Interacting with Recommenders—Overview and Research Directions

MICHAEL JUGOVAC and DIETMAR JANNACH, TU Dortmund

Automated recommendations have become a ubiquitous part of today's online user experience. These systems point us to additional items to purchase in online shops, they make suggestions to us on movies to watch, or recommend us people to connect with on social websites. In many of today's applications, however, the only way for users to interact with the system is to inspect the recommended items. Often, no mechanisms are implemented for users to give the system feedback on the recommendations or to explicitly specify preferences, which can limit the potential overall value of the system for its users.

Academic research in recommender systems is largely focused on algorithmic approaches for item selection and ranking. Nonetheless, over the years a variety of proposals were made on how to design more interactive recommenders. This work provides a comprehensive overview on the existing literature on user interaction aspects in recommender systems. We cover existing approaches for preference elicitation and result presentation, as well as proposals that consider recommendation as an interactive process. Throughout the work, we furthermore discuss examples of real-world systems and outline possible directions for future works.

CCS Concepts: • **Information systems → Recommender systems**; • **Human-centered computing → Human computer interaction (HCI)**; • **General and reference → Surveys and overviews**;

Additional Key Words and Phrases: User interfaces, interactive recommender systems, literature survey

ACM Reference format:
Michael Jugovac and Dietmar Jannach. 2017. Interacting with Recommenders—Overview and Research Directions. *ACM Trans. Interact. Intell. Syst.* 7, 3, Article 10 (September 2017), 46 pages.
https://doi.org/10.1145/3001837

1 INTRODUCTION

Automated recommendations have become an integral part of the user experience of many modern websites. The typical purpose of such recommendation components is to actively propose items of interest to the user, usually in the form of ordered lists. Thereby, they, for example, support users in exploring the space of available options or serve as a filtering component in situations of information overload.

Due to their high practical relevance, recommender systems (RS) are a topic of research in different fields. In the computer science literature, the focus of researchers is often to algorithmically determine which items should be placed in the recommendation lists of different users, possibly considering their particular contextual situation. A main assumption in many of these research works is that the preferences of the user are already given as an input. Assumptions about how users interact with the system are seldom made in such works. The minimal implicit assumption

The reviewing of this article was managed by associate editor Bamshad Mobasher.
Authors' addresses: M. Jugovac and D. Jannach, Department of Computer Science, TU Dortmund, Otto-Hahn-Str. 12, 44227 Dortmund, Germany; emails: {michael.jugovac, dietmar.jannach}@tu-dortmund.de.

https://doi.org/10.1145/3001837

is perhaps that the interaction between a recommendation component and the user consists of the presentation of the computed item list. Clicking on an item in the list will then transfer the user to another page that, e.g., provides more details about the selected item.

In many real-world recommender applications the set of possible user actions is indeed limited to the selection of one of the presented items in case they seem relevant. The users' preferences are often indirectly estimated by observing their behavior over time. While such an approach has the advantage of low complexity for the end user, it can also limit the value of a recommendation service in different ways. Users, for example, have no means to provide feedback on the presented recommendations in case they are outdated or based on a faulty interpretation of the user's actions. Moreover, presenting an ordered list of recommendations might not be the most suitable mechanism to support users in a decision-making problem, e.g., when purchasing a high-involvement product. Instead, side-by-side comparisons of items and the provision of further explanations might be more helpful for a user.

Implementing such functionalities requires more interactive and possibly more complex user interfaces, in which users can, for example, fine-tune their profiles and where the system has a richer repertoire of possible "conversational moves" [125]. In a number of real-world systems today, more advanced forms of user interactions can be found. On Amazon.com, for example, the system provides explanations for its recommendations and lets the user indicate whether certain items on which the recommendations are based should no longer be considered. Furthermore, for some product categories such as TV sets, domain-specific product advisors were deployed for some time on the site which interactively asked the user for their preferences or let the user explore the space of options by providing constraints on certain features. Other websites[1] offer similar "guided selling" systems for various domains that rely on knowledge-based advisory approaches as described in [53].

Generally, with richer user interaction models, more elaborate recommendation systems become possible, which can stimulate, accept, and process various types of user input. At the same time, the repertoire of actions is no longer limited to the one-shot presentation of recommendation lists, which can be insufficient in particular when the goal of the system is to offer decision support for the user. In this work, we provide an overview on existing approaches and techniques for improved human-recommender interaction from the literature. Throughout the article, we will identify potential research gaps and outline perspectives for future works, which will allow us to build more interactive recommender systems and thereby broaden the scope of recommendation technology.

1.1 Research Framework

In [4] and in many other research works, the recommendation problem is considered to be the problem of determining a function that computes a personalized relevance score for each item for each individual user. The goal of the corresponding algorithms is most often to optimize an abstract and domain-independent quality measure such as the Root Mean Squared Error.

Components of user interaction are not covered by such an algorithm-centric definition at all, even when we consider more recent works that aim at optimizing recommendation list characteristics such as diversity or novelty. In our work, we therefore adopt an interaction-oriented perspective of the recommendation problem. The framework of our research is sketched in Figure 1, which also provides a preview of our categorization of the different topics.

Preference Elicitation. The central task of a recommender system is to filter and rank the available items for the individual user. To be able to personalize the recommendations, many systems ask

[1] See, e.g., https://smartassistant.com or http://www.myproductadvisor.com.

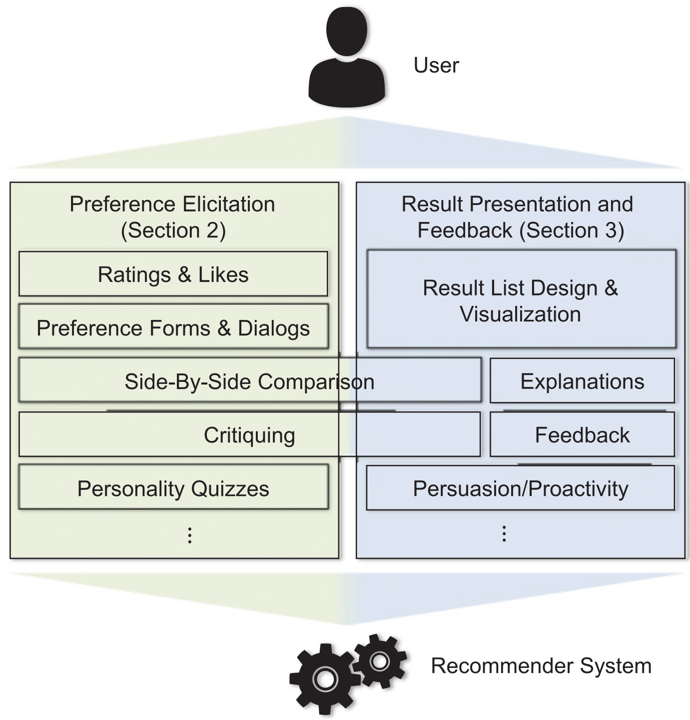


Fig. 1. Overview of the research framework of this article including the main interaction mechanisms between recommender systems and their users.

the users to explicitly state their preferences. This preference elicitation phase constitutes the first part of our research framework shown on the left side of Figure 1. Various types of user interactions are possible and will be discussed in this article. Traditionally, users are asked to *rate* a set of items on a predefined scale. Over time, however, various additional elicitation approaches were used in the literature. Some of them aim to reduce the cognitive effort for the user, e.g., by limiting the feedback scale to "thumbs up/down" or positive-only "like" statements. Others let the users choose between two items instead of asking them to rank several of them implicitly through the ratings. On the other hand, more complex ways of acquiring the preferences exist. Examples include multi-element *forms*, into which users can enter their constraints of preferences, and *dialog*-based systems, which guide users to their desired item in a conversational manner. *Critiquing* approaches, finally, represent a special form of preference acquisition as they allow the user to explore the item space when they refine their requirements. In addition to these established interaction mechanisms, the first part of our article covers a variety of less conventional preference acquisition methods, such as personality quizzes or picture-based approaches.

Interacting During Result Presentation. Once the user's tastes and preferences are known, the system is able to present the user with recommendations. The result presentation phase represents the second part of our research framework, shown on the right side of Figure 1. A number of proposals for designing the user interface and for possible user interactions in this phase were made in the literature. Different aspects can, for example, be considered in the design of the recommendation list itself, e.g., regarding the visual layout or the content features to be displayed. Furthermore, many academic and real-life systems additionally present *explanations* alongside the recommendations, which can be used to justify the system's item selection or to achieve a higher level of persuasion. In fact, when users understand the recommendation logic, they can even be empowered to correct the system's proposals. Besides these various forms of interaction in the result presentation phase, we will discuss off-mainstream research works, e.g., on *proactive recommendations*, in the second part of the article.

1.2 Scope and Outline

1.2.1 Scope. Our survey is focused on user interaction patterns for recommender systems and based on a structured review of the recent research literature. We systematically examined the proceedings of relevant computer science conferences of the last 10 years as well as journals from the field of information systems and computer science. Additional works were identified through keyword-based searches in digital libraries and Google Scholar.

Clearly, *what* is being presented in a recommendation list affects the user experience. The diversity of the recommendations, for example, can help users find new items or make it easier for them to understand which alternatives exist. To achieve these goals, a number of algorithmic approaches to create diverse item lists have been proposed in the literature. These works that concentrate on content selection aspects are not, however, in the focus of our work. Relevant questions in our context would rather be *when* to present such a recommendation list or—at least to some extent—how long such a list should be.

Furthermore, questions of general user interface design guidelines, response times, or aesthetic aspects are also not the key topic of our work. The general usability of more interactive recommendation approaches and the corresponding cognitive load for the user are, however, important aspects to be considered when richer interactions are supported by the system.

