User Experience and The Role of Personalization in Critiquing-Based Conversational Recommendation

ARPIT RANA, University of Toronto, Toronto, ON, Canada
SCOTT SANNER, University of Toronto, Toronto, ON, Canada
MOHAMED REDA BOUADJENEK, Deakin University, Australia
RON DICARLANTONIO, iNAGO Inc. Toronto, ON, Canada
GARY FARMANER, iNAGO Inc. Toronto, ON, Canada

Critiquing — where users propose directional preferences to attribute values — has historically been a highly popular method for conversational recommendation. However, with the growing size of catalogs and item attributes, it becomes increasingly difficult and time-consuming to express all of one's constraints and preferences in the form of critiquing. It is found to be even more confusing in case of critiquing failures: when the system returns no matching items in response to user critiques. To this end, it would seem important to combine a critiquing-based conversational system with a personalized recommendation component to capture implicit user preferences and thus reduce the user's burden of providing explicit critiques. To examine the impact of such personalization on critiquing, this paper reports on a user study with 228 participants to understand user critiquing behavior for two different recommendation algorithms: (i) *non-personalized*, that recommends any item consistent with the user critiques; and (ii) *personalized*, which leverages a user's past preferences on top of user critiques. In the study, we ask users to find a restaurant that they think is the most suitable to a given scenario by critiquing the recommended restaurants at each round of the conversation on the dimensions of price, cuisine, category, and distance. We observe that the *non-personalized* recommender leads to more critiquing interactions, more severe critiquing failures, overall more time for users to express their preferences, and longer dialogs to find their item of interest. We also observe that *non-personalized* users were less satisfied with the system's performance. They find its recommendations less relevant, more unexpected, and somewhat equally diverse and surprising than those of *personalized* ones. The results of our user study highlight an imperative for further research on the integration of the two complementary components of *personalization* and *critiquing* to achieve the best overall user experience in future critiquing-based conversational recommender systems.

CCS Concepts: • **Information systems → Recommender systems**; **Personalization**; • **Human-centered computing → User studies**; **Natural language interfaces**.

Additional Key Words and Phrases: Conversational Recommendations, Critiquing Failure, User Study, User Experience, Failure Analysis

ACM Reference Format:
Arpit Rana, Scott Sanner, Mohamed Reda Bouadjenek, Ron Dicarlantonio, and Gary Farmaner. 2023. User Experience and The Role of Personalization in Critiquing-Based Conversational Recommendation. *ACM Trans. Web* 1, 1, Article 1 (January 2023), 22 pages. https://doi.org/10.1145/3597499

Authors' addresses: Arpit Rana, University of Toronto, Toronto, ON, Canada, arana@mie.utoronto.ca; Scott Sanner, University of Toronto, Toronto, ON, Canada, ssanner@mie.utoronto.ca; Mohamed Reda Bouadjenek, Deakin University, School of Information Technology, 75 Pigdons Rd, Geelong, Victoria, Australia, 3216, reda.bouadjenek@deakin.edu.au; Ron Dicarlantonio, iNAGO Inc. Toronto, ON, Canada, rond@inago.com; Gary Farmaner, iNAGO Inc. Toronto, ON, Canada, garyf@inago.com.

1559-1131/2023/1-ART1 $15.00
https://doi.org/10.1145/3597499

1 INTRODUCTION

The process by which a user selects an item to consume (e.g. a restaurant to eat out) is often an iterative one: the user's requirements may not be fully observable (e.g. context, her mood, her ephemeral goals, etc.) or may be uncertain [29]. Conversational recommender systems (CRSs) can handle such cases by allowing repeated interactions between the user and the system. Typically, they propose a set of recommendations and invite the user to evaluate the recommended items and provide their feedback to refine these recommendations [38]. Specifically, in critiquing-based conversational recommendations, users propose 'tweaks' (e.g. "like this but cheaper") or 'replacements' (e.g. "like this with Italian Food") to attribute values which filter out the candidates and that would ultimately improve the recommendations [4].

Early critiquing-based systems assume that a user will be satisfied with any items meeting the critique constraints. However, with an increase in the size of item catalogs, the number of item attributes, and the size of the attribute domains, it becomes increasingly difficult for the users to express their preferences [29]. This is even more critical in trade-off situations where users have to decide what to retain and what to compromise [5], or in critiquing failures where the system returns no matching items [23] and users either have to compromise with partially satisfying items or revise their preferences [29]. *Trade-off navigation* [4], *soft navigation* [21], and *progressive critiquing* are a few techniques among others that have been proposed to handle trade-off conflicts and failures. However, such techniques may not scale-up well with the growing number of item features and also do not accommodate user's past preferences while resolving dialog inconsistencies. Here, it is important for the system to identify the feature values causing conflicts and failures, and subsequently their replacements that are in line with a user's preferences. These instances motivate us to investigate the role of personalization in critiquing — a topic we believe has been under-emphasized in the existing critiquing literature.

In this work, we seek to evaluate the importance of the personalization component in a critiquing-based conversational recommender especially when there are cases of critiquing failures. To achieve this, we present results of a user study with 228 participants in which we assign a scenario to the users (e.g. a formal lunch with a business client) and ask them to find a suitable restaurant by critiquing the recommendations that the system provides to them. We specifically explore two basic recommendation algorithms: (i) a *non-personalized* recommender (*NP-Rec*), that *solely* relies on user critiques; and (ii) a *personalized* recommender (*kNN*), which is a nearest neighbor-based collaborative filtering recommendation system [17] that leverages both a user's past preferences and her critiques. We compare the two recommendation algorithms in a between-subject trial and seek to answer the following questions:

RQ1 How does *personalization* affect occurrences of critiquing failures?
RQ2 How does *personalization* affect the amount of effort a user needs to reach an item of her interest?
RQ3 How does *personalization* affect the overall dialog flow?
RQ4 How does *personalization* affect the quality of recommendations in both an objective and subjective manner?
RQ5 What differences do users perceive between the dialogs of a *non-personalised* and a *personalised* recommendation algorithm?

Among a range of results that we discuss in our experimental section, we found that users assigned to the *kNN* recommender critiqued less, finished faster, spent less time making explicit critiques that were already implicitly captured by the recommender system, and found its recommendations to be more relevant and still competitive in terms of diversity and surprise.

In short, we can infer from our experimental results that a system which offers personalized recommendations can reduce critiquing burden on the user by implicitly capturing preferences and constraints that the user would otherwise have had to express as explicit critiques. As elaborated in our related work discussion, to the best of our knowledge, this is the first study to emphasize the importance of personalization on user experience for critiquing-based conversational systems.

2 RELATED WORK

Interactive conversational recommendation has shown significant advantages over single-shot recommendation [14], when (a) users have ephemeral goals different from their usual tastes, (b) users are not satisfied with initial top-n recommendations, and (c) user requirements are uncertain or are not fully observable [2, 13]. Historically, conversational recommendation has been common in knowledge-based (i.e., constraint-based) recommender systems [3], where the recommender suggests items to best satisfy the user's preferences; it is one approach to context-aware [1] and context-driven [26] recommendation, where the user can give feedback to steer the recommendations toward ones that best suit the context [12]. Many conversational recommender systems use GUI-based interactive recommendation [2, 37], which allows users to provide their feedback in one of four ways: (i) by asking questions for a value of a specific item feature [36]; (ii) by collecting user ratings on the proposed recommendations [39], (iii) by inviting users to select one of many recommendations [32], or, (iv) by allowing users to provide critiques on item features [40]. These feedback forms differ in their level of ambiguity and the efforts they demand from the user [38].

Critiquing is one important form of feature-level feedback in conversational recommendation, where instead of providing a specific value for an attribute, users propose 'tweaks' to attribute values to refine their recommendations [4]. Such 'tweaks' can be applied on one attribute (*unit* [4]) or on multiple attributes all-together (*compound* [33]), and they can be system-suggested [4] or user-initiated [9] or even the combination of the two [40]. Critiquing provides a relatively unambiguous indication of the user's current preferences, imposes low burden on the user, and might even be usable by users with minimal understanding of the item features [38]. Because of these characteristics, critiquing has been extensively explored in the literature.

Studies have also shown that online navigation tools can significantly increase decision accuracy by helping users select and compare options that share trade-off properties [29]. System-suggested critiquing is one such effort that pro-actively generates a set of knowledge-based critiques that users might accept as ways to improve the current recommendation. For example, the FindMe system in [4] provides critique suggestions that are pre-designed and fixed within a user's whole interaction session, so they are unable to reflect the user's changing needs and the status of remaining available products. For system-suggested approaches, some people believe that compound critiquing strategies potentially lead to higher accuracy with shorter dialog length [42]; others disagree [6]. Unlike system-suggested, user-initiated critiquing allows users to make self-motivated critiques: users can post either unit or compound critiques over any combination of features with freedom [28]. However, in user-initiated critiquing, a user remains unaware of available options and therefore it becomes difficult to express all of one's constraints. Subsequently, it was soon realized that the respective strengths of system-suggested and user-initiated critiquing could well compensate each other and hybrid approaches were proposed [7, 8, 40]. The sizable body of critiquing work is surveyed in [10]. There is additionally a small amount of work that combines question-answering with critiquing, e.g. [36].

