than those when they performed EOT, though they positively rated the three systems (above 5 on the 7-point Likert scale) for both tasks.

2) User Interaction: We further extracted major interaction data from participants' logs to examine the effect of task type and EC on users' interaction behavior. The results indicate significant main effects of task type on all the user interaction

[7]Based on the results from our online pilot study (involving 20 participants), we calculated the sample size as 111–159 for each study in a priori power analysis for an ANOVA F test (given a significance level $\alpha = .05$, a power level $(1\text{-}\beta) = .8$ and an expected effect size $f = .25$ or. 3) using G*Power [33].

[8]To ensure the quality of user responses, we set three attention checking questions (e.g.,"*Please indicate which of the following item is not fruit?").*

TABLE IV
MAIN EFFECTS OF TASK TYPE, I.E., BRT AND EOT, ON USER PERCEPTION AND USER INTERACTION

	Study 1 (BRT)		Study 2 (EOT)		Main effects (Task Type)	
	Mean	Std	Mean	Std	F	p
Perception metrics						
Q1: Interest	**6.06**	0.96	5.68	0.97	8.33	**
Q6: Interaction adequacy	**6.10**	1.03	5.76	1.27	4.47	*
Q8: Transparency	**6.04**	0.87	5.76	0.98	5.38	*
Q9: Control	**5.49**	1.22	5.14	1.38	3.99	*
Q12: Satisfaction	**5.74**	1.20	5.36	1.53	4.01	*
Interaction metrics						
#Listened songs	11.51	5.86	**41.09**	13.81	423.17	***
Duration (minutes)	4.13	1.95	**11.83**	4.78	247.63	***
#Dialogue turns (times)	14.07	7.24	**50.03**	18.03	402.52	***
#Button (times)	11.63	5.95	**42.55**	15.64	445.28	***
#Button-Next (times)	3.96	4.45	**13.55**	10.09	81.22	***
#Typing (times)	2.55	3.08	**7.69**	7.26	46.53	***
#Words per utterance	2.58	2.09	**3.23**	1.52	6.40	*

Note: Only significant results are included in this table. Significance: *** $p<.001$, ** $p<.01$, * $p<.05$

metrics (see Table IV). Compared with BRT, EOT led to significantly more listened songs by users, longer duration, more dialogue turns, which also resulted in more button clicks, more skipped songs by clicking the "*Next*" button, and more typed utterances. This can be explained by the explicit request of asking participants to add 20 songs into their playlist during exploration in EOT. We also find that users typed longer utterances to accomplish EOT than BRT. In-depth analysis shows that 57.94% of users typed equal to or more than four words on average in their utterances (such as "*I need a song for dancing*") when performing EOT, while more than a half of participants (56.25%) in BRT typed utterances with less than four words (e.g., "*less energy*" or "*lower danceability*"). In terms of utterance content, we find that users tend to tune the recommendation by music genres and artists more in EOT (55% of utterances) than in BRT (46% of utterances), suggesting that users are probably more motivated to explore different types of music in EOT.

From the abovementioned results, we can see that task type is an important factor that can impact user experience with the three critiquing systems. In particular, the exploration-oriented task (EOT) leads to more user interactions but lower perceived interaction adequacy and satisfaction than the basic recommendation task (BRT). It may be that, compared with BRT that allows users to engage in a listening session to find songs of their own interests, EOT requires a more active exploration and selection of diverse choices when listening to music recommendations in the lean-in scenario, and hence, likely induces more user interactions and a higher cognitive load from users [12].

B. Effects of Critiquing Technique

We have then investigated how the critiquing technique influences user perception and interaction in two user tasks, respectively, i.e., BRT and EOT. Specifically, for each user task, we analyzed users' responses to the twelve statements (see Table III) and their interaction behavior in the three ECs (see Table I), respectively. Since the results of the Shapiro–Wilk test show that the data are not normally distributed, we performed the nonparametric one-way ANOVA Kruskal–Wallis test for comparative analysis.

Basic Recommendation Task (BRT)

1) User perception: The results of Kruskal–Wallis tests show no statistically significant difference among the three conditions in terms of users perception metrics when users performed BRT. From the results reported in Fig. 4(a), we find that users positively rated all of the three critiquing systems in the majority of the perception metrics with average ratings above 5 on a 7-point Likert scale, including interest matching, interaction adequacy, ease of use, transparency, control, trust, confidence, and satisfaction. This may suggest that our music chatbot is useful for users to find songs that suit their preferences.

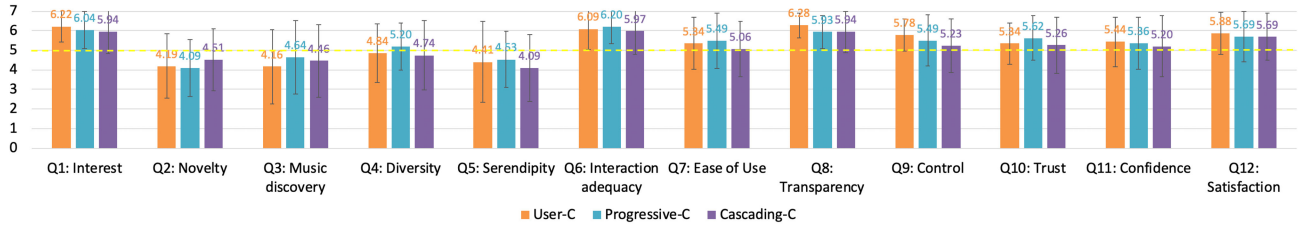
2) User interaction.