Finally, more and more recommender applications are nowadays used on mobile devices or in environments with limited interaction capabilities like TV sets. Clearly, the design of the user interface—and of the implemented interaction mechanisms in general—should take the capabilities and possible limitations of the end user's device into account. However, the research framework used in this work is independent of these constraints and organized along the phases and tasks of the recommendation process (e.g., preference elicitation, item presentation, explanation). The consideration of special device capabilities and limitations represents an orthogonal problem and many of these aspects like limited screen sizes are not specific to recommender applications. We do not, therefore, discuss interactions with mobile recommenders in a separate section, and refer the reader to corresponding survey papers such as [64] or [160].

Another relevant piece of related work is the recent survey on interactive recommenders by He et al. [79]. Their survey is complementary to ours as they focus mostly on *information visualization* approaches and discuss similarities and differences between existing interactive approaches. Also, they mainly concentrate on the result presentation and refinement phase.

1.2.2 Outline. We structure our review according to the described research framework into two major parts as shown in Table 1.

— In Section 2, we discuss the various ways of how a recommendation system can interactively acquire the preferences of a user. We will first review aspects and problems of explicit item ratings and then review methods for interactive preference elicitation and alternative forms of identifying the needs of a user.
— Section 3 is then devoted to user interaction mechanisms that are relevant once the initial user preferences are known. We discuss basic design alternatives for presenting the recommendations, review approaches for visualizing and navigating the space of available options, and finally focus on explanation interfaces and ways of putting the users into control of what is being recommended.

Throughout the article, we discuss open challenges and possible avenues for future research.

Table 1. Structural Outline of the Article

Section 2. User-preference-elicitation	
2.1. Explicit-item-ratings	Feedback Scale Design, Reliability and Biases of Ratings, New Users, Rating Support, Multi-Criteria Ratings, Reviews
2.2. Forms and Dialogs	Static Forms, Conversational RS
2.3. Comparison-based Techniques	AHP, Critiquing
2.4. Alternative Techniques	Personality Quizzes, Item Set Comparison, Tags, Pictures, Landscapes
Section 3. Result Presentation and Feedback on Recommendations	
3.1. List Design	List Design in Practice, Choice Set Size, Multiple Lists, Item Ordering and Organization
3.2. Visualizations Approaches	Highlighting and Item Presentation, Diagrams, Graphs, 2D, 3D, Maps
3.3. Explanation User Interfaces	Labels as Explanations, Knowledge-based Explanations, Explanations for CB/CF, Interactive Explanations
3.4. Feedback & User Control	Gathering and Incorporating Feedback, Recommendation Adjustments, Strategy Selection and Manipulation
3.5. Persuasive User Interfaces	Persuasive Explanations, Persuasive Item Selection and Presentation
3.6. Proactive Recommendations	When to Recommend

2 USER PREFERENCE ELICITATION

In this part of the article, we focus on interaction mechanisms to acquire *explicit* and *purposeful* statements from the users of a system about their preferences and interests. In practice, many recommendation systems leverage *implicit feedback* signals that are obtained by monitoring the user's behavior, e.g., their past website navigation or buying behavior. Such implicit feedback mechanisms, while often more important in practical applications than explicit item ratings, are not, however, in the scope of this section, in which we focus on approaches that utilize designated interaction mechanisms and UI elements for preference elicitation.[2]

Some of the UI mechanisms described in the following sections can be used for different purposes in an "interaction lifecycle." Star ratings, for example, are often used for the general acquisition of user tastes, but they can also be used as an instrument for users to give feedback on the provided recommendations (see Section 3.4). In this section, we focus on the initial or incremental elicitation of user preferences.

2.1 Explicit Item Ratings

Explicit, user-provided ratings are probably the most typical form of user preference information used in the recommender systems literature. Such ratings are available either given on a (discrete or continuous) numerical rating scale (e.g., one to five stars), binary (e.g., in terms of thumbs-up/thumbs-down feedback), or unary (e.g., when we only have positive feedback in terms of "like"

[2]Reviews on the use of implicit feedback signals for recommender systems can be found in [107, 143], or [96].

statements). The applicability and success of various recommendation strategies depend on the existence of a sufficient amount of such ratings. In the following sections, we will discuss approaches from the literature on how to design user interfaces for the rating elicitation process.

Note that in most practical applications the acquisition of rating-based preferences is temporally and visually disconnected from the recommendation process. Users of e-commerce and e-tourism sites are, for example, often asked to rate items a few days after a purchase. On other platforms such as media streaming or news portals, ratings can be provided immediately after the "consumption" of an item. A common problem in this context is that there is a tradeoff for users between the *immediate effort* of providing ratings and the *long-term reward* of better recommendations Swearingen and Sinha [216]. However, in many cases, it might not be immediately clear for the user that the ratings will be used to generate better recommendations. Furthermore, often a substantial fraction of the users is not willing to rate items at all. A typical goal of many research works – as will be discussed next – therefore aims to make the rating process as convenient as possible, to ensure that the user ratings are reliable, and in general to try to encourage users to provide more ratings.

2.1.1 Designing the Feedback Scale. An obvious first question to answer when designing the interface of a rating-based system is the level of feedback granularity. Typical rating scales are numerical, e.g., from 1 to 5 or from 1 to 10. Sometimes half-star increments are used and in some applications like in the Jester joke recommender [70] even more fine-grained scales—ranging from 1 to 100—can be found.

Such rating scales are common in a variety of research disciplines and, in particular, in survey-based approaches. The number of available options for the user is, however, only one of several design choices. Friedman and Amoo [60] discuss various aspects of ratings scales and their possibly undesired effects on the respondents. The design alternatives, for example, include the choice of the labels and the particular wording for the different grades, the extreme values, or if interval scales or ordinal scales are used.

In [10], the same authors explored the influence of different labels (e.g., 1 to 5) of the rating scale on the user rating behavior through a user study. Their results suggest that even when rating scales have the same number of options but a different range (e.g., 0 to 10 compared to −5 to 5), they do not evoke the same responses from the users because of the associated worse connotation for negative labels. Similarly, the study by Nowlis et al. [141] suggests that the existence of a neutral option can have an effect on the user's attitude toward certain items. On the other hand, Cosley et al. [32] found out that ratings from different scales, e.g., 5-stars compared to a binary scheme, can be correlated, but not always in an intuitive way. Their work additionally highlights that users prefer finer scales, most probably because they feel more in control.

A similar study has been done by Sparling and Sen [181], who report that users take more time to rate when shown finer scales and choose to rate fewer items because of the increased cognitive load. They also discovered that users need more time to rate an average item (in the middle of the scale) than a "good" item, which indicates decision support might be more important for those borderline items. In a related research, Swearingen and Sinha [216] found out that continuous sliders require less effort than traditional star ratings due to their "blurred boundaries" in situations where the users have to rate many items one after another. However, they report that in a general scenario users are more inclined toward the star rating scale. Additionally, their user study suggests that an upfront question about a favorite item, e.g., "*What is your favorite movie?*", which is not uncommon in real-life systems, can make users overly careful in fear of the consequences of a sub-optimal choice.

The level of detail of the feedback scale has also been the topic of one of the studies presented in [150], in which the authors compare eight different feedback styles, e.g., rating items, ordering the importance of item features, or giving feedback on an emotional scale. Among other insights, their study revealed that participants typically preferred feedback methods that required less effort. An exception was the emotion-based method, which the participants liked to some extent even though they found it rather difficult to use.

Besides the level of granularity, another question regarding the design of the elicitation interface relates to the used input mechanism. Different forms of clickable images of stars are common in desktop environments; such interfaces might, however, be too small and tedious to use in mobile scenarios. Wörndl et al. [197] investigated this problem and conducted a user study with a mobile rating elicitation prototype, where the participants could use different input mechanisms, including buttons as well as pinching or tilting gestures. Their results indicate that buttons performed best in terms of accuracy. Pinching gestures did not perform as well and tilting gestures were the worst option. The study also showed that the rating accuracy decreased when users were walking instead of sitting, which underlines the importance of taking the user's contextual situation into account when choosing a rating interface.

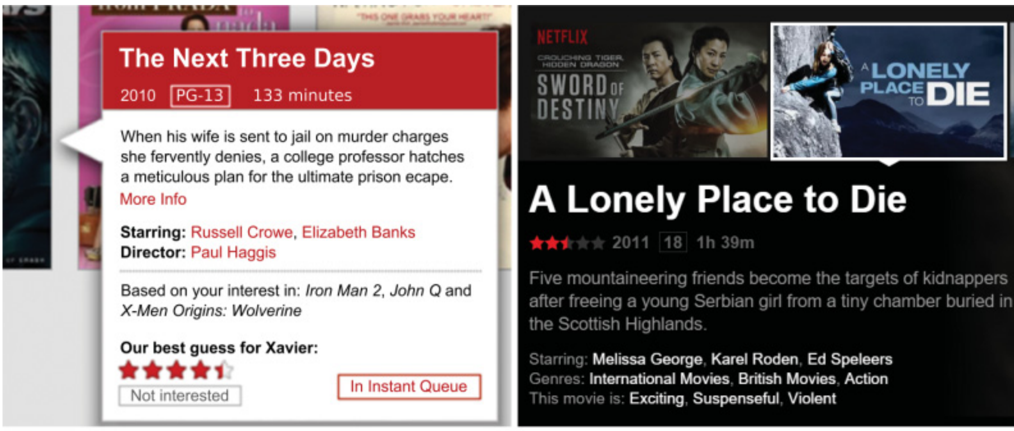
Overall, there exists quite a body of research on the design of rating scales and their associated labels. The various insights from other fields are, however, only considered to a marginal extent in the recommender systems literature and from a methodological standpoint some argue that it is not fully clear if it is correct to interpret answers on a numerical scale as interval ratings [214].