In contrast to the previously surveyed critiquing work that used an explicit attribute-value representation for structured item descriptions, other methods have explored less structured representations. For example, Vig et al. [40] extend the critiquing idea to items whose descriptions are (very large) sets of tags. More recently, a range of papers [18, 19, 24, 41] have proposed methods for using critiquing feedback to modulate latent embeddings of user preferences in recommendation systems, though all focused on synthetic validation of critiquing performance.

Despite the widespread use of critiquing in conversational recommendation, it is not without its limitations [22]. Critiquing has primarily been utilized as a method for filtering candidates consistent with user preferences (a type of contextual pre-filtering [27]). This usage can often lead to a *critiquing failure* where the system is unable to retrieve any candidates consistent with the user critiques [23]. Conflicting preferences and trade-offs are also quite common in critiquing such that users either accept a partially satisfying item or must otherwise revise their preferences [29]. Most past work on critiquing handle such conflicts and failures by helping users to revise their preferences, for example, through *trade-off navigation* [4], by maximally satisfying subsets of the stated preferences through *soft navigation* [21], or by allowing the system to re-recommend items that have already been suggested in the previous rounds of the dialog [23]. However, such techniques either may not be extended to items with unstructured representations or they may not resolve conflicts and failures as per the user's taste. In order to help a user find the item of her interest in an effective, efficient, and personalized manner, it is important for the system to identify the feature values that cause failures and their replacements that are in line with the user's tastes.

With the critiquing advances above, there have been a variety of user studies evaluating how critiquing-based systems can increase users' decision accuracy (e.g. [30]), save user effort (e.g. [9, 20, 31, 34]), and improve users' decision confidence (e.g. [28]). Further, recent studies examine the effect of personalization in conversational recommendations [35] and propose techniques to better handle ambiguity in user preferences [13, 16]. Rhee and Choi [35] study the persuasion mechanism in product recommendation on voice-based interaction with a conversational agent, which usually has no visual display. Specifically, they investigate whether the personalized content reflecting the customer‘s preferences and the agent‘s social role of a friend, rather than a secretarial assistant, generate a more positive attitude toward the product. In [13], He et al. proposes a novel memory network framework for conversational recommendation, which harnesses dialog historical information for reducing the ambiguity during interactions and improving the quality of conversational recommendation systems. However, to the best of our knowledge, there is no formal study that discusses the effect or importance of the underlying recommendation system (i.e., personalization) on the critiquing process, especially when the dialogs involve failures. We address this gap in our user study.

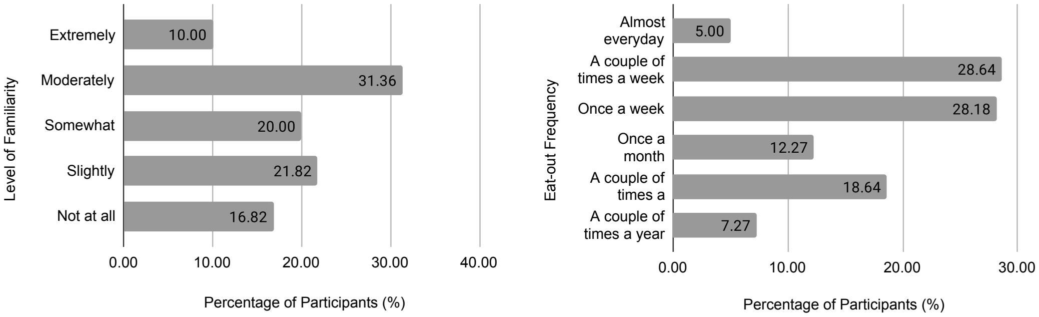
3 EXPERIMENTAL SETTINGS

To answer the research questions mentioned in the introduction, we designed a web-based online user study and compared two recommendation algorithms: one is *non-personalized* and another one is *personalized*, in a between-subject trial. We describe settings of our experiments in the following subsections.

3.1 Dataset

We used the Yelp open dataset[1] from which we selected restaurants data and reviews of the Greater Toronto Area (GTA). We selected this domain since most study participants would be expected to have extensive experience selecting restaurants and because the context we provide (e.g., a surprise dinner with parents) should have a strong influence on intrinsic user preferences as well as context-specific preferences (e.g., companion, occasion, etc.) to drive the conversational interaction. We restricted ourselves to restaurants for which the following attributes are available: price, cuisines,

[1] https://www.yelp.com/dataset



(a) Familiarity with Restaurants

(b) Eat-out (pre-COVID'19) frequency

Fig. 1. Details of Participants

categories (e.g., cafes, tea rooms, pubs), and neighborhood information. Thus, the dataset comprises a total of 3,628 restaurants, 87,162 users, and 332,135 user-reviews. On average, a typical restaurant has 91 reviews, ranging from 3 to 2,834, which shows a very high variance in the number of reviews. For each restaurant, we extracted the three most-frequent key-phrases to be shown to the user during the experiment (see Figure 2). Additionally, in order to increase the chances of familiarizing users with the cuisines and categories, we have selected the most frequent ones. This resulted in a total of 112 cuisines and 37 categories to be used in our experiment.

3.2 Algorithms

In our experiments, we specifically compare two recommendation algorithms:

(1) a *non-personalized* recommender, which we call *NP-Rec*, that randomly selects n restaurants consistent with the critiques to present to the user; and
(2) a *kNN*-based (*personalized*) recommender using cosine item-based similarity that selects the top-n[2] closest restaurants to the user profile that are consistent with the critiques.

The rationale for choosing these two systems is that *NP-Rec* relies solely on a user's critiques (without *any* further bias), whereas *kNN is* biased towards a user's past preferences in addition to filtering recommendations according to their critiques. Furthermore, we remark that *NP-Rec* replicates existing faceted search interfaces that only show items consistent with critique (facet) selections in different dimensions. In this sense, we would consider *NP-Rec* to be representative of the non-personalized state-of-the-art interface that we intended to compare to in this work.

One can think of using a *popularity-based* recommender (which is also a non-personalized recommender) in place of *NP-Rec*; however, at a given dialog state, *NP-Rec* considers all consistent candidates equally likely to be recommended whereas *popularity-based* recommendations are influenced by other users' opinions (this adds an additional layer of filtering on top of user critiques). Similarly, in the case of personalized recommendation, there are several more sophisticated models of recommendation; however, we chose *kNN* because we wanted this experiment to reveal the effect of the small difference between the two systems on the critiquing process. We otherwise tried to ensure that the two systems are as similar as possible.

[2] We chose $n = 3$ as manageable number for a user to look at each round of the dialog.

Table 1. List of locations and purposes that we used to create scenarios for our user study.

S.No.	Locations at GTA	Purpose of Restaurant Visit
1	Toronto Midtown (Yonge and Eglinton)	A Birthday Lunch with Coworkers
2	North York (Yonge and Sheppard)	A Brainstorming Lunch with your Boss
3	Thornhill (Yonge and Centre St.)	A Formal Lunch with a Business Client
4	Markham (Highway 7 and McCowan)	A Surprise Dinner with Parents
5	Scarborough (Ellesmere and McCowan)	A Get-together Dinner with Close Friends
6	Toronto Downtown (Queen and Bay)	A Celebration Dinner with Relatives including Children

3.3 Participants

We recruited participants online from our University. The majority of them were undergraduate and postgraduate students. Participants in the study were shown an experiment privacy policy; they were given the chance to opt-out or stop at any stage of the experiment. This experiment was approved by our institution's research ethics board (REB).[3] We did not explicitly collect demographic data, but it is most likely that they were predominantly young, Computer Science or Engineering students. They were not rewarded for participation. In total, 289 people attempted the user study in 8 weeks. Of 289 users, 228 completed all parts of the trial with the system they were assigned.

We also ensured the familiarity of our participants with the restaurants available in the surrounding Greater Toronto Area (GTA). While signing up on the web-interface, each participant was asked to indicate her level of familiarity with the restaurants in GTA and also her (pre-COVID'19) eat–out frequency. Figure 1(a) shows that around 84% of users consider themselves to be *slightly familiar* to *extremely familiar*, leaving only 16% who were *not at all familiar*. Similarly, Figure 1(b) indicates that over 90% users eat-out at least once a month, leaving ≈7% who visit restaurants a couple of times a year.

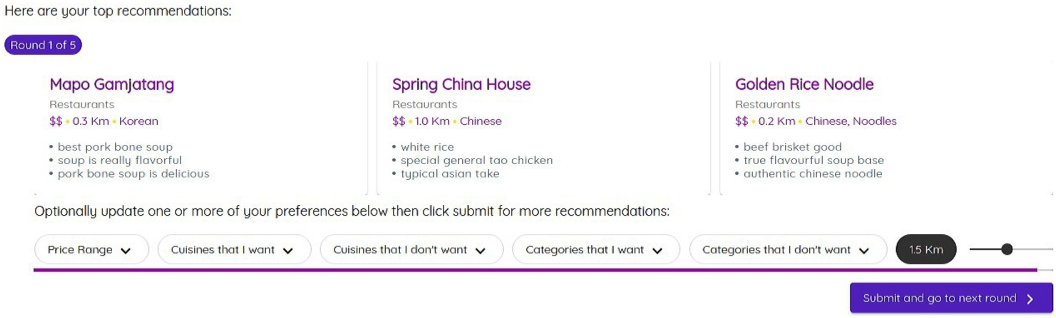
3.4 User study protocol

We recorded a compulsory 90 second instruction video to familiarize our participants with the interface functionalities before they sign up. Also, on-screen instructions were given at every stage of the study.