Interaction metrics: According to the Table V, there is a significant difference among the three ECs regarding times of clicking buttons ($H = 8.48, df = 2, p<.05$) in BRT. The post-hoc Mann–Whitney tests with Bonferroni corrected p-value show that users clicked significantly more buttons in Cascading-C than in User-C ($p<.05$). Another finding is that users tend to skip more songs (by clicking the "*Next*" button) in User-C than in the two hybrid systems especially Progressive-C, inferring that incorporating SC might be more effective in stimulating users to provide feedback on recommendations and find their liked songs.

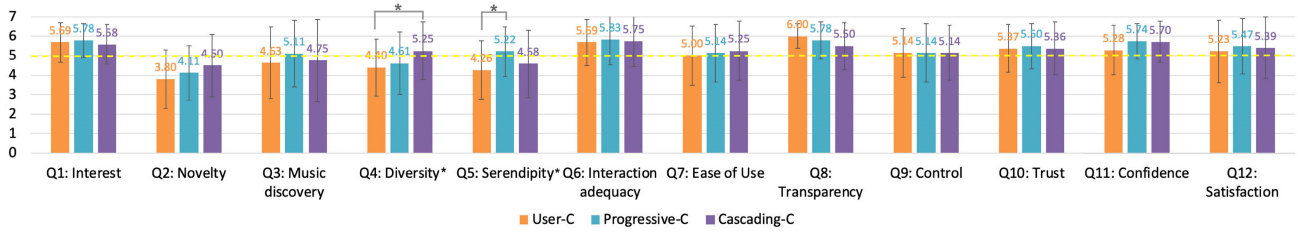
Recommendation performance metrics: Moreover, when we analyzed participants' listened songs and their liked songs, we find that they positively rated their liked five songs, with the average ratings above 4 out of 5 stars in all conditions. Also, users can find songs that suit their tastes from a new genre (compared with their initial profiles) during the interaction with the three critiquing systems when they performed BRT. All the three systems show no difference in terms of recommendation performance in BRT.

Critiquing behavior: Furthermore, to investigate in-depth the role of critiquing techniques for users to accomplish BRT, we analyzed their interaction data with a focus on their critiquing behavior. Specifically, we analyzed the actual use of UC and SC in the three different conditions. We counted the use of SC as requested by users by clicking the "*Let bot suggest*" button. As shown in Table VI, participants used UC more frequently in User-C than in Progressive-C and Cascading-C; more participants used SC in Progressive-C than in Cascading-C. In total, we find that 72 out of 117 users used UC in the three conditions and 24 out of 70 used SC in the two hybrid conditions that provide SC.

Since SC can be triggered by either clicking the "*Let bot suggest*" button (*Reactive SC*) or being proactively suggested by the system (*Proactive SC*), we calculated the acceptance rates of Reactive SC and Proactive SC in both Progressive-C and Cascading-C, respectively. The results show that the acceptance rate of Reactive SC in Cascading-C (100.00%) is higher than that in Progressive-C (86.84%), but the acceptance rate of Proactive



(a)



(b)

Fig. 4. Assessment results of statements related to user perception. A cut off value at 5 represents agreement on the 7-point Likert scale. * is marked for significant difference at the 5% level (p-value < 0.05). (a) Basic recommendation task (BRT). (b) Exploration-oriented task (EOT).

TABLE V
DESCRIPTIVE STATISTICS FOR USER INTERACTION BEHAVIOR DATA (SIGNIFICANCE: *** $p<.001$, ** $p<.01$, * $p<.05$)

	Basic recommendation task (BRT)							Exploration-oriented task (EOT)						
	User-C		Progressive-C		Cascading-C			User-C		Progressive-C		Cascading-C		
	Mean	Std	Mean	Std	Mean	Std	p	Mean	Std	Mean	Std	Mean	Std	p
Interaction metrics														
#Listened songs	13.06	8.51	10.73	4.59	11.09	3.93		42.06	12.92	39.78	12.97	41.47	15.62	
Duration (minutes)	3.93	1.51	3.74	1.78	4.82	2.33		10.95	4.43	12.04	4.59	12.47	5.28	
#Dialogue turns (times)	13.25	8.86	14.29	6.81	14.54	6.22		43.03	13.86	52.64	16.44	54.22	21.30	*
#Button (times)	10.16	6.86	12.02	5.61	12.46	5.38	*	33.40	9.65	46.39	12.69	47.61	19.08	***
#Button-Next (times)	5.25	6.87	3.13	2.87	3.86	2.91		13.97	9.40	12.81	8.52	13.89	12.22	
#Typing (times)	3.19	3.80	2.38	2.63	2.20	2.87		9.94	8.17	6.42	7.62	6.78	5.40	*
#Words per utterance	2.54	1.97	2.52	2.25	2.69	2.02		3.32	1.12	2.72	1.72	3.66	1.54	
Recommendation performance metrics														
Avg Rating (created playlist)	/	/	/	/	/	/		4.27	0.33	4.37	0.40	4.28	0.38	
Avg Rating (5 songs)	4.31	0.47	4.32	0.45	4.29	0.53		4.70	0.30	4.72	0.37	4.74	0.49	
#New Genres (listened songs)	1.84	1.42	1.80	1.01	1.63	0.94		3.83	2.26	4.19	2.58	3.97	2.13	
#New Genres (created playlist)	/	/	/	/	/	/		2.71	1.62	2.69	1.45	3.14	1.88	
#New Genres (5 songs)	1.13	1.01	1.22	0.97	1.09	0.92		1.40	1.22	1.56	1.05	1.44	1.08	