The particular research in the recommender systems field indicates that the different rating scales not only require different cognitive effort, but also that the reliability of the self-reported preference statements can be affected by the chosen rating scale, which will be discussed in the next section. A general shortcoming of the recommender systems literature is that researchers do not consider how the ratings for public rating datasets were actually acquired and if there might be biases in the data due to the way the data was collected.

2.1.2 The Reliability of Rating-Based Preferences and Bias Effects. A number of studies have investigated to what extent self-reported user preferences in the form of star ratings are actually reliable. Limited reliability can either be caused by a user's inability to state preferences accurately and consistently (over time), by a sub-optimal user interface for rating elicitation, or by the user's contextual situation, like in the already mentioned work by Wörndl et al. [197].

Kluver et al. [109], for example, measure the effectiveness of different rating scales by comparing the resulting prediction accuracy. Their observations are based on a dataset where users re-rated items after 6 weeks on different scales (see [32]). According to their findings, scales have to be chosen with the elicitation scenario in mind (e.g., using a mobile or a desktop computer) because of their tradeoff between information usefulness, user effort, and noise.

In a similar study, Amatriain et al. [7] also address the issue of noise in user-provided ratings but go one step further and evaluate if it would be beneficial in terms of accuracy if users would re-rate certain items. Their results indicate that rating the same item more than once can lead to substantial accuracy improvements. The consistency of the user's rating behavior over time was also in the focus of earlier studies. In contrast to [7], Cosley et al. [32], for example, observed that users rated items quite consistently after a period of 6 weeks. Nonetheless, there is some indication that it can be beneficial to ask users to re-rate items, e.g., because of the user's interest drift over time. However, none of the mentioned works so far discuss possible interaction mechanisms that would help to stimulate this user behavior without being annoying for users.



(a) Old version of the Netflix UI. Taken from [210]. (b) Current version of the Netflix UI.

Fig. 2. Screenshots of different versions of the Netflix UI. Both versions display a star rating. The old version shows the predicted rating for the user. In the current version, the meaning of the displayed rating is unclear.

In a different line of research, Bollen et al. [20] investigated if the time gap between item consumption and rating can lead to biases in the data and obtained mixed results. An analysis of existing rating data indicates that movies which were rated long after their release tend to receive higher ratings ("positivity effect").[3] In contrast, a user study indicated that the participants rated movies lower if the time when they actually watched them was longer ago ("negativity effect"). Overall, this suggests that there might be biases in existing rating datasets which are the result of how the ratings were acquired, e.g., immediately after consumption (e.g., on Netflix) or possibly disconnected from the consumption (as on the MovieLens platform[4]).

Additionally, even if the time of consumption is not an issue, users might still not rate items according to their "own enjoyment." Instead, star rating scales might tempt users to act like a critic and rate items according to their supposed "quality." An example would be a user who is bored while watching "Citizen Kane" but rates it 5 stars afterward, trying to assign an objective rating to help the community. McAlone [211] even reports that Netflix plans to give up their star rating system in favor of a more useful one to avoid this problem.

The work in [32] furthermore raises questions related to anchoring effects in the rating elicitation process. When the average of the user community ratings is displayed in the user interface, a measurable effect on the user's rating behavior can be detected. Similar results were reported in [2], where the presentation of rating predictions as done on the Netflix video streaming platform (as shown in Figure 2(a)) had an effect on the users' ratings. In that respect, the current version of the user interface of Netflix (Figure 2(b)) is not very clear because the displayed movie rating is not labeled as being a community average or predicted rating. These observations indicate that what is presented during the rating process should be carefully chosen. Displaying average community

[3]One additional reason for the effect not reported in the article can, however, also lie in a selection bias, i.e., that only comparably "good" old movies are contained in the rating dataset, whereas newer movies are in general of mixed quality.
[4]http://movielens.org.

ratings is a common feature on many online platforms and more work is therefore required to understand and prevent possible biases in the rating data.

In some real-world applications, we can actually observe such skewed rating distributions and users sometimes tend to only provide very positive ratings or only extreme ratings, i.e., they only use a smaller part of the available rating scale. One prominent example is the YouTube platform, which initially used 5-star ratings, but later on switched to binary ratings as they nearly only observed 5-star ratings and a few 1-star ratings Rajaraman [213]. This suggests that in practice it can be important to monitor the behavior of the user community to be able to appropriately re-evaluate and adapt the feedback scale.

One reason for the imbalance of the rating distributions can be that the rating scale might be socially or culturally "constructed," with ratings below 4 stars being rarely used. Wulff and Hardt [198] analyzed if there are such cultural or site-specific differences by looking at the rating distributions for the same domain on two websites: a Danish movie site and IMDb. They discovered very strong differences in the rating distributions. The Danish website users had a slight tendency to use the upper half of the rating spectrum but the ratings were still spread quite evenly. In contrast, on IMDb the highest rating value (10) is dominating and there are more 10-star ratings than the 9-star and 8-star ratings combined. The lowest rating option is the least chosen one on the Danish site. It is, however, quite often used in the IMDb data, which suggests that the fraction of openly negative users on that platform is higher. From their study it is not fully clear yet which factors contribute to the observed differences. In addition to cultural or site-specific aspects, the choice of the feedback scale could have an influence and more research is required to understand whether different elicitation interfaces could help to avoid these skewed distributions.

In the end, rating datasets that contain mostly very positive and a few very negative ratings can lead to biases when algorithms are benchmarked under the assumption that ratings are missing at random (MAR) [126]. In addition, the value for the end user might be limited when almost all items have a very high average community rating.

2.1.3 Dealing with New Users and Acquiring More Ratings. In research settings, the existence of a mandatory minimum amount of ratings per user and item is often assumed or users are asked to rate items before they can use the system. In most practical settings, such an approach to overcome the cold-start problem is infeasible and initial user profiles are usually constructed via implicit feedback signals or alternative ways for users to state their preferences.

In [15], for example, a TV recommendation system is proposed in which the user cold-start phase presents itself like a search system, i.e., users can enter query strings. After the first interactions, the system then incrementally develops an automated filtering and recommendation system. In earlier versions of the Amazon.com website, a landing page for their recommendation system was used where users were asked basic questions about their tastes in a conversational manner [216].

However, even if we ask users to rate items, it is not always clear how many item ratings are enough for making reliable recommendations. In addition, if we ask too few questions, users might be worried that the system will not be capable of estimating their tastes accurately enough [216]. The study by Drenner et al. [45]—even though done in an academic setting—indicates that asking as many as 15 questions can be acceptable for many users and only few users leave the site when asked to rate more than a dozen items. Along the same line of research, the studies of Cremonesi et al. [35] suggest that collecting between 5 and 20 ratings is an optimal range considering the tradeoff between user effort and recommendation quality. A similar observation that users are willing to trade their time for better recommendation accuracy was made in [93], where 85% of the

Fig. 3. One of the rating support systems by Nguyen et al. [140], which displays the item to rate and an item from the user's history with a Venn diagram to show the relation between the tags of the items. ©Nguyen et al.

visitors of a deployed advisory application stepped through a longer preference elicitation process before the recommendations were presented. Furthermore, the work of Drenner et al. [45] showed that users can easily be persuaded to accomplish a mandatory task before using the system, even if it only helps the community as a whole, e.g., by providing tags for some items.

In practical applications such mandatory tasks are, however, very uncommon and the initial user preferences are often acquired by letting users select from predefined interest categories. We will discuss such user interfaces in more depth in Section 2.2.

2.1.4 Helping the User to Rate. In case we decide to ask users to rate a *gauge set* of items, the question arises if the system should automatically select the items to rate for the user or if the user should be able to choose items freely. A corresponding study by McNee et al. [131] revealed a trade-off between putting users into control and decreasing their effort. When users were provided with item suggestions, they were more likely to finish the process; with self-selected items users were more satisfied and expressed a higher tendency to use the system in the future. A user interface that combined both approaches did not, however, lead to the measurable additional effects.

In order to reduce the load for users when rating items, Nguyen et al. [140] proposed a supporting user interface where users are shown an item to rate and an additional item from their past history that has some commonalities with the current item, e.g., a similar genre (see Figure 3). Their observations showed that the support interface led to less noise in the data and lower effort for the user compared to the baseline approach.

The problem of determining the most informative items that a user should rate is related to the *active learning* problem in machine learning. Factors that influence the system's decision for an optimal rating set may include the ratability (i.e., if the user is likely able to rate the item), the rating cost (i.e., how much effort the user has to invest in rating the item), or the saliency (i.e., how much effect the acquired rating will have on the quality of the subsequent recommendations) [163]. A number of research works on active learning exist in the recommender systems literature, including [11, 50, 101, 104, 155, 156], or [38]. Most of these works typically address one single optimality criterion, e.g., by minimizing the entropy in the model distribution. A very specific elicitation procedure was presented by Elahi et al. [48], who propose to ask the users questions about their personality before the rating elicitation phase. The underlying idea is to reduce the set of potentially relevant items based, for example, on the user's openness to new experiences.

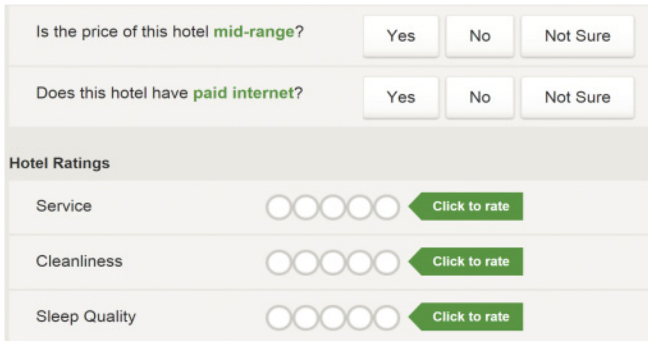


Fig. 4. User interface fragment for structured feedback on TripAdvisor.