We adapted this protocol for critiquing from [32]. Each participant in our experiment was asked to first create a profile by selecting 10 restaurants that she may like. The user profile captures a user's long-term preferences. Once the profile was created, we asked the participant to use the recommender system (blindly assigned) to find a restaurant suitable for a specific fictitious scenario, e.g., "driving near Toronto Downtown, Queen and Bay (core downtown near city hall and the financial district), and would like to have a celebration dinner with relatives including children". A scenario in our experiment was a combination of a specific location in the GTA and a purpose of visiting a restaurant. Overall, we chose 6 different GTA locations (such that for each location there are nearly an equal number of restaurants available in our dataset) and 6 different purposes (see Table 1). We randomly pick one of the 6 locations and one of the 6 purposes to form a scenario. Each participant was randomly assigned such a combination as her scenario. While assigning scenarios to the participants, we again make sure that their distribution remains nearly the same among the participants of both systems.

Each user trial comprises a conversation of 5 rounds with the user randomly assigned to either *NP-Rec* or *kNN*. As shown in Figure 2, the participant was shown 3 restaurants in each round. Then

[3] University of Toronto REB-approved Ethics Protocol #00039634 titled "Evaluation of Conversational Recommender Systems".



Here are your top recommendations:

Round 1 of 5

Mapo Gamjatang
Restaurants
$$ • 0.3 Km • Korean
- best pork bone soup
- soup is really flavorful
- pork bone soup is delicious

Spring China House
Restaurants
$$ • 1.0 Km • Chinese
- white rice
- special general tao chicken
- typical asian take

Golden Rice Noodle
Restaurants
$$ • 0.2 Km • Chinese, Noodles
- beef brisket good
- true flavourful soup base
- authentic chinese noodle

Optionally update one or more of your preferences below then click submit for more recommendations:

Price Range
Cuisines that I want
Cuisines that I don't want
Categories that I want
Categories that I don't want
1.5 Km
Submit and go to next round

Fig. 2. A screenshot showing top-3 recommendations for the first round of conversation. User can provide her preferences through unit (one-by-one) or compound (all-together) critiquing for *price, cuisines that she wants, cuisines that she doesn't want, categories that she wants, categories that she doesn't want,* and *distance.* By default distance is set as at most 1.5km.

she was asked to optionally adjust six preference criteria (price, cuisine wanted and not wanted, category wanted and not wanted, and distance) to meet her preferences. For all but distance, there were drop-down lists with available options, a user could select one or more or none of them. Preferences are applied disjunctively while dispreferences are applied conjunctively. Critiques were used to constrain the 3 restaurants recommended in the next round. If the system found less than 3 matching preferences, an alert was shown to the participant to relax her constraints.[4] We require every participant to run the system for a full 5 rounds, so that the dialog has a length of five, even if she sees a restaurant earlier that she thinks is ideal. The advantage of this is that every participant's responses are based on the same number of restaurants on the screen (except the dialogs with failures), which makes for fair comparisons. We think this outweighs the possible disadvantage that, if a user has seen a 'perfect' item, she must nevertheless continue with the dialog, presumably receiving sub-optimal recommendations until she has completed 5 rounds, which may negatively affect her opinion of the system.

After the 5th round, the screen displayed the entire conversation (ideally) containing 15 recommended restaurants. The participant was asked to select one of the 15 restaurants — the one she thinks best suits to her scenario. Then she was asked a set of 6 survey questions (see section 4.4.2). Finally, the participant was shown a questionnaire to provide her qualitative feedback on the overall experience.

3.5 Evaluation Measures

In addition to measuring user effort involved in the dialog and soliciting users' subjective perceptions of their overall experience, we especially compute objective measures of the recommendation algorithms' behavior with respect to 'beyond-accuracy' measures: *diversity*, *surprise*, and *novelty*.

In each round, for each user u, we generate a list of top-n $(= 3)$ recommendations, R_u. We evaluate this list using the 'beyond-accuracy' measures stated above as an average of all users in the online study (denoted $\mathbb{U}_T$) using definitions given in Section 7 of [15]. We briefly describe these metrics as follows.

Diversity. This measures the diversity of the recommendation list R_u as the average pairwise distance among its elements. In content-based settings, we calculate the distance between two

[4] we describe *alerts* and *critiquing failure* in detail in section 4.1.

items (i, j) as the complement of their Jaccard similarity computed on their features $sim(F_i, F_j)$.

$$\frac{1}{|\mathbb{U}_T|} \sum_{u \in \mathbb{U}_T} \frac{1}{|R_u|(|R_u| - 1)} \sum_{i \in R_u} \sum_{j \in R_u \setminus i} 1 - sim(F_i, F_j) \tag{1}$$

Surprise. This measures the surprise of a recommended item as the minimum distance between the item and items in the user's profile P_u. This is averaged over the recommended items $i \in R_u$.

$$\frac{1}{|\mathbb{U}_T|} \sum_{u \in \mathbb{U}_T} \frac{1}{|R_u|} \sum_{i \in R_u} \min_{j \in P_u} 1 - sim(F_i, F_j) \tag{2}$$

Novelty. This is based on the fraction of users in the dataset who rated the item i. The logarithm is used to emphasize the novelty of the most rare items.

$$\frac{1}{|\mathbb{U}_T|} \sum_{u \in \mathbb{U}_T} \frac{1}{novelty_{max} \cdot |R_u|} \sum_{i \in R_u} - \log_2 \frac{|u \in \mathbb{U},\ r(u, i) \neq 0|}{|\mathbb{U}|} \tag{3}$$

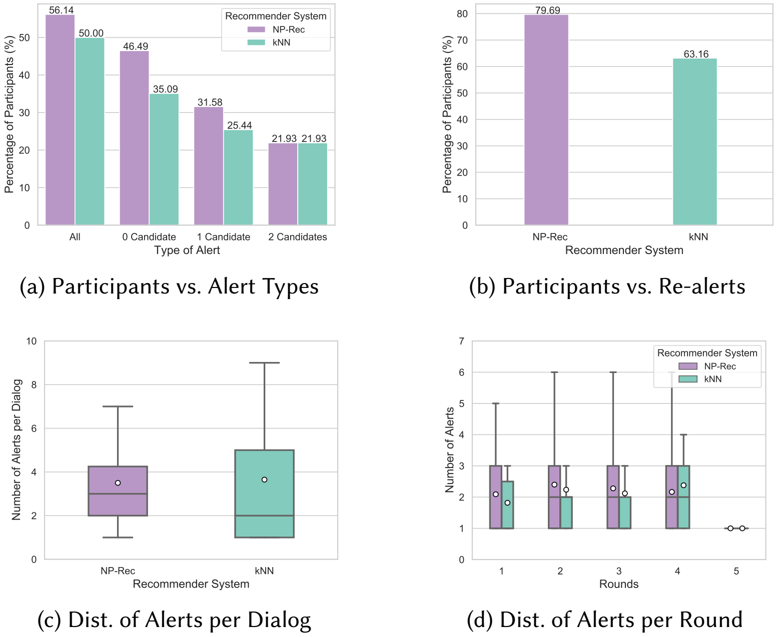
Here $novelty_{max} = -\log_2 \frac{1}{|\mathbb{U}_T|}$ is the maximum possible novelty value, which is used to normalize the novelty score of each individual item into $[0, 1]$.

4 EXPERIMENTAL EVALUATION

We now report and discuss the main results of our user study with the aim to understand the impact of the recommender choice on various aspects of user behavior in critiquing-based conversational recommender systems especially when dialogs involve *critiquing failures*. We assigned half the participants to the *NP-Rec* recommender and the other half to the *kNN*. In total, 228 users completed all parts of the trial to which they were assigned. This generated 1140 rounds of user interaction data (each user interacts with the system for 5 rounds) and the survey data for 228 users. We exploit this data to perform the following analyses.

(1) *Failure Analysis*: We analyze and compare *NP-Rec* and *kNN* dialogs on the basis of the number of users who were affected with critiquing failures, type and severity of such failures, distribution of failures over the dialogs, and at each round of the dialog.
(2) *User Effort Analysis*: We determine how much effort users of *NP-Rec* and *kNN* expend to finish the dialog. We do this using the effort involved in critiquing, the total time-taken, and the number of rounds needed to show the final choice to the user.
(3) *Flow Analysis*: This is a more fine-grained analysis of user behaviour over all five rounds of the dialog. We represent through Sankey diagrams the flow of the dialog in response to the user critiques. We also compare *NP-Rec* and *kNN* based on how users' dialog states were changed while interacting with the system to which they were assigned.
(4) *Analysis of Recommendation Quality*: At each round, the system returns the top-3 (or less in case of failures) recommendations. We analyze diversity, surprise, and novelty trends over the dialog rounds using the formulations that we described in section 3.5. We also asked survey questions regarding the relevance and beyond-accuracy measures of the recommended items. We analyze users' survey response to see how it relates to the objective analysis.
(5) *User Feedback Analysis*: At the end of the trial, each participant provided her qualitative feedback as a free-form text. We analyze such feedback to determine users' sentiment for the system they were assigned.

We discuss each of the above analyses in detail in the following subsections. It is noteworthy that we test significance of difference for each aspect using a one-sided t-test, with $p < 0.05$, with null hypothesis that *NP-Rec* needs less or equal effort to the *kNN*.