TABLE VI
DESCRIPTIVE STATISTICS FOR THE ACTUAL USE OF UC AND SC, AND THE PROVENANCE OF LIKED SONGS

	Basic recommendation task (BRT)			Exploration-oriented task (EOT)		
	User-C	Progressive-C	Cascading-C	User-C	Progressive-C	Cascading-C
Actual use of UC						
Percentage of using UC	68.75% (22/32)	60.00% (27/45)	65.71% (23/35)	94.29% (33/35)	75.00% (27/36)	94.44% (34/36)
Average times of using UC per user	4.32	3.67	3.13	9.52	8.19	6.71
Actual use of SC						
Percentage of using SC	N/A	42.22% (19/45)	14.29% (5/35)	N/A	61.11% (22/36)	63.89% (23/36)
Average times of using SC per user	N/A	1.53	2.2	N/A	2.36	3.18
Acceptance of SC						
Acceptance rate (Reactive SC)	N/A	86.84%	100.00%	N/A	92.62%	77.43%
Acceptance rate (Proactive SC)	N/A	94.66%	91.37%	N/A	92.13%	80.71%
Provenance of 5 preferred songs						
# Recommendations before critiquing	41.87%	26.67%	27.43%	10.86%	3.89%	6.11%
UC	58.13%	26.67%	31.43%	89.14%	33.33%	50.00%
Reactive SC	N/A	14.67%	4.57%	N/A	11.67%	8.89%
Proactive SC	N/A	32.00%	36.57%	N/A	51.11%	35.00%

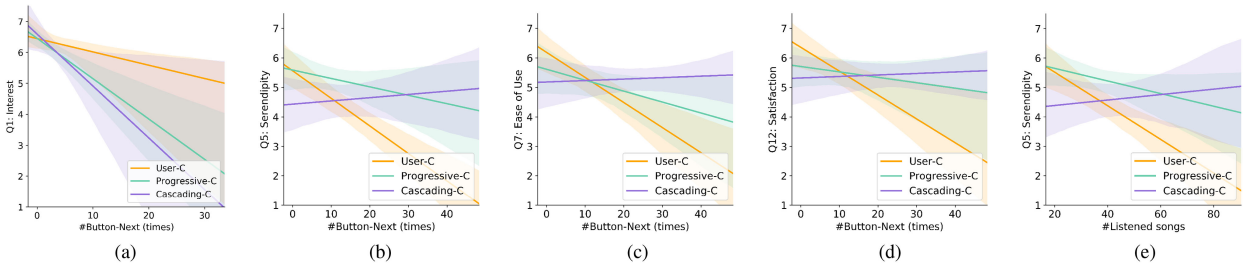


Fig. 5. Moderation effects of EC on the relationship between user interaction metrics and perception metrics. (a) The relationship between #Button-Next and Interest in BRT. (b) The relationship between #Button-Next and Serendipity in EOT. (c) The relationship between #Button-Next and Ease of Use in EOT. (d) The relationship between #Button-Next and Satisfaction in EOT. (e) The relationship between #Listened songs and Serendipity in EOT.

SC in Progressive-C is slightly higher (94.66%) than that in Cascading-C (91.37%).

Moreover, to investigate which kind of critiquing is more effective for users to find their liked songs, we analyzed the provenance of the five songs preferred by users (see Table VI). We find that users tend to find more songs of their interests from SC in Progressive-C and Cascading-C, suggesting that both *Progressive SC* and *Cascading SC* may help users find songs that suit their preferences.

3) Moderation effect of EC on the relationship between user interaction and user perception: In order to investigate how the three ECs moderate the relationship between user interaction and user perception of music recommendations when performing their tasks, we followed the procedure for moderation analysis as suggested by [38, Ch. 15]: First, we performed a Spearman's rank correlation analysis within each EC, and tested the significance of the difference between paired correlation coefficients by applying the Fisher-Z-Transformation [39]. This step serves as a preliminary analysis to assess the potential moderation of EC on the relationship between user interaction metrics and user perception metrics. Second, for the possible presence of moderation, we carried out a moderated regression analysis to examine the influence of EC (moderating variable) on the relationship between two variables (i.e., an interaction metric and a perception metric). Moderation effects are detected when the interaction term is statistically significant in the regression model.