Overall, most of these works are based on offline analyses or simulations and do not explicitly address questions of how to integrate the rating elicitation process in the user interface. Furthermore, if users are asked to rate many items which they do not know, they might either have a tendency to rate unknown items too low [95], provide noisy or even arbitrary ratings, or stop using the system as a whole.

2.1.5 Multi-Criteria Ratings, Structured and Unstructured Item Reviews. Most of the literature on recommender systems is based on the assumption that users can provide a single rating value according to their *overall assessment* for each item. In many practical systems, however, more fine-grained forms of providing explicit user input are available to the users. Different online platforms in the tourism domain, for example, allow the customers to rate hotels along different quality dimensions, such as staff friendliness, cleanliness, or value for money (see Figure 4).

In the RS research literature, a number of works exist that try to leverage these multi-criteria ratings for improved recommendations (see e.g., [3, 61, 134]). The findings of Jannach et al. [98], which are based on a hotel rating dataset, however, also indicate that incorporating certain criteria into the approach can actually lead to a *decrease* in the prediction accuracy. This indicates that some of the criteria ratings contain noise, which could be caused by the fact that some customers did not understand the meaning of a rating dimension or interpreted them incorrectly. Moreover, customers could have chosen some arbitrary values simply because they were overwhelmed by the many rating dimensions. In a related work, Rutledge et al. [164] looked at how participants rate the features of an item compared to its overall rating. They conclude that item features are generally rated higher than the items themselves, making it difficult to use compound feature ratings for item recommendation.

Overall, these observations suggest that more research is required regarding the design of the user interface for multi-criteria ratings systems. In some of the datasets used by Jannach et al. [98], more than one dozen rating dimensions were available to the users. However, as the obtained results indicate, it would have been better to let the user only rate a *relevant* subset of the item features to obtain less noisy user feedback. In this case, the relevance of the different criteria could, for example, be ascertained by means of feature selection techniques from the field of machine learning.

Besides multi-criteria ratings, platforms like Tripadvisor.com provide additional forms of *structured* feedback as shown in the user interface fragment in Figure 4. Limited works exist in the literature both with respect to how such additional feedback can be integrated into existing algorithms and with respect to the design of user interfaces which do not overwhelm users with their complexity.

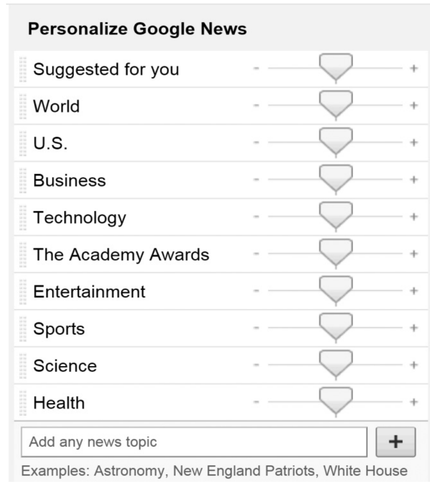


Fig. 5. User interface for Google News personalization.

Finally, a common feature of many online platforms (including TripAdvisor) is that users can provide free-form textual reviews. How to derive sentiment and rating information from such reviews that can be fed into recommendation algorithms has been explored in different research works during recent years [62, 88, 122].

2.2 Preference Elicitation Using Forms and Adaptive Dialogs

2.2.1 Static User Profile Forms. In practice, many recommendation and information filtering applications ask their users to fill out static forms about their preferences and general interests. Typical news aggregation applications, for example, let users select their interest categories such as "Entertainment," "Politics," or "Sports" and these settings are subsequently used to filter the incoming news stream in a personalized way. In some domains they are also used to acquire "indirect item ratings" by asking the users to name a few favorite artists or writers, e.g., in the context of music or book recommendation.

A more advanced and fine-grained form of putting the user into control is the user interface of Google News (as of 2015) as shown in Figure 5. Users of the site can specify their interest in different categories on a fine-grained level using slider controls. Furthermore, personal interest categories can be added using keywords. Individual categories can also be removed.

A user interface comparable to the Google News settings screen was used in the study by Knijnenburg et al. [111]. In their prototype system, users were also presented with an opportunity to assign weights to their "possible needs." The authors compared this interaction method, for example, with a system that recommends items based on the users' browsing behavior. Overall, the user study revealed no significant differences between the interaction methods, e.g., in terms of perceived control, trust, or satisfaction, even when compared to a non-personalized popularity-based strategy. However, the authors observed various significant differences for certain user subgroups. Users that had more domain knowledge, for example, experienced higher choice satisfaction when using the more complex system. In contrast to Google News, their system does not, however, remember the assigned weights in a long-term user profile.

Generally, static forms have the advantage that they are simple and intuitive to use for most online users. Nonetheless, even such static forms like the one in Figure 5 can become challenging to operate for users. It is, for example, unclear if setting all sliders to the highest value has the same effect as setting all sliders to the middle position. Furthermore, the very first entry in the list of categories in Figure 5 seems to have a different meaning than the other predefined entries.

Another limitation of form-based approaches is that users have to manually adapt the settings in case their interests change over time. Technically, such explicit profiles could be used in cold-start settings and then adapted or overwritten by a machine-learned user profile over time. Communicating such a complex mechanism to the users might, however, be challenging in particular as manually provided user settings are generally not automatically changed by today's systems. A possible strategy for a recommender system could be to use the user-provided preferences as (strict) interest filters and then to rank the items in each category based on a learning algorithm.

2.2.2 Conversational Recommender Systems. In some application domains of recommender systems determining the most relevant set of products requires that the system obtains input from the users about their specific *short-term* requirements and preferences. A hotel recommender system, for example, at least has to know the travel destination and the number of travelers. When recommending digital cameras, some technical specifications or the intended use of the camera have to be known.

The most simple form of acquiring the users' constraints is to provide a static fill-out form in which the users specify their requirements such as the maximum price, minimum camera resolution, or some other required product feature. Such "product finder" or item filtering forms can often be found on today's e-commerce websites and provide more or less advanced search functionality.

Such one-size-fits-all fill-out forms, however, have their limitations, which is why a number of approaches toward more *conversational recommendation systems* have been made in the literature. One of the possible problems of static forms is that in high-involvement, complex product domains not all users might be able to specify their requirements on a technical level, e.g., provide values for the desired shutter speed of a camera. Another typical problem of detailed search forms is that situations can easily arise in which no products remain that fulfill all customer requirements.

Conversational approaches to item recommendation (also termed "advisory systems") try to overcome such limitations in complex item domains. Instead of asking users to provide all requirements in one step, advisory systems guide the users through an interactive dialog. During this dialog, these systems typically ask the users about functional requirements rather than technical constraints. These requirements are then mapped through an internal logic to desired product characteristics. Using these internal rules and other explicitly encoded domain knowledge, these systems are capable of providing a variety of interaction types including the provision of additional hints and explanations or mechanisms to help the user recover from a situation in which no product satisfies all customer constraints.

Early examples of such conversational systems include the work of Linden et al. [123], who implemented an interactive travel planner, and the work by Göker and Thompson [69], who improve the idea of conversational travel planning by adapting the level of presentation detail and the filtering strategy to the user. Their system called *Adaptive Place Advisor* offers a comprehensive set of features including, e.g., the use of voice input and output as well as the construction of long-term user models.

In the research literature, a number of attempts have been made to improve upon various subtleties of the conversational process, e.g., to adapt it more to the individual user, to better deal with unfulfillable requests, or to reduce user effort in general. For example, Grasch et al. [73] also provided users with a speech-based interaction model and observed that users who employed this form of interaction reported higher satisfaction and needed less interaction cycles.

Other works such as [46, 100, 112, 121], and [200] focus on offering various automated adaptations on the level of navigation, content, and presentation. These adaptations can, for example, be

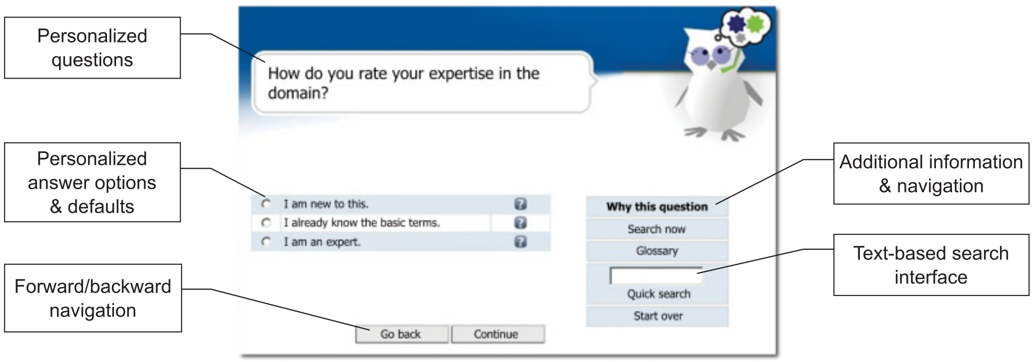


Fig. 6. Dialog page built with AdvisorSuite.

based on the users' current search goals, their navigational history, or as in the case of [111] on their domain expertise.

A specific problem of incremental filtering-based approaches is that sometimes no items are left because the user has made too restrictive choices. To resolve such an issue, McSherry [133] propose to guide the user to recovery via incremental relaxations. Similarly, McCarthy et al. [129] try to minimize the user effort by showing meaningful compound recovery options that combine multiple feature relaxations in one action. Finally, Chen [27] tries to avoid asking the user for possible recovery actions. Instead, in her approach the system generates a solution that it considers to be acceptable for the user and presents it alongside a corresponding explanation.