(a) Participants vs. Alert Types

(b) Participants vs. Re-alerts

(c) Dist. of Alerts per Dialog

(d) Dist. of Alerts per Round

Fig. 3. Details on Alerts

4.1 Failure Analysis

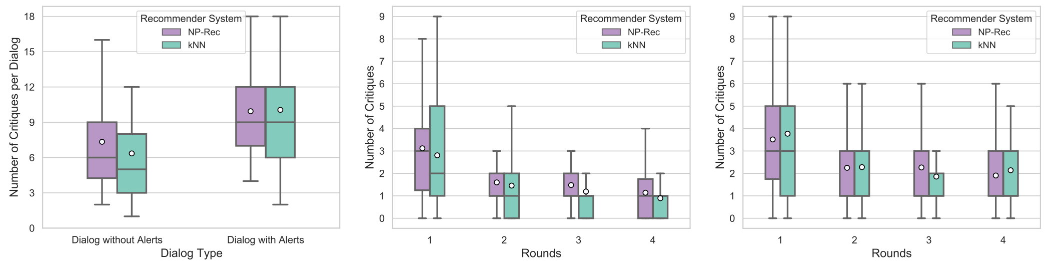
In our experiment, the system proposes 3 restaurants at each round of the dialog. The user reviews recommended items and provide her feedback in the form of critiquing, i.e. by optionally adjusting price, cuisine wanted and not wanted, category wanted and not wanted, and distance to find the restaurant that fits to her scenario. The system takes the user's feedback into account to generate next round of recommendations. More specifically, the user's preferences are applied disjunctively while dispreferences are applied conjunctively to find the next round of candidates.

Let $\mathbb{I}$ be the set of all candidate items that can be recommended to the user. At any round of the dialog r, user specifies her constraints in terms of her preferences C_r^+ and her dispreferences C_r^-. The set of consistent candidates for the dialog round $r + 1$ can be obtained as:

$$\mathbb{I}_{r+1} = \bigcup_{c \in C_r^+} \mathbb{I}_c - \bigcap_{c^{‘} \in C_r^-} \mathbb{I}_{c^{‘}} \tag{4}$$

Here, $\mathbb{I}_c$ returns the set of candidates consistent with the constraint c.

At any given round (say $r + 1$), when the system is unable to retrieve any candidates consistent with the user critiques (i.e. $|\mathbb{I}_{r+1}| = 0$), such a state of the dialog is known as *Critiquing Failure* [23]. In our experiments, we define *critiquing failure* as when the system returns $0 \leq n < 3$ matching restaurants in response to user critiques (adapted from [23]). In these cases, it is visually apparent to the user that no additional results are available and hence the current critiques have exhausted available candidates. More specifically, in case of $|\mathbb{I}_{r+1}| = 0$, system cannot proceed to round $r + 1$ unless user relaxes her constraints, we call it a state of *failure*; while in the other two cases $|\mathbb{I}_{r+1}| \in \{1, 2\}$, though the system proceeds to the next round, it demands the user to relax her constraints, otherwise it would generate *failure* in the subsequent round of the dialog (in this case, at round $r + 2$). We refer these states as *Alert1* (i.e. $|\mathbb{I}_{r+1}| = 1$) and *Alert2* (i.e. $|\mathbb{I}_{r+1}| = 2$) respectively.



(a) Dist. of Critiques

(b) Critiques per Round (without Alerts)

(c) Critiques per Round (with Alerts)

Fig. 4. Details on Critiques

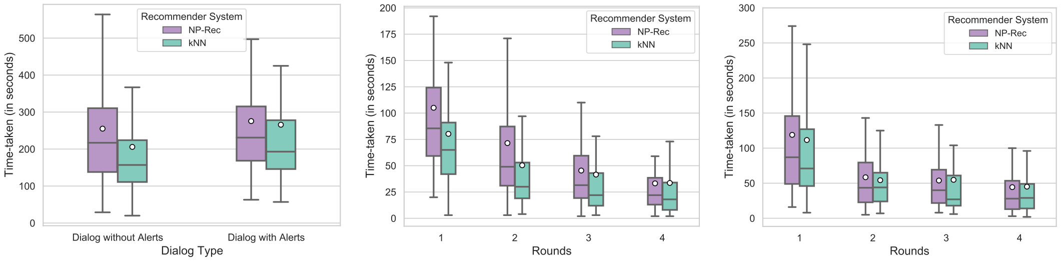
In these situations, users either accept a partially satisfying item or must otherwise revise their preferences. Hence, the fewer the failures (and/or alerts), the better the dialog quality.

In Figure 3(a), we observe that overall 56.14% of *NP-Rec* users encountered failures; while, it is 50% for *kNN*. Most of the failures occur when the user has no item recommended, with fewer failure cases where only one or two candidates were recommended. We find that 46.5% of *NP-Rec* users encountered failures with no item recommended, in case of *kNN* it is just 35%. This difference is statistically significant ($p = 0.039$). Figure 3(b) shows that nearly 80% of *NP-Rec* users encountered alerts and failures repeatedly in the same dialog, in case of *kNN* it is just 63%. This difference is again statistically significant ($p = 0.022$). There is clear evidence that *NP-Rec* and *kNN* lead to a different number of failures in a trial (Figure 3)(c). When we break out failures per round in Figure 3(d), there are clear failure trends. *NP-Rec* remains mostly unchanged, while *kNN* decreases up to round 3 and then increases in round 4. As we will see in the next subsections, most *kNN* users find their item of interest by round 3 and then start exploring further in the search of an even better option; they critique more, and hence, they encounter more failures. Here, we observe that *NP-Rec* leads to significantly more failures, more dialogs with repeated failures, and more round-wise failures.

4.2 User Effort Analysis

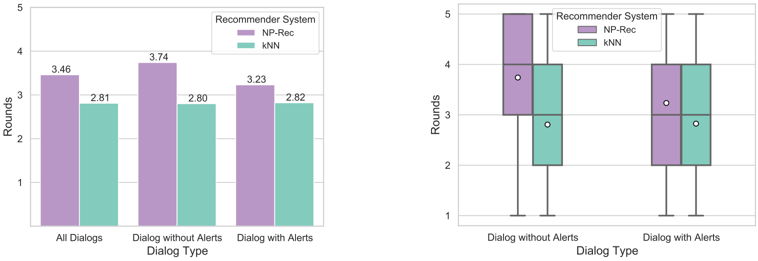
Now, we consider how much effort users expended to finish the 5 rounds of conversation. This includes (a) the number of critiques users have applied during their interaction with the system assigned; (b) the average task completion time; and (c) the average number of rounds needed in order for the final choice to be shown. We detail each of these one-by-one as below.

Critiquing effort: In Figure 4(a), we observe that users apply more critiques when they encountered failures (a.k.a. Dialog with Alerts) irrespective of the systems they are assigned. In case of dialog without failures, the number of critiques for *NP-Rec* is slightly higher than the *kNN*. When we break it into round-wise analysis, for dialogs without alerts (Figure 4(b)), the number of critiques for *NP-Rec* decreases after round 1 and then remains nearly the same for the rest of the dialog; however, in case of *kNN*, it decreases as dialog proceeds. In Figure 4(c), we observe that *NP-Rec* shows similar trend as for dialogs without alerts except that the number of critiques are higher in this case; while, for *kNN*, the number decreases up to round 3 and then increases in round 4. This is likely because most *kNN* users start exploring in round 4. Notably, the number of critiques when using the *NP-Rec* recommender is slightly higher compared to the *kNN* recommender over



(a) Dist. of Time-taken per Dialog (b) Time-taken per Round (without Alerts) (c) Time-taken per Round (with Alerts)

Fig. 5. Details on Time-taken



(a) Final Choice vs. Dialog Type) (b) Dist. of Final Choice

Fig. 6. Details on Final Choice

all rounds — we conjecture that users apply less critiques with the *kNN* recommender because it already implicitly captures many of their preferences. The difference of critiquing effort between *kNN* and *NP-Rec* is not statistically significant ($p = 0.28$).

Time taken: Figures 5(a) and 5(b, c) respectively show the overall time distribution of participants to complete a trial and the time spent per round of each trial without and with alerts. Overall we observe a distributional shift towards faster times for *kNN* users to complete a full trial (which is statistically significant for the dialogs without alerts, $p = 0.049$) and all rounds of a trial vs. *NP-Rec*.

However, the difference between the time taken by the two systems is not much in case of dialogs with alerts (see 5(c)). These results are generally consistent with the overall observations that *NP-Rec* users are critiquing more, encountering more critiquing failures, and thus overall expending more cognitive effort to express preferences as reflected in the time distributions.

Final choice: Figure 6(a) refers to the average number of rounds needed to reach the restaurant in their final choice. It is 3.5 rounds for *NP-Rec* vs. 2.8 for *kNN* (which is statistically significant, $p = 0.003$). It does not change with the dialog type: whether the dialog involves failure or not, it remains approximately equal.