Results show that EC only moderates the relationship between the number of "*Next*" button clicks and perceived interest matching ($F(2, 106) = 3.19$, $p<.05$) when users performed BRT. Users are more likely to perceive lower interest matching of recommendations if they skip more songs in Progressive-C and Cascading-C [see Fig. 5(a)], while this trend is not distinct in User-C probably because recommendations are mainly adjusted based on user-initiated critiques posted by users themselves.

Exploration-Oriented Task (EOT)

1) User perception: In the EOT, the results of comparative analysis indicate that the differences in terms of perceived diversity ($H = 6.81$, $df = 2$, $p<.05$) and perceived serendipity ($H = 7.64$, $df = 2$, $p<.05$) are significant among the three conditions. The post-hoc tests show that users perceived greater diversity of recommendations in Cascading-C ($M = 5.25$, $SD = 1.48$) than in User-C ($M = 4.40$, $SD = 1.46$, $p<.05$), and higher serendipity in Progressive-C ($M = 5.22$, $SD = 1.27$) than in User-C ($M = 4.26$, $SD = 1.52$, $p = .01$), but no significance is found in other pairwise comparisons. This may be explained by the fact that *Progressive SC* in Progressive-C can bring users different songs that are close to their interests, while *Cascading SC* in Cascading-C aims to introduce users to new types of music.

Similar to the result regarding BRT, users also gave high ratings (above 4 out of 5 stars) on the three critiquing systems in EOT in terms of their perceived interest matching, interaction adequacy, ease of use, transparency, control, trust, confidence, and satisfaction [see Fig. 4(b)].

2) User interaction.

Interaction metrics: In terms of user interaction, there are significant differences among the three conditions in EOT regarding dialogue turns ($H = 7.75$, $df = 2$, $p<.05$), times of clicking buttons ($H = 20.22$, $df = 2$, $p<.001$), and times of typing ($H = 6.13$, $df = 2$, $p<.05$). The post-hoc tests show that both Cascading-C and Progressive-C led to significantly more dialogue turns than User-C ($p<.05$), and users clicked significantly more buttons in Progressive-C and Cascading-C than in User-C ($p<.005$). One explanation for these results might be that the design of SC may introduce more dialogue turns and button clicks. To better understand how SC influences user interaction, see a dialogue example that illustrates how a user interacted with the hybrid system (that supports both UC and SC) in Appendix A.

Recommendation performance metrics: Table V summarizes the ratings of songs in users' created playlists and the number of newly explored genres in each case (i.e., listened songs, created playlist, and selected the top-5 preferred songs). It shows that users gave high ratings (above 4 out of 5 stars) on their liked songs in all conditions. Moreover, by comparing with users' preferred genres in their initial profiles, we find that all the three critiquing systems succeeded in getting users to explore 2 to 3 new genres as shown in their created playlists.

Critiquing behavior: Users' critiquing behaviors in EOT are similar to those in BRT (see Table VI). First, they used UC more often in User-C than in Progressive-C and Cascading-C in both EOT and BRT. Second, similar proportions of participants used SC in the two hybrid systems when performing both tasks. In EOT, they used SC relatively more frequently in Cascading-C than in Progressive-C. In total, 94 out of 107 users used UC, and 45 out of 72 used SC in the two hybrid conditions in EOT.

When further looking at the acceptance rates of the two activation manners of SC (i.e., Reactive SC and Proactive SC), we find that Progressive-C leads to higher acceptance rates of

both Reactive SC (92.62%) and Proactive SC (92.13%) than Cascading-C (77.43% and 80.71%, respectively). This probably implies that users might be prone to accept the progressive SC that matches their current preferences [11]. The results also show that the way of triggering SC seems to have little impact on user acceptance of SC.

We also analyzed the provenance of the top-5 songs preferred by users to see the effectiveness of different critiquing methods in helping users to perform EOT. As shown in Table VI, more than half of users' selected top-5 preferred songs are from Proactive SC in Progressive-C, whereas in Cascading-C, half of their most preferred songs are from UC. This may suggest that Progressive SC is more effective in helping users discover music that suits their tastes in Progressive-C, while users seem to explore more preferred music by actively critiquing the recommendation (i.e., UC) even when the system-suggested critiques are offered in Cascading-C.

In addition, to better understand when users would like to make critiques for exploring diverse music, we analyzed users' interaction flows, and have two major observations: 1) Users tend to use UC to explore new songs after they clicked the "*Like*" or "*Next*" on three recommended songs consecutively; 2) Users tend to request SC after accepting one or more critiques proactively suggested by the system, namely that some users are more likely to trigger Reactive SC if they have benefited from Proactive SC. For the detailed analysis of this part, please refer to our prior publication [26].