The presentation of complex recovery actions can generally lead to high cognitive load for the end user. The study by Felfernig and Gula [55] suggests that such recovery options are therefore rather suited for expert users and might even increase the user effort for novices. Additionally, their study suggests that to increase user satisfaction the final recommendations after a conversational session should be displayed as a comparison of two items.

Many of the works discussed so far focus mostly on specific details of the conversational process. There are, however, also research works that aim to develop more comprehensive and integrated frameworks for conversational recommendation. These approaches, for example, include the works by Göker and Thompson [69] or Ricci and Del Missier [161], who combine the core conversation functionality with adaptivity and recovery features.

Building such comprehensive and domain-dependent systems can be complex and tedious, and their development process should be at least partially automated. Bridge [23] therefore, for example, proposes a solution on a "dialog grammar" which can be used to define the dialog structure of a mixed-initiative system. These formal descriptions can then be used to automatically generate a conversational recommender. Other attempts have been made to improve upon this idea, e.g., by Kaindl et al. [103], who use OWL (Web Ontology Language) definitions to model the conversational process and even the visualization of the questions.

A comprehensive software tool for the automated development of conversational recommender systems called AdvisorSuite was presented in [53, 93, 94]. Their knowledge-based system implements a variety of user interactions, with the flow of the interactions and the dialog pages being automatically generated at runtime based on the contents of the knowledge base and the user's most recent actions (see Figure 6). The system also features user-specific explanations, which are used as a means for users to give feedback to the system about the importance of different recommendation rules. Additionally, the system provides automatically computed recovery options [91] and tools for the knowledge engineer to debug the dialog process [52].

A common shortcoming of such knowledge-intensive approaches lies in the effort that is required to set up and maintain the underlying knowledge base. An integrated software environment was developed by Felfernig et al. [54] which provides different visual editing tools for the knowledge engineer. A variety of real-world applications were reported to be built with AdvisorSuite and the authors claim that in many domains the knowledge engineering efforts remain manageable. However, a limitation of the system is that it does not implement any "learning" capacity, e.g., to update the recommendation rules over time based on the current product catalog or technical features of the items.

2.3 Item-Oriented and Comparison-Based Elicitation Techniques

An alternative to asking users directly about their specific preferences, e.g., about item features, before presenting recommendations is to follow an item- or product-oriented approach. The general idea is to determine the preferences of a user given certain reference items. Different approaches were proposed in the literature.

2.3.1 AHP. The analytic hierarchy process is a method for preference elicitation that was developed in the context of multi-criteria decision making in the 1970s, in particular for group decisions [165, 166]. The general idea is to hierarchically structure a given decision problem according to several criteria. In a recommendation scenario, different features of the items such as the prize, brand, and so forth, could be decision dimensions. In the preference elicitation phase, users then have to compare *pairs of options* according to the given criteria by expressing which of the two options they prefer and how strong this preference is. The pairwise preference matrix is then converted through eigenvector computations into numerical values which express the importance (priority) both of the individual decision options as well as the individual criteria.

Several limitations of using the AHP method are known including the fact that all pairwise preferences toward item attributes have to be known before the results can be computed. Because of this disadvantage, AHP-based approaches are rarely used as a means of preference elicitation for recommender systems. However, some research works try to mitigate this shortcoming by introducing workarounds so that not all pairwise criteria preferences have to be known before recommendations can be generated. For example, in the approach proposed by Schmitt et al. [172], users can provide both constraints on individual criteria as well as AHP-based weighted relations between the criteria. They then combine both types of input based on the principles of multi-attribute utility theory (MAUT).

2.3.2 Critiquing. A quite different product-oriented approach for preference elicitation (and item space exploration) in the recommendation domain is called *critiquing*. Roughly speaking, the general idea is to propose a certain item to the user along with a description of its most relevant features and let the user then give feedback on (*criticize*) the product's features.

For example, when presented with a certain restaurant in a recommender application for leisure activities in tourism, the user could ask for a restaurant that is *closer* or *cheaper*. Given this user feedback, the recommender then tries to identify other restaurants that fulfill these new criteria. This process is then repeated until the user finds a suitable restaurant or gives up after having explored the item space by varying the preference statements.

Early applications of this approach to recommendation problems are the *RentMe* and *FindMe* systems described in [24, 25]. Later on, a variety of proposals have been made in the literature to improve the general idea or specific details of the procedure. For example, in *compound critiquing* approaches, users can criticize several item features in one single interaction. In the restaurant scenario, a compound critiquing system could, for example, offer the user a button labeled "*I want a cheaper and closer restaurant.*" The automated selection of the critiquing options—called *dynamic*

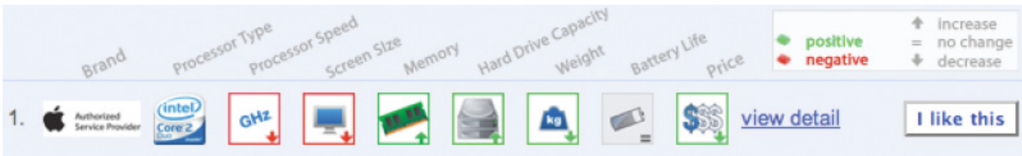


Fig. 7. The critiquing systems by Zhang et al. [204], which visualizes compound critiques with icons and colors. ©Zhang et al.

critiquing—is based upon the remaining space of available options and what the system considers to be the most probable next steps by the user [128, 157].

Both Zhang and Pu [205] and Chen and Pu [28] compare compound critiquing strategies with classic critiquing approaches by conducting user studies. They, however, come to different conclusions about their usefulness. The first study suggests that compound critiques can lead to higher accuracy with fewer interaction cycles. In contrast, the latter indicates that compound critiques result in lower accuracy and higher user effort. Some research works try to mitigate this potential problem by offering the user both the option to provide individual critiques as well as compound ones [158]. Others try to reduce the user effort of compound critiques, which are normally fairly long text blocks, by displaying the attributes of the combined critique visually with the help of colored icons (see Figure 7) [204].

Apart from dynamic and compound critiques, a wide array of other improvements to critiquing schemes have been proposed in the literature. These range from approaches to integrate the critiques back into a long-term user model [162], over studies about the domain-specific usefulness of critiquing [154], to very specialized solutions, e.g., for multi-party critiquing [77] or mobile critiquing [119]. Finally, researchers like Nguyen and Ricci [139] acknowledge the difficulty of testing critiquing approaches in user studies, a problem which they propose to mitigate by "replaying" critiquing sessions obtained from previous user studies. An obvious drawback of such an offline evaluation approach is that the recommendations on which the critiques are based cannot be changed after the fact. The authors try to overcome this limitation with different strategies, e.g., by finding an item in the recommendation list that is similar to the one that was actually displayed to the user, and by basing the simulated critiques on this proxy item.

A main advantage of critiquing-based approaches is clearly that the complexity of the general user interaction pattern is comparably low and the structured interaction mechanism can even be operated through voice commands [73]. On the other hand, the way in which users can specify their preferences is also quite limited, in particular if a basic approach with individual critiques is applied. This might lead to the effect that too many user interactions might be required to find a suitable item. Furthermore, again situations may arise in which no more products remain for the given user preferences. Additional recovery techniques are then required as mentioned above, e.g., by allowing users to specify the importance of different criteria [90].

In the context of this research, we consider critiquing primarily as a form of user preference elicitation. The nature of the underlying approach is, however, also helpful for the user as a means to explore the item space and to learn more about the domain itself, e.g., about the relevant features of the items. On the current MovieLens website, for example, a critiquing approach based on tags as described in [190] is provided for users to explore similar movies.

2.4 Alternative Elicitation Techniques Used in Recommender Systems

In the previous sections, we have discussed the most typical *explicit* mechanisms for preference elicitation in the recommender systems literature. In the following, we will discuss a selected set of alternative and partially indirect ways of capturing the user preferences from the recent literature. Again, we will only focus on explicit interactions and not on preference elicitation approaches that rely on the observation and implicit interpretation of user behavior.

2.4.1 Personality Quizzes. In recent years, a number of *personality-based* approaches to recommender systems have been proposed in the literature [142]. The underlying idea is that there is an inherent connection between the user's interests, their decision-making processes, and their personality. In particular, in emotion-driven domains such as music it seems plausible that the user's personality traits are connected to their listening preferences as discussed, e.g., in [148]. To determine the user's personality traits, the well-established *five-factor* personality model ("Big Five") is often adopted as an underlying theory (see, e.g., [130]). The elicitation of the personality traits of a user is then based on questionnaires (personality quizzes) of different levels of detail.

Personality profiles can be particularly helpful in cold-start situations and when users have little knowledge about the domain. Another use case is active learning situations, especially when the item space is very large. In these situations the personality traits of the user can be helpful to identify items to be rated. As mentioned above, Elahi et al. [48], for example, incorporate the user's openness to new experiences or their level of extroversion to find appropriate tourist attractions for them. Furthermore, studies like the one presented by Hu and Pu [85] indicate that such personality quizzes can also represent a viable way of developing a user profile when recommendations are sought for a friend.

In [46], three different user interfaces for acquiring personality traits were compared in a user study. Two of the interfaces were measurably better accepted by the participants, but the study produced no clear winner and both explicit and implicit forms of personality acquisition seemed to work. Somewhat surprisingly, only the explicit and form-based acquisition method led to a higher perceived ease of use. The authors' hypothesis is that the clear and transparent nature of this elicitation method was particularly well appreciated by the participants.