This is further reflected in Figure 6(b), where we show the distribution of users over rounds in which they find their final choices. There is a clear distributional shift towards earlier rounds for *kNN* users vs. *NP-Rec*. Around 45% of *kNN* users find their final choice in the first two rounds

Table 2. Round-wise count of constraints for *NP-Rec* and *kNN* dialogs

Category	**Node /**	**Round-1**		**Round-2**		**Round-3**		**Round-4**	
	Constraint Name	*NP-Rec*	*kNN*	*NP-Rec*	*kNN*	*NP-Rec*	*kNN*	*NP-Rec*	*kNN*
Overall Constraint Count	No Failure	293	277	130	138	126	115	106	107
	Failure	25	24	46	37	43	37	47	49
	Alert1	0	0	19	9	16	13	9	12
	Alert2	0	0	7	7	8	10	13	12
Individual Constraint Count	`Price(+)`	72	73	22	27	21	17	18	8
	`Price(-)`	2	2	2	3	2	4	7	7
	`D-Cus(+)`	59	51	30	35	24	23	18	21
	`D-Cus(-)`	3	2	7	3	8	6	6	11
	`U-Cus(+)`	38	35	32	18	32	20	15	18
	`U-Cus(-)`	2	1	4	1	4	1	1	2
	`D-Cat(+)`	35	31	22	14	15	18	19	23
	`D-Cat(-)`	3	3	4	2	6	4	8	3
	`U-Cat(+)`	46	31	19	20	16	20	11	10
	`U-Cat(-)`	3	0	8	1	4	1	3	1
	`Dist(+)`	39	45	24	28	25	22	21	22
	`Dist(-)`	10	14	4	9	13	5	10	13
	`No Action`	6	13	24	30	23	34	40	41

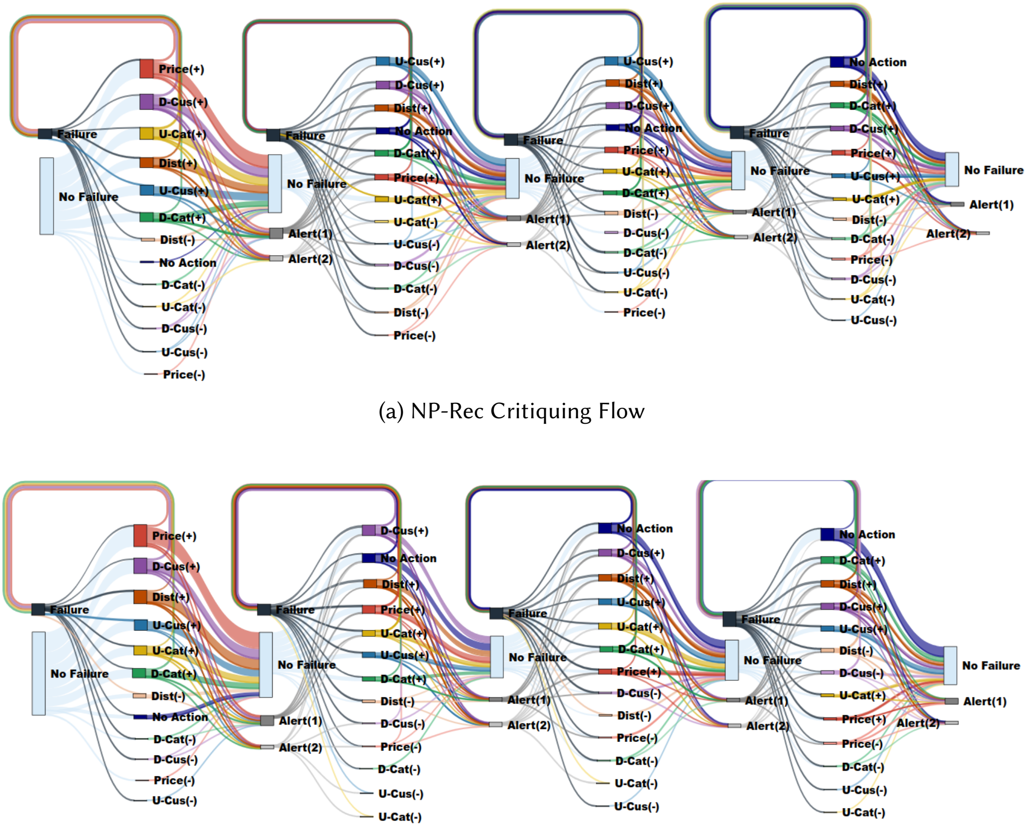
of the trial, whereas over 54% of *NP-Rec* users find theirs in the last two rounds. However, such differences are limited to dialogs without alerts; in case of dialogs with alerts, there is no clear evidence of the difference in the rounds of the final choice.

Overall, we find that *NP-Rec* users apply more critiques when dialogs involve no failures, they take significantly more time to finish the dialog, and find their item of interest in significantly later rounds of the dialog.

4.3 Flow Analysis

To better understand the results of the previous subsection, we now wish to undertake a more fine-grained analysis of the specific critiquing and user flow of *NP-Rec* and *kNN* users in our experiment. To do this, we present Sankey diagrams in the following sections.

4.3.1 Critiquing Flow Analysis. In Figure 7 that shows the workflow of the six actions taken by the participants during the 4 critiquing rounds of the trial (there were no critiques after the 5th and final round). Specifically, users could change the price or distance options, specify a cuisine wanted and not wanted, specify a category wanted and not wanted. In the diagrams, we have added prefix to action names with 'D-' for (desired) wanted and with 'U-' for (undesired) unwanted; similarly, we have also added suffix '(+)' to denote that the user has added more options to the associated critique feature and '(-)' for removing existing ones. It is noteworthy that in general adding options to unwanted cuisines and categories constrains a query; while, increasing the range of price and distance, and adding options to wanted cuisines and categories relaxes user's query for the next round. At any given round, when the user does not make any changes in the constraints, we show it as `No Action`. Further, as a result of the critique, the user could end up in a state of critiquing failure or non-failure. The fraction of time each type of critique (in conjunction with all other critiques made in the same round) led to a failure or non-failure state is shown as the width of the flow. At each round, loops from action nodes to the state of failure indicates that on applying such constraints, users moved back to the state of failure of that round.



(a) NP-Rec Critiquing Flow

(b) kNN Critiquing Flow)

Fig. 7. Sankey diagrams for Critiquing Flow Analysis

Overall, starting in the beginning of Round-1 with three initial recommendations (by definition in a non-failure state), we observe the following notable behaviors.

(1) For both the *NP-Rec* and *kNN* recommenders, we observe that the number of critiquing actions (the cumulative height of bars in each round) clearly decreases over trials. Table 2 shows that *NP-Rec* users exactly take 293 actions in total in Round-1 and they end up with 106 critiques actions in Round-4; in case of *kNN*, users apply 277 critiques in Round-1 and end up with 107 actions in Round-4.
(2) Looking at the individual constraints, we find that for both the recommenders, most users take '(+)' actions (i.e. add more options) on their preference criteria than to the '(-)' ones. Also, we see that as dialog proceeds, users prefer to apply `No Action` and keep their constraints unchanged.
(3) Regardless of the recommender, users clearly have a strong preference to initially critique price and cuisine vs. other critiquing options.
(4) As observed in the previous analysis, *NP-Rec* users have nearly the same number of critiquing failures in all rounds while *kNN* users have fewer initial round failures and relatively more failures in Round-4. When we look at the exact number of critique actions taken on the

failure state of each round (in Table 2), we find that for *NP-Rec*, it goes up from 25 in Round-1 to 46 in Round-2 and then remains nearly the same; while for *kNN*, it increases between Round-1 and Round-2 and then again between Round-3 and Round-4. We observe that at Round-4, *kNN* users again change their preferences for cuisines (e.g. *D-Cus(+), D-Cus(-)*), distance (e.g. *Dist(+), Dist(-)*, and category (e.g. *D-Cat(+)* among others which indicates that they want to explore in order to find even better option.

(5) We can see especially in Rounds 1–3 that the most likely actions to escape a failure state were to increase the price range, to increase the number of cuisine options, and to increase the distance.

(6) We observe that *kNN* users make fewer negative cuisine critiques in Rounds 2–3, potentially indicating that the *kNN* recommender better captured their cuisine dislikes without requiring explicit critiques. For example, *NP-Rec* users consistently tell the system what they do not want (e.g. `U-Cus(+)`, *U-Cat(+)*); while for *kNN* users, it is about the distance (i.e. `Dist(+)`. This is clearly visible in the individual counts part of the Table 2.

4.3.2 User Flow Analysis. Figure 8 that shows the transition of participants among various dialog states over the 5 rounds of the conversation. A transition occurs as a result of actions taken during the 4 critiquing rounds of the trial that we described in the last subsection. A user can be in one of the 4 dialog states: *Failure, No Failure, Alert(1),* and *Alert(2).* At Round-1, there are no *Alert(1)* and *Alert(2)*; similarly, at Round-5, there are no state of *Failure.* All other rounds have all the 4 dialog states.

At the beginning of Round-1, 114 users start the flow from *Non Failure* state of the dialog. The width of the flow is in proportion to the number of users moving from its *source* to its *target.* However, the exact values of the number of users transitioning between the *source* and *target* are shown in Table 3. Loops on the *Failure* states indicate that the critique actions taken by the users results in the repeated failures.

Overall, starting in the beginning of Round 1 with three initial recommendations (by definition in a non-failure state), we observe the following notable behaviors.