3) Moderation effect of EC on the relationship between user interaction and user perception: In this part of analysis regarding EOT, we find that EC moderates the relationships between the number of "*Next*" button clicks and three user perception metrics: perceived serendipity ($F(2, 101) = 4.99$, $p<.01$), perceived ease of use ($F(2, 101) = 3.71$, $p<.05$), and satisfaction ($F(2, 101) = 3.16$, $p<.05$). Figs. 5(b)–(d) show a tendency that users who clicked more "*Next*" buttons seem to have lower perception of serendipity, ease of use and satisfaction in User-C and Progressive-C, while an opposite tendency is shown in Cascading-C. Compared with User-C and Progressive-C, Cascading-C can produce more diverse songs along with more user interactions, which may in turn enhance user perception metrics related to music exploration. Also, EC moderates the relationship between the number of listened songs and perceived serendipity ($F(2, 101) = 3.51$, $p<.05$). Fig. 5(e) shows that users who listened to more songs tend to perceive lower serendipity in User-C and Progressive-C. On the contrary, users in Cascading-C tend to perceive higher serendipity when listening to more songs. In short, these results of EOT reflect that critiquing techniques are more likely to influence the correlations between some interaction metrics and perception metrics.

VI. Discussions

In this section, we discuss our major findings in response to the two research questions raised at the beginning (summarized in Table VII). We also offer some practical implications for designing critiquing-based recommendation chatbots.

RQ1: **How do task types influence users' perception of and interaction with the three critiquing systems?** We conducted two task-oriented user studies that consider two typical types of user tasks: basic recommendation task (BRT) and exploration-oriented task (EOT), to investigate the influence of task type on user perception of and interaction with the three proposed critiquing systems. The results of our studies indicate that task type induces a significant impact. While EOT stimulates more user interactions, such as more listened songs, more dialogue turns, and more button clicks, BRT leads to more positive user experiences. A previous study on music discovery [40] demonstrated that making a playlist with different mindsets (i.e., focused, open, and exploratory mindset) leads to different interaction behavior with the system and perceptions of recommendations. In our studies, compared to BRT where users are likely in an open mindset to find music relevant to their interests, EOT requires users to actively step outside of their "comfort zones" to explore and try diverse music recommendations with an exploratory mindset [40]. As the latter can be more risky (i.e., with some uncertainty) and challenging [12], it may undermine the perceived quality of recommendations if the additional user exploration efforts fail in developing new music preferences. These findings are also consistent with the previous observations in the information retrieval domain [25], [41] that the exploratory tasks (i.e., browsing) require more interactions between users and the system than the searching tasks. It is therefore of vital importance to take into account the task type when designing recommendation chatbots.

TABLE VII
Summary of the Major Findings in Our Studies

	User Perception	User Interaction
Effects of Task Type (RQ1)		
BRT vs. EOT	BRT > EOT : - Interest - Interaction adequacy - Transparency - Control - Satisfaction	EOT > BRT: - # Listened songs - Duration & # Dialogue turns - # Button (times) - # Tying (times) - # Words per utterance
Effects of Critiquing Technique (RQ2)		
BRT		Cascading-C > User-C: - # Button (times)
EOT	Progressive-C > User-C: - Serendipity Cascading-C > User-C: - Diversity	Progressive-C & Cascading-C > User-C: - # Dialogue turns - # Button (times)

Note: A > B indicates that the performance of A is significantly better than that of B regarding some particular metrics.

RQ2: **How do critiquing techniques influence user perception and interaction in the basic recommendation task and the exploration-oriented task respectively?** We compared users' perception of conversational recommendations and their interaction behavior data among the three ECs (i.e., User-C, Progressive-C, and Cascading-C) in the two user tasks, respectively (i.e., BRT and EOT). The results show that when users performed BRT, the critiquing techniques did not lead to any significant differences in terms of user perceptions. When they performed EOT, on the other hand, *Cascading SC* is more effective in helping users discover diverse songs, while *Progressive SC* helps users find more songs with serendipity. This may be

because while both techniques enable the system to proactively provide suggestions, the latter offers suggestions considering users' feedback to facilitate their exploration, which likely generates serendipitous recommendations that are not only relevant to users' expectations but also a pleasant surprise [42]. The *Cascading SC*, however, focuses on guiding users to explore new genres, which introduces different types of music and is likely to be perceived by users as diverse [29]. The higher user-perceived serendipity brought by *Progressive SC* in EOT is also reflected in users' greater tendency of picking the songs recommended by it as their top 5 preferred song. As mentioned previously, recommendations introduced by *Progressive SC* are more likely to be favoured by users because they consider users' feedback in the previous interactions. The users' feedback, however, plays a lesser role in *Cascading SC*, because it mainly aims at guiding users to explore more different music genres.