In addition to five-factor personality quizzes, other methods of identifying the user's personality traits have been explored in the recent literature. One example is to provide users with a color wheel to let them state their mood in an indirect way. In their work, Chen and Pu [31] use this low-effort form of input to decide upon the next music track in a group listening scenario. Similarly, Lamche et al. [118] try to simplify the personality acquisition process for the users by showing them pictures of personality "stereotypes" to choose from. The pictures can, for example, show a person dressed either in casual or in business clothes. Finally, a few approaches exist which try to acquire the user's personality traits indirectly, e.g., through their social media profile [63] or by observing their tendency to compromise in group decision scenarios [168].

Overall, the literature suggests that personality quizzes and personality-based approaches can be helpful in different situations, e.g., to learn a user profile for cold-start users, to choose items to rate in active learning scenarios, or to improve group decision processes. However, personality quizzes can be complex and tedious for users. Thus, appropriate measures have to be taken in order not to overwhelm users with too complicated or time-consuming procedures, in particular in mobile situations.

2.4.2 Comparing Item Sets and Rating Tags. Instead of showing the users individual items or features to rate or to compare, Loepp et al. [124] propose to present the user with successive choices between two item *sets*. To select the items for comparison, first a latent factor model is computed via classic matrix factorization. From then on, the user can choose between two sets of items which have very different characteristics with respect to one of the item factors in the matrix factorization model.

An advantage of this approach is the reduction of the user effort as the items in each set should be easy to compare because they have strongly differing attributes. Additionally, after the initial elicitation phase, no additional model training is needed, because the user has already been positioned in the vector space of the latent factors and recommendations can be made instantly. However, finding the most important factors in the model let alone asking the user to decide between all factors could be challenging.

Instead of asking users to rate items (or item sets), [49] and [86] follow a different idea to simplify the elicitation phase and ask users to state their preferences toward *tags*. The underlying idea is that it is easier for users to assess their attitudes toward the concepts behind a tag than to assess an item. In both mentioned works, the authors address a mobile scenario where users can provide like and dislike statements for tags, hoping that this more simple interaction approach decreases the chances of the user being overwhelmed in this on-the-go situation.

More sophisticated approaches such as [40, 66, 177] allow users to assign weights (ratings) to tags. In the implementation of di Sciascio et al. [40], for example, the influence of each tag on any of the item recommendations in the final list is encoded in different colors. This helps users to understand how their recommendations are generated and they can revise the tag weights to see an instant impact. Finally, Donkers et al. [43] relate tags to latent factors to be able to use the power of matrix factorization and at the same time aim to reduce the user effort by letting them rate tags instead of items.

2.4.3 Selecting Pictures and Shaping Landscapes. Neidhardt et al. [138] present a *picture-based* approach for customer preference elicitation in the tourism domain. Instead of asking users to rate a set of items or touristic locations, they are asked to select a number of pictures from a larger collection, which they find appealing when thinking about their next holidays. These picture selections are then related to a number of travel "factors" that were identified by the authors. Although the empirical evaluation presented in [138] is partially preliminary, the work represents an innovative way of allowing users to express their preferences in an often emotional recommendation domain, thereby avoiding fill-out forms that typically relate to objective and non-emotional features.

Finally, in the work by Kunkel et al. [114] users explore the item space in the form of a two-dimensional landscape. In this visualization, items are represented through pictures and the space between them is determined by their distance in the latent factor model. At the beginning of the process the item pictures reside on a flat landscape, which the users are then able to shape themselves. They can either "dig deeper" at certain points or "raise terrain" elsewhere indicating either interest or dislike in the respective item areas. In the end, the height of the landscape on which an item resides determines its importance, which can then be used for generating recommendations.

2.5 Discussion

In this section, we have discussed typical mechanisms both for acquiring the users' *long-term taste profiles* through ratings and for determining immediate *short-term constraints and preferences* through interactive elicitation techniques.

With respect to long-term profiles, works that rely on explicit user ratings are dominating the recommender systems research landscape today, despite the fact that in practice such explicit information is often sparse or non-existent while implicit feedback is typically available in abundance. A common drawback of using implicit feedback signals, however, is that the observed user actions have to be translated somehow, e.g., with the help of heuristics, into preference signals and that the interpretation of the signals can be subjective and noisy [102, 106]. While explicit feedback data can also be noisy or contain rating biases [6, 7, 208], explicit preference statements remain part of many recommender systems architectures, and more research is required to develop better mechanisms for acquiring such rating information.

This, in particular, includes new forms of stimulating users to provide reliable item ratings (see, e.g., [140]). "Gamification" could be one possible approach to obtain reliable rating information as discussed in [14]. The tendency of users of being willing to give feedback on items could probably also be dependent on the time and the user's situation. Similar to the question of when to provide proactive recommendations examined in [18], research could be done on determining the optimal timing to ask users for ratings. At the same time, *persuasive* approaches should be further explored as a means to stimulate users to give feedback on items. In some applications and websites including Amazon.com, users can navigate to a certain page with profile settings which allows them to fine-tune their profile to improve their recommendations. Future works could try to identify new ways of better encouraging users to leave feedback during the regular browsing experience. Rating data that is obtained "on-the-go" could furthermore carry important information about the user's context, which is typically lost, e.g., when users rate items long after they have actually consumed them.

From a design perspective of rating scales, much research has been conducted in other fields, but not many papers exist that aim to transfer these insights to the design of rating systems for recommenders. Do we, for example, need scales of different granularity for different domains or even for different users? Is it meaningful to adapt the rating scale to the screen size of the user's device [195]? How should we design the labels of the individual scales to avoid that users only provide extreme or biased ratings?[5] How can we avoid biases that might emerge when study participants are forced to make a choice even though they would not purchase any of the presented items in reality [39]?

More research is also required for systems that rely on interactive elicitation techniques to determine the user's short-term preferences. Knowledge acquisition and maintenance are known limitations of such systems and require the creation of appropriate development tools. Many questions are also open in terms of the "machinery" that is used to reason about the next interactional move and how to avoid that the user is overwhelmed by the complexity and the dynamics of the user interface. Despite being dynamic and adaptive, the interactive dialog systems from the literature follow predetermined interaction paths. With the recent advances in natural language understanding as demonstrated, for example, by voice-controlled smartphone assistants, more natural interactions and advisory applications should become possible.

Table 2 summarizes some of the key research challenges in the context of preference elicitation mentioned in this section.

3 RESULT PRESENTATION AND FEEDBACK ON RECOMMENDATIONS

The visual representation of a ranked list of recommendations is the central—and often the only—element of user interaction with a recommendation system. Typically, users can click on individual elements of the recommendation list, which leads them to further information about an item or the digital representation of the item itself, e.g., an audio or video playback.

In this part of the article, we will first discuss design alternatives related to the presentation of the recommendation lists and then review more advanced interaction mechanisms that can be implemented in the context of the presented recommendations.

[5] See [9] for a study on biases introduced by the choice of the labels.

Table 2. Selected Challenges in the Area of Preference Acquisition

Category	Specific challenges
Biases	Designing feedback (rating) mechanisms that do not lead to biases in the collected data
	Helping users state their preferences more consistently, e.g., by giving them a point of reference
Detailed feedback	Acquiring more fine-grained feedback, e.g., about different aspects of the item or the utility or relevance of an item in a certain context
User engagement	Stimulating users to provide *more* feedback, e.g., through gamification approaches
	Recognizing and dealing with user interest drifts
Novel interaction methods	Testing alternative ways of user interaction, e.g., natural language in dialog-based systems
Adaptation	Tailoring the elicitation approach to the individual user's needs
	Determining the next interactional move in conversational systems
	Combining long-term preferences with short-term needs

3.1 Design Alternatives for Item List Presentation

Let us first look at examples of how recommendations are presented in real-world applications. Various design alternatives are possible even if we limit ourselves to the case where the system merely presents clickable item lists and does not implement any advanced interaction pattern.

3.1.1 Design Alternatives in Practice. Table 3 shows a list of examples of such alternatives, which we created based on a review of typical real-world recommendation systems. Most of these design choices can have an impact on how the recommendation component is perceived and how it is accepted by the users. If, for example, not enough immediate information is provided for each list element, users may find it tedious to inspect each single element to assess if it is truly relevant or not. A poor choice of the list label, a too blatant placement, or a visual design that is not different from typical advertisements found on websites might give the user the impression that the recommendation list is just another list of ads.

Some of the design choices have to be made with the purpose of the recommendations in mind. If the main goal of the recommendations is to help the user find items that are similar—and thus possible alternatives—to the currently inspected items, showing too many items could increase the decision effort and cognitive load for the user. If, on the other hand, the goal is rather to help the user explore the catalog, the recommendation list could be longer, scrollable, or even "virtually endless" as in the music or video streaming domain.

In the following sections, we will review the literature on these design alternatives, which all can have an impact on the users' experience and satisfaction. Again, our focus is on user interaction aspects and not on questions related to the general strategy quality of the underlying item and ranking selection process, e.g., whether the list should present complements or substitutes [42].

3.1.2 Choice Set Size. Item list characteristics have long been a field of study in the Decision-Making literature [87, 174, 178] and many of the obtained insights are relevant for the design of recommender systems. A core question in that context is how many items we should present a user to choose from. Research shows that providing more options is not always better [174] as

Table 3. Examples for Design Alternatives for Recommendation Lists

Design aspect	Considerations
Item description	What kind of information is displayed for each item, e.g., title, thumbnail picture, community rating, short summary when hovering, price?
List label	How is the label of the list chosen, e.g., "*Recommended for you*"?
Screen position	Is the list positioned (horizontally) below a reference item or (vertically) on its side? Is it displayed on an item detail page, integrated into the search results, or presented on its own page?
List length	How many items are displayed? Can the list be scrolled (endlessly)?
Number of lists	Are all recommendations displayed in one list or do we have multiple lists as on Amazon.com or Netflix?
Organization	Does the list contain a grouping of items? Does the list support side-by-side comparisons of items? How are items ordered and are they highlighted in some way?
Advertisements	Does the list contain advertisements, e.g., in the form of editorially selected items; are these promoted items visually different from the other recommendations?
User control	Can users change the default order of the recommended items based on other criteria, e.g., based on the price or the star rating? Can users give feedback on the recommendations or delete certain items from the list?
Visibility	In case of dynamically activated recommendation popups: When should the overlay be displayed? Where should it be placed?

this can easily lead to a *choice overload* situation. This in turn can result in longer decision times, higher decision effort, lower satisfaction, and less accurate choices. Diehl and Poynor [41], for example, identified that when users are faced with a larger assortment, they also expect to find better items. However, when they do not find an item that matches their optimistic expectations, their disappointment is also stronger.