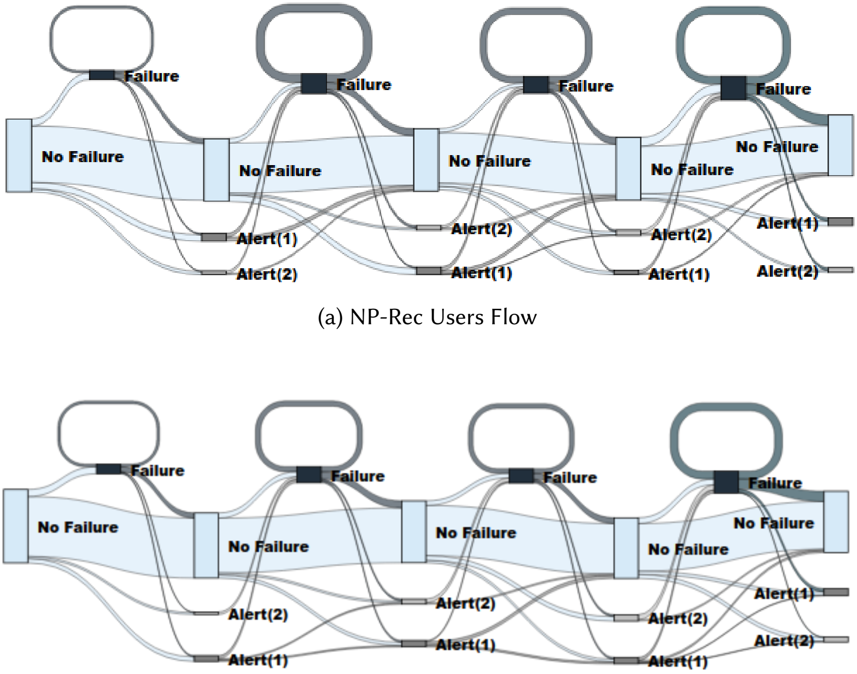
(1) For both the *NP-Rec* and *kNN* recommenders, we observe that as dialog proceeds, the number of users transitioning between the *No Failure* state of the current and the next rounds are decreasing. Specifically, for the first two rounds, more *kNN* users directly transit from and to the *No Failure* state than those of *NP-Rec*; while, the number remains nearly the same for the last two rounds of the dialog.

(2) As observed in previous analyses, for both the *NP-Rec* and *kNN* recommenders, less number of users encounter with *Alert(1)* and *Alert(2)* states than those who face *Failure.* The numbers are slightly less in initial rounds for *kNN* users while high or equal for the last round.

(3) Looking at the loops, we can clearly see that more *NP-Rec* users stuck in *Failure* loops for Rounds 2 and 3 than the *kNN* users. For the last round, this becomes equal. It is evident from our previous analysis that most *kNN* users found their item of interest by Round-3 and they apply critiques in Round-4 to further explore the catalog.

4.4 Analysis of Recommendation Quality

As stated earlier, our first aim is to understand the overall aggregate impact of returning a *NP-Rec* item consistent with current critiques vs. recommending top-ranked items consistent with current critiques according to a standard *kNN* recommender [17]. For this analysis, we present a variety of aspects of recommendation quality that we measured objectively and subjectively from users' interaction data and survey responses.



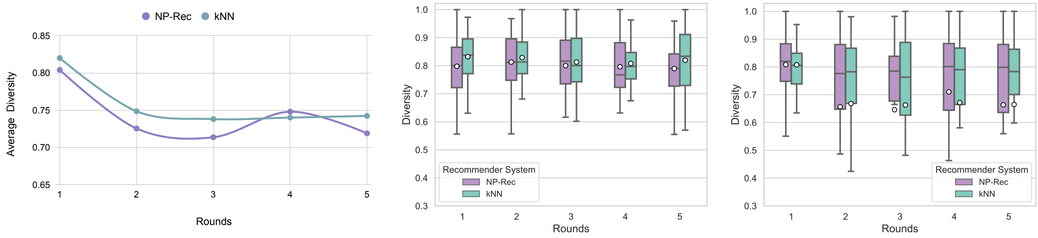
(a) NP-Rec Users Flow

(b) kNN Users Flow

Fig. 8. Sankey diagrams for User Flow Analysis

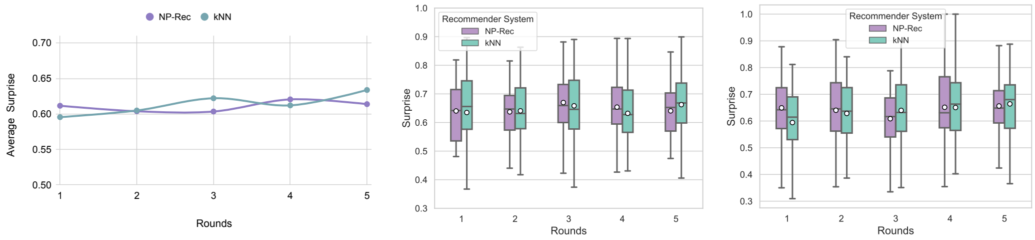
Table 3. System-wise and round-wise number of users moving from *source* to *target* as a result of their critiquing

Round	Source / Target	Same Round		Next Round					
		Failure		No Failure		Alert1		Alert2	
		NP-Rec	*kNN*	*NP-Rec*	*kNN*	*NP-Rec*	*kNN*	*NP-Rec*	*kNN*
1	No Failure	11	11	88	93	10	7	8	3
	Failure	4	4	8	8	2	2	1	1
	Alert1	0	0	0	0	0	0	0	0
	Alert2	0	0	0	0	0	0	0	0
2	No Failure	7	8	78	85	10	5	3	3
	Failure	13	8	11	11	2	3	4	3
	Alert1	6	5	6	0	0	2	2	2
	Alert2	4	4	2	0	0	0	0	0
3	No Failure	6	7	79	78	5	4	5	7
	Failure	10	7	11	9	2	3	3	3
	Alert1	6	2	5	5	0	3	0	0
	Alert2	4	6	3	2	0	0	0	0
4	No Failure	14	9	74	74	7	5	3	6
	Failure	12	12	16	16	5	5	4	1
	Alert1	5	5	2	3	0	1	1	1
	Alert2	6	8	3	2	0	0	0	0



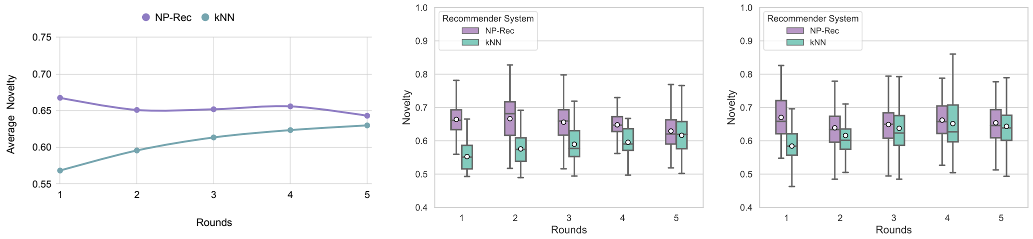
(a) Average Diversity over All Dialogs (b) Dialogs without Alerts (c) Dialogs with Alerts

Fig. 9. Details on Diversity Trends over Dialog Rounds



(a) Average Surprise over All Dialogs (b) Dialogs without Alerts (c) Dialogs with Alerts

Fig. 10. Details on Surprise Trends over Dialog Rounds



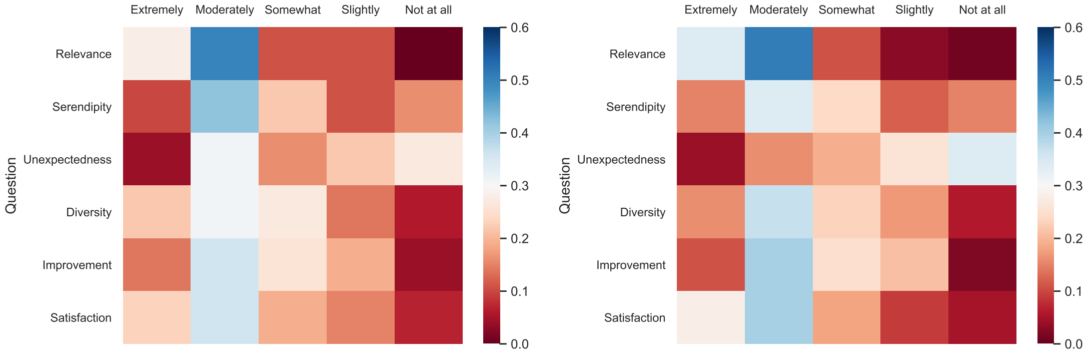
(a) Average Novelty over All Dialogs (b) Dialogs without Alerts (c) Dialogs with Alerts

Fig. 11. Details on Novelty Trends over Dialog Rounds

4.4.1 *Objective Analysis.*

We use evaluation measures mentioned in section 3.5 to specifically measure diversity, surprise, and novelty objectively for each of the 5 rounds of recommendations to see how these quantities vary over the dialog rounds on average over all users.

- *Diversity*: Figure 9 shows trends of diversity over dialog rounds. First, we observe from Figure 9(a) that diversity on average for *NP-Rec* users decreases up to round 3, increases for round 4, and again decreases for the last round. However, these fluctuations are minor. In case of *kNN*, diversity decreases up to round 2 and then remains unchanged for the rest of the dialog. When we split the dialogs into without (Figure 9(b)) and with alerts (Figure 9(c)), we see that diversity of dialogs without alerts remains nearly same over dialog rounds for both



(a) NP-Rec Users' Response Heatmap

(b) kNN Users' Response Heatmap

Fig. 12. Users' Responses to Survey Questions

the systems; while dialogs with alerts show the similar trends to all dialogs for *NP-Rec* and *kNN*. It reflects that the variations are due to the alerts and most likely show such a pattern due to the critiquing pattern that we described earlier in section 4.2.