The comparison results about interaction behavior data also show that SC results in more button clicks for both tasks, and more dialogue turns for EOT, which are in line with the findings of a previous user study [7]. Moreover, it is found that critiquing techniques significantly moderate the relationships between some interaction metrics (e.g., the number of listened songs, number of "*Next*" button clicks) and users' perceived serendipity and satisfaction in EOT. Cascading-C exhibits a positive correlation between user interaction and user perception; User-C and Progressive-C, however, show the opposite results. This may be related to that, when the task is exploration-oriented like EOT, users who try more music are likely to anticipate various kinds of music in an exploratory mindset [40]. In Cascading-C, users who interact more with the system are likely to encounter more diverse types of music and even some surprising discoveries, thereby perceiving higher serendipity and having a better experience. On the contrary, in User-C and Progressive-C, more user interactions may not improve user experience, probably because the recommended songs rely more on users' incremental preferences rather than steering them into new music tastes. These results suggest that the strengths of different critiquing techniques should be well noted when chatbots are designed to serve different purposes.

Implications of our work: Our studies inform that *task type* should be taken into account during the design and evaluation of critiquing-based recommendation chatbots, as it may lead to different user perceptions and interaction behaviors.

To be specific, for a less demanding task (like BRT, i.e., finding songs based on the user's preferences), users perceive no much difference on the system with or without richer supported critiques (i.e., SC), because it normally takes less interaction effort for accomplishing this task. Thus, the practitioners may choose either only UC or the hybrid critiquing approach that incorporates both UC and SC when designing critiquing-based systems for BRT.

As for supporting a relatively high demanding task (such as EOT, i.e., exploring diverse types of songs), effective critiquing techniques can positively influence user perception since they may enhance users' exploration interaction. In particular, our results show that UC allows users to explicitly initialize exploration when they have a clear exploration goal, while SC guides users to explore recommendations when they have no specific goal [26]. Thus, it might help to provide the hybrid critiquing approach (both UC and SC) for supporting EOT. Regarding the two types of SC, practitioners may choose between *Progressive SC* and *Cascading SC* according to whether the exploration is mainly for serendipity or diversity.

Moreover, for the exploration-oriented task, the period of exploring music may also influence the choice of SC. *Progressive SC* can be more helpful in the initial period of exploration when users are more likely to accept the songs that are close to their current preferences. *Cascading SC*, on the other hand, can be more useful in the later period of exploration when users reasonably expect to see more diverse types of songs.

VII. Limitations

This study has three major limitations. First, the proactive SC in the two hybrid critiquing systems is triggered under the preset condition. More flexible methods would be desired for identifying more appropriate timing to proactively offer SC. Second, our studies have a relatively small sample size, which may undermine the power of the statistical analysis. Third, we validate our results with only music recommendation domain. The results need to be further validated in other domains.

VIII. Conclusion

In conclusion, we performed two task-oriented user studies, focusing on two typical types of user tasks (i.e., BRT and EOT), to compare three critiquing-based recommendation chatbots (i.e., User-C, Progressive-C, and Cascading-C) in terms of user perception and interaction behavior. Results show that the type of user task significantly impacts user perception of and interaction with the three systems. Compared with performing BRT, critiquing techniques exert more influence on user perception when they perform EOT. In particular, it is found that system-suggested critiquing (SC) in the two hybrid systems (Progressive-C and Cascading-C) can significantly increase users' perceived serendipity and diversity when the task is exploration-oriented. In addition, in EOT, critiquing techniques bring about significant moderation effects on the relationship between user interaction and user perception.

Overall, our work sheds light on leveraging critiquing techniques to enhance the interaction between users and recommendation chatbots. In the future, we plan to investigate the impact of users' personal characteristics such as personality since individual differences may also influence the way users interact with a computer system in different tasks [2], [15]. We also intend to verify if the findings can be generalized to other application domains such as e-commerce.

APPENDIX A
DIALOGUE EXAMPLE

TABLE A1

Role	Utterances
Bot	Hi there. Now you need to create a playlist that contains 20 good songs. I have found some songs for you based on your preference, but you can also search for other songs by using the tips shown on the right side. We recommend this song because you like the songs of pop, rock, folk. Music Recommendation: Chasing Cars
User	Next song. (The user clicks the "*Next*" button.)
Bot	Music Recommendation: Girls
User	I need more valence. (The user gives a **user-initiated critique**.)
Bot	Just play these songs because you want higher valence. Music Recommendation: Jockey Full Of Bourbon
User	I like this song. (The user clicks the "*Like*" button.)
Bot	You also need to rate the song in terms of pleasant surprise in the left panel. (After the user rate the song) OK, I recommend this song to you, because you like folk music. Music Recommendation:Cathedral
User	Next song. (The user clicks the "*Next*" button.)
Bot	Music Recommendation: Life Eternal
User	I need some suggestions. (The user clicks the ***"Let bot suggest"*** button.)
Bot	Compared with the last played song, do you like the song of higher energy? (The system provides a **system-suggested critique**.)
User	I don't want. (The user clicks the "*No*" button to reject the suggestion.)
Bot	Compared with the last played song, do you like the song of lower energy? (The system provides a **system-suggested critique**.)
User	Yes, please! (The user clicks the "*Yes*" button to accept the suggestion.)
Bot	Music Recommendation: Love Hurts
User	i need more energy. (The user gives a **user-initiated critique**.)
Bot	Just play these songs because you want higher energy. ... ... (After the user consecutively listened to five songs)
Bot	Compared with the last played song, do you like the song of alternative metal? (The system **proactively offers a system-suggested critique**.)
User	Yes, please! (The user clicks the "*Yes*" button to accept the suggestion.)
Bot	Music Recommendation: Whore
	... ...