Other studies have found that even the motivation to make a choice at all is decreased when too many options are present. Participants in two different studies were, for example, more likely to buy gourmet food or take part in optional study assignments when the option size was 6 instead of 24 or 30 [87]. This phenomenon can generally be quantified by an inverted U-shaped function with the choice set size on the x-axis and either the sales volume or the choice satisfaction on the y-axis, indicating that either too few or too many choices result in non-optimal sales effects [159, 178]. However, it has also been observed that the peak of this function, i.e., the point that leads to the highest satisfaction, shifts to the right when the items are easier to compare [159]. Vice versa, studies like [51] suggest that too *diverse* lists can increase choice difficulty. Thus, the optimal choice set size depends on the item characteristics, the items' relation to one another, and the user's affinity toward them.

In the field of recommender systems, Bollen et al. [21] conducted a study whose results indicate that a U-shaped relation between list size and satisfaction could also exist in recommendation scenarios. Other studies [5, 116] explore whether the presentation of *personalized* item lists, which are typical in recommender systems, can help to reduce the choice overload problem. The underlying idea is that these lists presumably only contain items that are generally relevant for the user. This assumption was confirmed in the study by Aljukhadar et al. [5], where participants were more likely to seek the help of recommenders when they suffered from overload. At the same time, a higher choice quality in complex situations was observed when the participants were provided with personalized recommendations. In contrast to these findings, the work by Cremonesi et al. [33] suggests that personalization can cause an even higher overload because the items are in general more relevant, which stimulates the users to explore the options in more depth.

The approaches discussed so far assume that the choice set size is the same for every user. Schwartz et al. [175], however, argue that using the same item list size for an entire user population can be inappropriate because different groups of users have different strategies when choosing items. For the so-called *maximizers* an optimal choice set size, i.e., a list that is neither too short to be engaging nor too long to be overwhelming, is highly effective because those users always search for the best option until exhaustion. However, so-called *satisficers* do not necessarily consider all options. Instead, they stop searching when they find a satisfactory solution, which means that using larger choice sets is an acceptable strategy for such users. A corresponding user-centric approach was proposed by Scholz and Dorner [173]. Their system tailors the size of the recommendation list to the users based on their own rating distribution. The novelty of their approach is that in their system not only the content is personalized but also the presentation. However, the number of research works in this field is scarce, which is possibly due to the difficulty of evaluating such approaches.

Besides the choice set size, other list characteristics like its diversity can impact the users' decision-making process as mentioned above (see [180] or [193]). Additionally, including dominated items which are objectively worse than others can make it easier for users to decide or to persuade the user—with the help of psychological effects—to adopt a certain recommendation [1, 108, 183]. We will discuss these aspects later on in Section 3.5.

Overall, the Decision-Making research literature suggests that the choice set size can have a significant impact on the utility of the recommendation system. However, only limited research exists in the RS literature and more work is required to understand how to determine an optimal list size depending, e.g., on the domain or the user's personality.

3.1.3 Clustered Lists and Multiple Item Lists. In most RS research works the assumption is that the goal is to produce one personalized ranked item list. In case such a list contains a comparably diverse set of items, it might, however, be helpful to group the recommended items in order to make the decision process easier for the user. A few works such as [29, 135] exist which suggest that grouping the items into meaningful clusters can improve user satisfaction. Nanou et al. [135], for example, propose to group movie recommendations by genre instead of ranking them only by the predicted rating in order to achieve a higher acceptance level.

The use of multiple recommendation lists—e.g., one for each item category—was already reported by Schafer et al. [169], who surveyed real-world systems such as Amazon.com, CD NOW, and drugstore.com. Often, the used list headers help the users understand the underlying recommendation strategy or the recommendation purpose, e.g., "*Customers who bought ... also bought*" or "*Gift ideas.*" Having multiple lists should in general help users find relevant items more easily depending on their shopping intent. In addition, the provided list headers can have an explanatory character as will be discussed later.

In academic research, two different lists were, for example, used in the work of Plate et al. [149]. Besides a traditional recommendation list, the authors also included a list of reminders (as navigation shortcuts) in the user interface, which displays those items that the user has most recently visited or rated. Similarly, Knijnenburg et al. [111] offer the user a feature similar to a shortlist by allowing them to re-sort the items from the recommendation list. The user actions are then directly fed back as a signal into their system.

Generally, the recommender systems literature on using multiple item lists is very scarce, despite the fact that many e-commerce platforms like Amazon.com or media streaming sites like Netflix heavily rely on the usage of up to a few dozen lists and in the case of Netflix use machine-learning algorithms to determine an “optimal” organization of the lists Alvino and Basilico [209].

3.1.4 Item Ordering and Organization. The order in which the items of a choice set are displayed is also rarely a research topic in recommender systems literature. The implicit assumption is that items are ranked according to their assumed relevance, e.g., in terms of the predicted rating. Having the presumably most relevant item at the top of the list also seems reasonable as many studies show that these items receive the highest attention by users.

In some domains, predicted relevance might not be the most important criterion [81, 132], and, for example, novelty or diversity might have a stronger impact on user satisfaction. To allow the user to influence such criteria, some approaches were proposed in the literature where users can, e.g., manually filter the recommendations [170], directly fine-tune the underlying strategy [78], or change the strategy mix [146]. We will focus on such user control techniques in Section 3.4.

Another aspect that is relevant in practice is that presenting the same set of top-ranked items to the user on every visit might lead to a limited attractiveness of the recommendations. One possible remedy could be to simply randomize the order of the top-ranked items to introduce some variety and diversity. Given that the predicted relevance scores of the top-ranked items are often very close to each other, such a shuffled selection of items might probably not even negatively influence the perceived recommendation quality.

Two more examples of non-traditional list layouts were explored in the study by Chen and Tsoi [30]. They compared both a grid and a circle arrangement to a traditional top-to-bottom list. Through their study they observed that the grid and list layouts received most of their clicks in the top-3 item region. In contrast, the circle layout showed a more even distribution among the recommended items and also increased the user’s decision confidence and enjoyment.

In general, research in alternative list organizations and item orderings is not very rich. The study reported in [65] is one of the few examples which analyzes the effects of different item orderings. Furthermore, as the relevance of an item can depend on a user’s current context and vary over time, it seems advisable to provide the user with a means to re-rank or re-organize the recommended item list. We will discuss such user control mechanisms in more depth in Section 3.4.

3.2 Visualizations Approaches

The most common form of displaying recommendations in practice is the presentation of the items within an ordered list. Each item is typically displayed in the exact same visual form, no clues are given regarding possible relationships between the items, and only items with the highest presumed relevance are displayed. In this section, we will review alternative approaches that deviate from this form of presentation.

3.2.1 Item Highlighting and Visual Item Presentations.

Highlighting. In a classical Information Filtering perspective on recommender systems, non-relevant items are completely filtered out from an incoming stream of information. In some applications in contrast—including, in particular, news feeds or social media sites—most or all information items should be displayed and the order of the items is not (only) based on relevance but, for example, on recency.

Waldner and Vassileva [192] propose a highlighting approach to help the user more easily detect the most relevant items on Twitter streams (see Figure 8). One advantage of such highlighting

Fig. 8. The highlighting system devised by Waldner and Vassileva [192], which instead of filtering twitter feeds indicates the items' importance visually. ©Waldner and Vassileva

approaches is that the familiar user experience remains unchanged with the added benefit of providing pointers to the most important posts. At the same time, as no item is removed, users might feel more confident not to miss any information compared to situations in which the presumably non-relevant items are not displayed at all.

Highlighting items can, however, also serve other purposes than to discriminate between relevant and non-relevant items. For example, di Sciascio et al. [40] employ a keyword-based document recommender system that calculates how often each keyword appears in the recommended documents. Then, instead of simply displaying the most important keywords beside each recommended document, color coded bars are used to indicate the relative importance of the keywords for each item.

Item Presentation. Usually, when presenting recommendation results as a ranked list, only limited information about each recommended item can be displayed. In many domains including e-commerce settings, the probably most common presentation form is to display a thumbnail picture of each item along with a short textual characterization that includes the product name and its price.

Only a few works exist in the literature that explore different strategies of presenting the items or analyze which details should be displayed in the recommendation list. In [135], for example, the authors compare different ways of presenting movie recommendations, e.g., using only text, text and images, or videos. They also assess if users prefer top-n lists or lists that are organized by genre. Regarding the latter aspect, users found the genre-organized lists more persuasive. As for the tested modalities, the combination of text and video led to the highest user satisfaction.

In another study, Karvonen et al. [105] examined the importance of the inclusion of reputation information (e.g., in terms of average customer ratings on hotel booking sites) for each recommended item. Their work suggests that this information should be displayed prominently and that it should be presented in a form that is easy to comprehend. This is reflected in many real-world systems, where the inclusion of the average community rating along with each recommended item is nowadays quite common. Nevertheless, how to visualize the community ratings can in itself be a design challenge, and the choice of the visualization approach can have an effect on how users perceive the quality of an item [83].