- *Surprise*: In Figure 10(a),(b),(c), we observe that surprise remains somewhat unchanged over the dialog rounds regardless of the change of recommendation algorithm and the type of the dialog. This indicates that the users apply critiques in such a way that they do not let the system recommend items that are much different from their usual tastes.
- *Novelty*: Figures 11(b),(c) show that novelty of *NP-Rec* recommendations slightly decreases up to round 2 and remains nearly the same; for dialog; in case of *kNN*, for dialogs without alerts, novelty clearly increases as the dialog proceeds; while, for dialogs with alerts, it incrases up to round 3 and then remains the same for rest of the dialog.

4.4.2 Subjective Analysis. As stated before, we adopted a between-subjects trial: users interact with only one recommendation system to which they were assigned. For half the users, recommendations came from *NP-Rec*; for the other half of the users, they were from *kNN*. Users were completely unaware of the recommendation algorithm to which they were assigned.

As a user finishes her 5 rounds of interaction, we ask the user to select one of the 15 restaurants (ideally, with no failures), the one that she thinks the most appropriate to the scenario she was given. Then we ask her to answer the following questions:

- Relevance: How much do you think the <*selected restaurant name*> is appropriate to the <*scenario*>?
- Serendipity: Is <selected restaurant name> a pleasantly surprising recommendation?
- Unexpectedness: Is <selected restaurant name> different from your usual preferences?
- Diversity: Did you feel there was enough variety in the recommendations at each cycle?
- Effectiveness: On the whole, did the recommendations improve over the course of the interaction?
- Satisfaction: Would you be interested in using this system for finding restaurants in the future?

Their answers were on a 5-point: Not at all, Slightly, Somewhat, Moderately, Extremely. 228 participants completed the survey, 114 per system. Figure 12 summarizes users' responses for both the systems separately through *heatmap* plots.

- *Relevance question:* 78% of participants found *NP-Rec* recommendations to be *moderately* or *extremely* relevant, 11% found recommendations to be *somewhat* relevant, leaving 11% finding *NP-Rec* recommendations to be *slightly* or *not at all* relevant; in case of *kNN*, 85% of participants found recommendations to be *moderately* or *extremely* relevant, 11% found recommendations to be *somewhat* relevant, leaving 4% finding *kNN* recommendations to be *slightly* or *not at all* relevant.
- *Serendipity question:* 52% of participants found *NP-Rec* recommendations to be *moderately* or *extremely* pleasantly surprising, 22% found recommendations to be *somewhat* pleasantly, leaving 26% finding *NP-Rec* recommendations to be *slightly* or *not at all* pleasantly surprising; in case of *kNN*, 49% of participants found recommendations to be *moderately* or *extremely* pleasantly surprising, 24% found recommendations to be *somewhat* surprising, leaving 27% finding *kNN* recommendations to be *slightly* or *not at all* pleasantly surprising.
- *Unexpectedness question:* 55% of participants found *NP-Rec* recommendations to be *moderately* or *extremely* unexpected, 16% found recommendations to be *somewhat* unexpected, leaving 49% finding *NP-Rec* recommendations to be *slightly* or *not at all* unexpected; in case of *kNN*, 20% of participants found recommendations to be *moderately* or *extremely* unexpected, 19% found recommendations to be *somewhat* unexpected, leaving 60% finding *kNN* recommendations to be *slightly* or *not at all* unexpected.
- *Diversity question:* 53% of participants found *NP-Rec* recommendations to be *moderately* or *extremely* diverse, 27% found recommendations to be *somewhat* diverse, leaving 20% finding *NP-Rec* recommendations to be *slightly* or *not at all* diverse; in case of *kNN*, 53% of participants found recommendations to be *moderately* or *extremely* diverse, 24% found recommendations to be *somewhat* diverse, leaving 23% finding *kNN* recommendations to be *slightly* or *not at all* unexpected.
- *Effectiveness question:* 50% of participants found *NP-Rec* recommendations were *moderately* or *extremely* improved over the course of the dialog, 26% found recommendations were *somewhat* improved, leaving 24% finding *NP-Rec* recommendations were *slightly* or *not at all* improved; in case of *kNN*, 51% of participants found recommendations were *moderately* or *extremely* improved over the course of the dialog, 25% found recommendations were *somewhat* improved, leaving 24% finding *kNN* recommendations were *slightly* or *not at all* improved.
- *Satisfaction question:* 59% of participants found *NP-Rec* system that they would be *moderately* or *extremely* interested for using in the future, 19% participants were *somewhat* interested, leaving 22% finding *NP-Rec* system that they would be *slightly* or *not at all* interested in future use; in case of *kNN*, 68% of participants found the system that they would be *moderately* or *extremely* interested for using in the future, 18% participants were *somewhat* interested, leaving 14% finding *kNN* system that they would be *slightly* or *not at all* interested in future use.

On all but one criteria, *kNN* produced better recommendations. However, the difference was statistically significant only for the *Relevance* and *Satisfaction* questions. On the contrary, *NP-Rec* users found their final choices to be statistically significantly more unexpected than *kNN*. (We used a one-sided Z-test for proportions, with significance level $p < 0.05$. The null hypothesis was that those preferring *NP-Rec* are greater than or equal to those preferring *kNN*, ignoring those who were neutral i.e. who answered *Somewhat*.)

Table 4. Summary of annotations to survey responses

Category	*kNN* Recommender							*NP* − *Rec* Recommender						
	Annotator−1			Annotator−2			*kappa*	Annotator−1			Annotator−2			*kappa*
	+ve	*−ve*	*na*	*+ve*	*−ve*	*na*	*score*	*+ve*	*−ve*	*na*	*+ve*	*−ve*	*na*	*score*
Relevance of final choice	80	3	27	87	6	17	0.646	64	10	36	68	11	31	0.747
Consistency of constraints	17	8	85	22	7	81	0.885	9	11	90	11	14	85	0.790
Ease of use	20	11	79	21	11	78	0.918	14	6	89	16	5	89	0.857
Size of Cuisine & Category lists	13	16	81	10	18	84	0.843	9	23	78	3	25	82	0.741
Experience with failure alerts	0	8	102	0	8	102	0.865	0	11	99	0	11	99	1.000

4.5 User Feedback Analysis

At the end of the trial, we asked each participant to provide qualitative feedback based on their experience with the system to which they were assigned. Out of 228 users who finished all rounds of conversation, 220 provided their feedback.

To annotate user feedback, we first separated user comments for each of the two systems (totaling 110 comments per system), then classified along 5 different categories as shown in Table 4. We then asked two annotators to separately annotate the same user comments for the two systems, labeling each user comment as either *positive (+ve)* or *negative (-ve)* or *neutral (na)* based on the user's sentiment towards the category. If a user comment does not apply to a category, it is considered as *neutral* for that category. In Table 4, we also show *Cohen′s kappa score* for all categories and systems indicating substantial to near perfect inter-annotator agreement in each case.

We observe that:

- more *kNN* users find their final choice suitable to the scenario that was assigned to them than the *NP-Rec* users.
- more *kNN* users find recommendations to be more consistent with their critiques than the *NP-Rec* users,
- more *kNN* users find the system easy to use than the *NP-Rec* users,
- users were easily annoyed with the overwhelming size of cuisine and category lists — they have to find their preferred cuisines and categories from long lists,
- users disliked when they encounter failures repeatedly – they consider such failures as the system forcing them to compromise with their preferences.

We can see that more *NP-Rec* users complained about the last two cases than *kNN* users.

5 CONCLUSION

In this paper, we have undertaken a user study to evaluate the impact of recommender system choice (personalized via *kNN* vs. non-personalized via *NP-Rec*) on user critiquing behavior in a conversational recommender system, especially when dialog involves critiquing failures. While it should not be surprising that personalization reduces user critiquing burden, what we did find surprising is just how marked these performance differences were across a variety of analyses:

1. *Failure Analysis*: Participants using a *kNN* recommender encountered significantly less number of failures (i.e. $|\mathbb{I}_{r+1}| = 0$). The number of participants who faced repeated failures were also statistically significantly less than those who were assigned to *NP-Rec*.
2. *User Effort Analysis*: Participants using a *kNN* recommender critiqued less, finished faster, and found their final choice earlier. The results were statistically significant for the time-taken to finish the trial and the number of rounds needed to show the final choice to them.

(3) *Flow Analysis*: Participants using a *kNN* recommender spent less time making explicit critiques that were already implicitly captured by the recommender system. There were fewer participants stuck in failure loops and those who encountered repeated failures were able to “escape” in less number of attempts of taking critique actions.
(4) *Analysis of Recommendation Quality*: Participants using a *kNN* recommender found their recommendations to be more relevant, less unexpected, and still competent for diversity and surprise than those of *NP-Rec*. Overall, *kNN* users found greater improvement in recommendations on applying critiques over the rounds of dialogs and were more satisfied than the users of *NP-Rec*.
(5) *User Feedback Analysis*: Finally, analysing qualitative feedback for both systems show that more participants using a *kNN* recommender found their recommendations to be consistent with their critiques, suitable to their scenario, and the system easy to use (in comparison to those assigned to *NP-Rec*).

This empirically supports our key claim, which we believe to be under-emphasized in the existing literature, that the choice of recommendation system in a critiquing-based conversational recommender is critically important for the best overall user experience as outlined in (1)–(5) above.

We observed that critiquing failures and an overwhelming number of options for critiquing annoy users and force them to compromise with their preferences. In the future, we plan to leverage the recommender to preempt or suggest ways to recover from critiquing failures. We also want to explore additional objective measures that may predict the subjective characteristics we describe here.