ACKNOWLEDGMENT

Dr. Meilin Chen at the Language Centre of HKBU helped improve the linguistic presentation of the manuscript. The authors would like to thank the Editors and Reviewers for their constructive suggestions and comments and also like to thank all participants for their time in taking part in our experiment.

REFERENCES

[1] T. T. Taijala, M. C. Willemsen, and J. A. Konstan, "Movieexplorer: Building an interactive exploration tool from ratings and latent taste spaces," in *Proc. 33rd Annu. ACM Symp. Appl. Comput.*, 2018, pp. 1383–1392.

[2] Y. Jin, N. Tintarev, N. N. Htun, and K. Verbert, "Effects of personal characteristics in control-oriented user interfaces for music recommender systems," *User Model. User-Adapted Interact.*, vol. 30, no. 2, pp. 199–249, 2020.

[3] D. Jannach, A. Manzoor, W. Cai, and L. Chen, "A survey on conversational recommender systems," *ACM Comput. Surv.*, vol. 54, no. 5, pp. 1–36, May 2021.

[4] K. Christakopoulou, F. Radlinski, and K. Hofmann, "Towards conversational recommender systems," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2016, pp. 815–824.

[5] W. Cai and L. Chen, "Predicting user intents and satisfaction with dialogue-based conversational recommendations," in *Proc. 28th ACM Conf. User Model., Adapt. Personalization*, 2020, pp. 33–42.

[6] L. Chen and P. Pu, "Critiquing-based recommenders: Survey and emerging trends," *User Model. User-Adapted Interact.*, vol. 22, no. 1/2, pp. 125–150, Apr. 2012.

[7] Y. Jin, W. Cai, L. Chen, N. N. Htun, and K. Verbert, "Musicbot: Evaluating critiquing-based music recommenders with conversational interaction," in *Proc. 28th ACM Int. Conf. Inf. Knowl. Manage.*, 2019, pp. 951–960.

[8] J. Reilly, K. McCarthy, L. McGinty, and B. Smyth, "Incremental critiquing," in *Proc. Int. Conf. Innov. Techn. Appl. Artif. Intell.*, 2004, pp. 101–114.

[9] C. Li, H. Feng, and M. D. Rijke, "Cascading hybrid bandits: Online learning to rank for relevance and diversity," in *Proc. 14th ACM Conf. Recommender Syst.*, 2020, pp. 33–42.

[10] D. Jannach and G. Adomavicius, "Recommendations with a purpose," in *Proc. 10th ACM Conf. Recommender Syst.*, 2016, pp. 7–10.

[11] Y. Liang and M. C. Willemsen, "Personalized recommendations for music genre exploration," in *Proc. 27th ACM Conf. User Model., Adapt. Personalization*, 2019, pp. 276–284.

[12] P. Knees, M. Schedl, and M. Goto, "Intelligent user interfaces for music discovery," *Trans. Int. Soc. Music Inf. Retrieval*, vol. 3, pp. 165–179, 2020.

[13] M. Taramigkou, E. Bothos, K. Christidis, D. Apostolou, and G. Mentzas, "Escape the bubble: Guided exploration of music preferences for serendipity and novelty," in *Proc. 7th ACM Conf. Recommender Syst.*, 2013, pp. 335–338.

[14] J. Liu *et al.*, "Search behaviors in different task types," in *Proc. 10th Annu. Joint Conf. Digit. Libraries*, 2010, pp. 69–78.

[15] C. Liu, J. Liu, M. Cole, N. J. Belkin, and X. Zhang, "Task difficulty and domain knowledge effects on information search behaviors," *Proc. Amer. Soc. Inf. Sci. Technol.*, vol. 49, no. 1, pp. 1–10, 2012.

[16] Y. Li, X. Yuan, and R. Che, "An investigation of task characteristics and users' evaluation of interaction design in different online health information systems," *Inf. Process. Manage.*, vol. 58, no. 3, 2021, Art. no. 102476.

[17] H. Shimazu, "Expertclerk: Navigating shoppers' buying process with the combination of asking and proposing," in *Proc. 17th Int. Joint Conf. Artif. Intell.*, 2001, pp. 1443–1448.

[18] C. A. Thompson, M. H. Goker, and P. Langley, "A personalized system for conversational recommendations," *J. Artif. Intell. Res.*, vol. 21, no. 1, pp. 393–428, Mar. 2004.

[19] J. Kang, K. Condiff, S. Chang, J. A. Konstan, L. Terveen, and F. M. Harper, "Understanding how people use natural language to ask for recommendations," in *Proc. 11th ACM Conf. Recommender Syst.*, 2017, pp. 229–237.

[20] L. Yang, M. Sobolev, C. Tsangouri, and D. Estrin, "Understanding user interactions with podcast recommendations delivered via voice," in *Proc. 12th ACM Conf. Recommender Syst.*, 2018, pp. 190–194.