Although we observed promising results, they are currently limited to a single domain and task. They are also limited to single configurations of each of the tested algorithms; alternative settings may result in somewhat different performance. Further experiments in other domains can examine the objective and subjective characteristics we have studied to see whether their relationships hold across domains and recommendation tasks.

Overall, the results of our user study highlight an imperative for further research on the integration of the two complementary components: *personalization* and *critiquing*, to achieve the best overall user experience in future critiquing-based conversational recommender systems. Further afield, it is not hard to imagine that these two components may be combined with a third component of large language models such as OpenAI’s ChatGPT [25] or Google’s PaLM [11] to facilitate personalized critiquing-based recommendation interactions through expressive natural language interfaces.

ACKNOWLEDGMENTS

We thank Arpit Jain for help with web design. The research in this paper was partially supported by funding from iNAGO Inc. and grants from the Ontario Centres of Excellence (OCE) and Mitacs.

REFERENCES

[1] Gediminas Adomavicius, Bamshad Mobasher, Francesco Ricci, and Alexander Tuzhilin. 2011. Context-aware recommender systems. *AI Magazine* 32, 3 (2011), 67–80.

[2] Derek Bridge, Mehmet H. Göker, Lorraine McGinty, and Barry Smyth. 2005. Case-based recommender systems. *Knowledge Engineering Review* 20, 3 (2005), 315–320.

[3] Robin Burke. 2000. Knowledge-based recommender systems. In *Encyclopedia of Library and Information Systems, vol.60, no.32.* Marcel Dekker, 180–200.

[4] Robin D. Burke, Kristian J. Hammond, and Benjamin C. Young. 1997. The FindMe approach to assisted browsing. *IEEE Expert: Intelligent Systems and Their Applications* 12, 4 (1997), 32–40.

[5] Li Chen and Pearl Pu. 2005. Trust building in recommender agents. In *Proceedings of the Workshop on Web Personalization, Recommender Systems and Intelligent User Interfaces at the 2nd International Conference on E-Business and*

Telecommunication Networks. Citeseer, 135–145.

[6] Li Chen and Pearl Pu. 2006. Evaluating critiquing-based recommender agents. In *AAAI*, Vol. 6. 157–162.

[7] Li Chen and Pearl Pu. 2007. The evaluation of a hybrid critiquing system with preference-based recommendations organization. In *Proceedings of the 2007 ACM conference on Recommender systems*. 169–172.

[8] Li Chen and Pearl Pu. 2007. Hybrid critiquing-based recommender systems. In *Proceedings of the 12th international conference on Intelligent user interfaces*. 22–31.

[9] Li Chen and Pearl Pu. 2009. Interaction design guidelines on critiquing-based recommender systems. *User Modeling and User-Adapted Interaction* 19, 3 (2009), 167.

[10] Li Chen and Pearl Pu. 2012. Critiquing-based recommenders: survey and emerging trends. *User Modeling and User-Adapted Interaction* 22, 1 (2012), 125–150.

[11] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. 2022. *arXiv preprint arXiv:2204.02311* (2022).

[12] Negar Hariri, Bamshad Mobasher, and Robin Burke. 2014. Context adaptation in interactive recommender systems. In *Procs. of the Eighth ACM Conference on Recommender Systems*. 41–48.

[13] Ming He, Jiwen Wang, Tianyu Ding, and Tong Shen. 2022. Conversation and recommendation: knowledge-enhanced personalized dialog system. *Knowledge and Information Systems* (2022), 1–19.

[14] Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2020. A Survey on Conversational Recommender Systems. *arXiv preprint arXiv:2004.00646* (2020).

[15] Marius Kaminskas and Derek Bridge. 2016. Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Transactions on Interactive Intelligent Systems* 7, 1 (2016), 2:1–2:42.

[16] Heeyoung Kim, Sunmi Jung, and Gihwan Ryu. 2020. A study on the restaurant recommendation service app based on AI chatbot using personalization information. *International Journal of Advanced Culture Technology* 8, 4 (2020), 263–270.

[17] Yehuda Koren and Robert M. Bell. 2011. Advances in Collaborative Filtering. In *Recommender Systems Handbook*, Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul B. Kantor (Eds.). Springer, 145–186.

[18] Kai Luo, Scott Sanner, Ga Wu, Hanze Li, and Hojin Yang. 2020. Latent Linear Critiquing for Conversational Recommender Systems. In *Proceedings of The Web Conference 2020*. 2535–2541.

[19] Kai Luo, Hojin Yang, Ga Wu, and Scott Sanner. 2020. Deep Critiquing for VAE-based Recommender Systems. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-20)*. Xi'an, China.

[20] Kevin McCarthy, James Reilly, Lorraine McGinty, and Barry Smyth. 2005. Experiments in dynamic critiquing. In *Proceedings of the 10th international conference on Intelligent user interfaces*. ACM, 175–182.

[21] David Mcsherry. 2005. Retrieval failure and recovery in recommender systems. *Artificial Intelligence Review* 24, 3-4 (2005), 319–338.

[22] David McSherry and David W Aha. 2006. Avoiding long and fruitless dialogues in critiquing. In *International Conference on Innovative Techniques and Applications of Artificial Intelligence*. Springer, 173–186.

[23] David McSherry and David W Aha. 2007. The Ins and Outs of Critiquing.. In *IJCAI*. 962–967.

[24] Preksha Nema, Alexandros Karatzoglou, and Filip Radlinski. 2021. Disentangling Preference Representations for Recommendation Critiquing with ß-VAE. In *CIKM '21: The 30th ACM International Conference on Information and Knowledge Management*. ACM, 1356–1365.

[25] TB OpenAI. 2022. Chatgpt: Optimizing language models for dialogue. *OpenAI* (2022).

[26] Roberto Pagano, Paolo Cremonesi, Martha Larson, Balázs Hidasi, Domonkos Tikk, Alexandros Karatzoglou, and Massimo Quadrana. 2016. The contextual turn: From context-aware to context-driven recommender systems. In *Procs. of the Tenth ACM Conference on Recommender Systems*. 249–252.

[27] Umberto Panniello, Alexander Tuzhilin, Michele Gorgoglione, Cosimo Palmisano, and Anto Pedone. 2009. Experimental comparison of pre-vs. post-filtering approaches in context-aware recommender systems. In *Proceedings of the third ACM conference on Recommender systems*. 265–268.

[28] Pearl Pu and Li Chen. 2005. Integrating tradeoff support in product search tools for e-commerce sites. In *Proceedings of the 6th ACM conference on Electronic commerce*. 269–278.

[29] Pearl Pu and Li Chen. 2008. User-involved preference elicitation for product search and recommender systems. *AI Magazine* 29, 4 (2008), 93–103.

[30] Pearl Pu, Li Chen, and Pratyush Kumar. 2008. Evaluating product search and recommender systems for E-commerce environments. *Electronic Commerce Research* 8, 1-2 (2008), 1–27.

[31] Pearl Huan Z Pu and Pratyush Kumar. 2004. Evaluating example-based search tools. In *Procs. of the Fifth ACM Conference on Electronic Commerce*. 208–217.

[32] Arpit Rana and Derek Bridge. 2020. Navigation-by-Preference: A New Conversational Recommender with Preference-Based Feedback. In *Proceedings of the 25th International Conference on Intelligent User Interfaces*. ACM, 155–165.

[33] James Reilly, Kevin McCarthy, Lorraine McGinty, and Barry Smyth. 2004. Dynamic critiquing. In *European Conference on Case-Based Reasoning*. Springer, 763–777.

[34] James Reilly, Jiyong Zhang, Lorraine McGinty, Pearl Pu, and Barry Smyth. 2007. Evaluating compound critiquing recommenders: a real-user study. In *Proceedings of the 8th ACM conference on Electronic commerce*. 114–123.

[35] Chong Eun Rhee and Junho Choi. 2020. Effects of personalization and social role in voice shopping: An experimental study on product recommendation by a conversational voice agent. *Computers in Human Behavior* 109 (2020), 106359.

[36] Hideo Shimazu. 2002. ExpertClerk: A conversational case-cased reasoning tool for developing salesclerk agents in e-commerce webshops. *Artificial Intelligence Review* 18, 3-4 (2002), 223–244.

[37] Barry Smyth. 2007. Case-based recommendation. In *The Adaptive Web*, Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl (Eds.). Springer-Verlag, 342–376.

[38] Barry Smyth and Lorraine McGinty. 2003. An analysis of feedback strategies in conversational recommenders. In *Procs. of the Fourteenth Irish Artificial Intelligence and Cognitive Science Conference*.

[39] Taavi T. Taijala, Martijn C. Willemsen, and Joseph A. Konstan. 2018. MovieExplorer: Building an Interactive Exploration Tool from Ratings and Latent Taste Spaces. In *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*. ACM, 1383–1392.

[40] Jesse Vig, Shilad Sen, and John Riedl. 2011. Navigating the tag genome. In *Procs. of the 16th International Conference on Intelligent User Interfaces*. 93–102.

[41] Hojin Yang, Tianshu Shen, and Scott Sanner. 2021. Bayesian Critiquing with Keyphrase Activation Vectors for VAE-based Recommender Systems. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR-21)*. Online.

[42] Jiyong Zhang and Pearl Pu. 2006. A comparative study of compound critique generation in conversational recommender systems. In *International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems*. Springer, 234–243.