[21] M. Jain, P. Kumar, R. Kota, and S. N. Patel, "Evaluating and informing the design of chatbots," in *Proc. Designing Interactive Syst. Conf.*, 2018, pp. 895–906.

[22] Z. Peng, Y. Kwon, J. Lu, Z. Wu, and X. Ma, "Design and evaluation of service robot's proactivity in decision-making support process," in *Proc. CHI Conf. Hum. Factors Comput. Syst.*, 2019, pp. 1–13.

[23] S. M. McNee, J. Riedl, and J. A. Konstan, "Making recommendations better: An analytic model for human-recommender interaction," in *Proc. Extended Abstr. Hum. Factors Comput. Syst.*, 2006, pp. 1103–1108.

[24] B. P. Knijnenburg, M. C. Willemsen, Z. Gantner, H. Soncu, and C. Newell, "Explaining the user experience of recommender systems," *User Model. User-Adapted Interact.*, vol. 22, no. 4/5, pp. 441–504, 2012.

[25] X. Hu and N. Kando, "Task complexity and difficulty in music information retrieval," *J. Assoc. Inf. Sci. Technol.*, vol. 68, no. 7, pp. 1711–1723, 2017.

[26] W. Cai, Y. Jin, and L. Chen, "Critiquing for music exploration in conversational recommender systems," in *Proc. 26th Int. Conf. Intell. User Interfaces*, 2021, pp. 480–490.

[27] J. Zhang and P. Pu, "A comparative study of compound critique generation in conversational recommender systems," in *Proc. Int. Conf. Adaptive Hypermedia Adaptive Web-Based Syst.*, 2006, pp. 234–243.

[28] L. Chen and P. Pu, "Preference-based organization interfaces: Aiding user critiques in recommender systems," in *Proc. Int. Conf. User Model.*, 2007, pp. 77–86.

[29] P. Castells, S. Vargas, and J. Wang, "Novelty and diversity metrics for recommender systems: Choice, discovery and relevance," in *DDR-2011: International Workshop on Diversity in Document Retrieval at the ECIR 2011*, 2011, pp. 29–36.

[30] K. Lee and K. Lee, "Escaping your comfort zone: A graph-based recommender system for finding novel recommendations among relevant items," *Expert Syst. Appl.*, vol. 42, no. 10, pp. 4851–4858, 2015.

[31] W. Wu, L. Chen, and Y. Zhao, "Personalizing recommendation diversity based on user personality," *User Model. User-Adapted Interact.*, vol. 28, no. 3, pp. 237–276, Aug. 2018.

[32] E. Peer, L. Brandimarte, S. Samat, and A. Acquisti, "Beyond the turk: Alternative platforms for crowdsourcing behavioral research," *J. Exp. Social Psychol.*, vol. 70, pp. 153–163, 2017.

[33] F. Faul, E. Erdfelder, A. Buchner, and A. G. Lang, "Statistical power analyses using G* power 3.1: Tests for correlation and regression analyses," *Behav. Res. Methods*, vol. 41, no. 4, pp. 1149–1160, 2009.

[34] Y. Jin, N. Tintarev, and K. Verbert, "Effects of individual traits on diversity-aware music recommender user interfaces," in *Proc. 26th Conf. User Model., Adapt. Personalization*, 2018, pp. 291–299.

[35] J. Kumar and N. Tintarev, "Using visualizations to encourage blind-spot exploration," in *Proc. Recsys Workshop Interfaces Decis. Mak. Recommender Syst.*, 2018, pp. 53–60.

[36] L. Chen and P. Pu, "Evaluating critiquing-based recommender agents," in *Proc. 21st Nat. Conf. Artif. Intell.-Volume 1*, 2006, pp. 157–162.

[37] C. Matt, A. Benlian, T. Hess, and C. Weiβ, "Escaping from the filter bubble? the effects of novelty and serendipity on users' evaluations of online recommendations," in *Proc. 35th Int. Conf. Inf. Syst.*, 2014, ppp. 1503–1520.

[38] R. M. Warner, *Applied Statistics: From Bivariate Through Multivariate Techniques*. Newcastle upon Tyne, U.K.: Sage, 2012.

[39] W. Lenhard and A. Lenhard, "Hypothesis tests for comparing correlations," *Bibergau, Germany: Psychometrica*, 2014.

[40] C. Hosey, L. Vujović, B. St. Thomas, J. Garcia-Gathright, and J. Thom, "Just give me what I. want: How people use and evaluate music search," in *Proc. Conf. Human Factors Comput. Syst.* 2019, pp. 1–12.

[41] J. Arguello, W. C. Wu, D. Kelly, and A. Edwards, "Task complexity, vertical display and user interaction in aggregated search," in *Proc. 35th Int. ACM SIGIR Conf. Res. Develop. Inf. Retrieval*, 2012, pp. 435–444.

[42] M. Kaminskas and D. Bridge, "Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems," *ACM Trans. Interactive Intell. Syst.*, vol. 7, no. 1, pp. 1–42, 2016.