Finally, Yang and Yuan [201] propose a travel destination recommender which presents the items primarily through pictures, with an emphasis on the match between the pictures' emotional connotations and the users' search goals. Unfortunately, no systematic evaluation of this approach that "considers the psychological emotion needs" was done so far.

3.2.2 Diagrams and Graphs. One problem of typical list-based approaches is that they are not very well suited to visualize relationships between different items or the relationships of the items to other things. A small number of alternative forms of presentation using diagrams and graphs was presented in the literature.

For example, the goal of the work by Vlachos and Svonava [191] is to visualize the relationships between items. They organize the set of recommended movies in a graph structure and cluster them according to their similarity. Users of the system can browse these clusters to find additional relevant movies and they can also fine-tune the number of clusters, which allows them to adapt the visualization to their taste.

Another form of visualizing item relations was suggested by Zhao et al. [207], who propose a presentation based on word clouds extracted from the social web. Clicking on words in the cloud then reveals people that are related to this latent topic as well as pointers to additionally relevant topics. The work of Graells-Garrido et al. [72] has similar goals but uses different visualization strategies. In one of their approaches, the authors, for example, propose to display groups of users in a "circle packing" layout by clustering them according to the latent topics they are tweeting about.

A more interactive system is presented by Knijnenburg et al. [110]. Their idea is to present a list of recommended music tracks next to the users' listening history and their social friends. When users click on a recommended track, the system displays a graph overlay that connects the track to friends related to the track and to tracks that the user has listened to in the past. This additionally presented information serves both as a form of explanation and as an inspection tool for users to judge the relevance of an item. In their user study the authors observed among other things that understandability was increased with their visualization approach compared to a traditional list. However, only marginal improvements in the perceived recommendation quality were observed.

A similar social graph based visualization has been proposed by Gretarsson et al. [75]. In their approach, users can additionally move the nodes of the graph as a form of preference feedback. Among other insights, a user study revealed that participants preferred the proposed interactive visualization approach over Facebook's built-in recommender system.

In an effort to highlight the relationship between a recommended item and the underlying recommendation strategy of a hybrid recommendation system, Parra et al. [147] created a visualization based on Venn diagrams (see Figure 9). Specifically, the diagram indicates which of the individual strategies was responsible for the inclusion of an item. Similarly, Verbert et al. [188] employ a cluster-based visualization approach that displays the interrelationship between recommendations (generated by different strategies) and users who bookmarked the items.

3.2.3 Comparing and Locating Items—2D, 3D, and Maps. Showing items in a diagram or graph structure can help explain their relation to each other. However, the absolute positions have no meaning in such representations without a coordinate system. Approaches like the one proposed in [16] therefore position the items in a 2D coordinate system. In their system, the item price is represented on one axis and their estimated *true value* on the other, which allows users to easily identify the items with a high cost-benefit ratio and compare their relevance on a visual level.

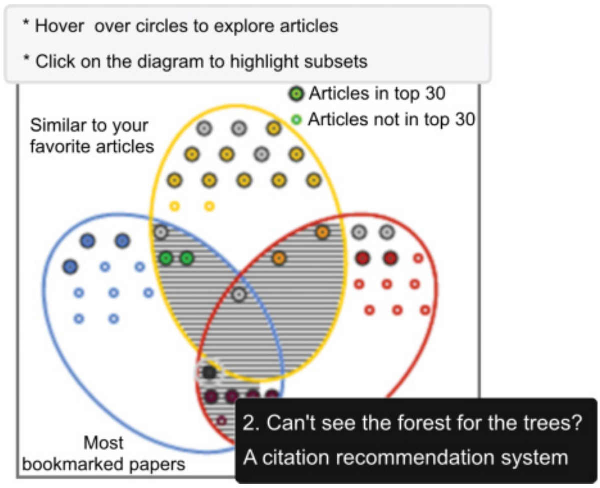


Fig. 9. One component of the recommendation visualization system by Parra et al. [147], which displays the items in a Venn diagram. The diagram indicates which of several strategies led to the recommendation of each item. ©Parra et al.

In contrast, in the 2D representation of Andjelkovic et al. [12] the coordinate system that is used to display musical artist recommendations is based on an underlying "latent mood space." In addition, a user avatar is positioned in the latent space according to the user profile, and users can give feedback, e.g., by repositioning their avatar closer to artists they like. A similar approach was proposed by Kunkel et al. [114]. Their visualization method positions similar items next to each other on a 2D plane, while the user's current interest in an "item area" is visualized in the third dimension. Thus, both of the above-mentioned systems capture the relation of the recommended items to each other as well as the user profile in the same 2D or 3D space, respectively.

A number of further works propose to display the recommended items on a geographical (city) map. This choice is natural for certain application domains, e.g., in the tourism domain when the recommendation result is a set of places to visit [200]. A map is also employed in the system by Daly et al. [36] to help the user explore housing recommendations that take into account the traveling distance to important points that the user regularly visits.

3.2.4 Discussion. In practical applications, we can often see that product finder systems support a side-by-side view of several items, which allows users to compare the individual features of the items. Such comparably simple approaches to support the user in the decision-making process are not in the focus of the recommender systems literature.

On the other hand, in the research field of Information Visualization, see e.g., [127], a considerable number of proposals exist to visualize larger information spaces and relationships between objects in order to support the human cognition process. The use of such advanced visualization methods to support users when exploring the item space appears to be a promising area of research in the field of recommender systems.

3.3 Explanation User Interfaces

In many domains, presenting a set of presumably relevant items without any further information about why they were recommended might be insufficient for users to make well-founded decisions. System-generated explanations, which are displayed together with the proposed items, are known to be valuable decision-making aids for general decision support systems [74]. In the context of recommender systems, Tintarev and Masthoff [185, 186] identified a number of ways in which a system and its users can benefit from providing explanations, e.g., in terms of increased efficiency, perceived transparency, user satisfaction, and trust. In this section, we will review explanation components as a part of the user experience of recommender systems from an academic and practical perspective.

3.3.1 Labels as Explanations. On many online sites that serve recommendations to their users, only little or even no information is provided on why the system decided to present a particular set of items in the given order. In some cases, the labels of the item lists indicate how the system came up with the recommendations. A typical example is Amazon.com's explanatory label "*Customers who bought ... also bought,*" which suggests that the recommendations are based on the behavior of the user community. Other typical labels that carry some information are "*Inspired by your browsing history,*" "*Similar products,*" "*Trending,*" and so forth. In some cases, however, labels like "*Recommended for you*" or "*You might be interested in*" give no indication about the way the recommendations were created or which purpose they serve. In particular, when multiple recommendation lists are used as discussed in Section 3.1.3, using meaningful labels seems to be a necessity.

In the research literature, Pu and Chen [152] use different labels in their "trust-inspiring" explanation interface. In their user interface the recommendations are organized in different groups based on their tradeoff characteristics compared to a reference product, e.g., "*Products that are cheaper but have lower processor speed.*" Their study indicates that users found a system with such an organization interface is better capable of supporting them in their decisions than a system that uses a traditional item presentation approach.

3.3.2 Knowledge-Based Explanations with Textual Arguments. One possible goal of explanations is to make the system's reasoning process transparent for the end user. How to generate human-understandable explanations for computer-generated advice has some history and was explored, for example, in the context of rule-based expert systems in the 1980s. The particularity of such systems is that explanations can be derived based on the internal reasoning "traces" of the underlying inference engine, e.g., a rule processor.

In knowledge-based approaches to recommendation, similar principles can be applied. The system could, for example, use the set of filter rules that were applied given a set of user preferences as a basis to derive a human-understandable explanation as done in [53]. Zanker and Ninaus [203], in contrast, discuss an approach which also applies a knowledge-based method but aims to decouple the recommendation logic from the explanation logic. The specific goal of their approach is to automatically construct convincing user-specific arguments why a certain item is a good match for the user's interest profile. Technically, this is accomplished with a knowledge-based explanation model and a reasoning method that is independent of the applied recommendation mechanism.

In the context of conversational recommenders, explanations can also be helpful in situations where the user requirements cannot be fulfilled. In such cases, explanations can be used to convey the cause of the situation to the user and offer possible recovery options in natural language [27]. Such explanatory texts can also be applied to compound critiques to increase transparency and thereby reduce the choice difficulty [129].

Many of the discussed approaches have in common that they rely on manually created explanation text fragments, which are selected and combined by some reasoning component to form a coherent textual explanation. The resulting explanations can in the best case feel "natural" and convincing. Building such an explanation generation component, however, requires additional domain- and knowledge engineering efforts and the quality also depends on the writing skills of the domain expert or engineer.

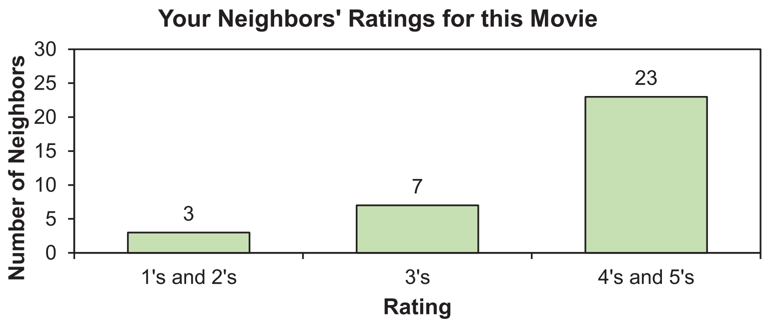


Fig. 10. Explanatory visualization of the best-performing method from [80]. The explanation interface shows the rating behavior of the "neighbors" of the current user with respect to a certain movie.

3.3.3 Explaining Content-Based and Collaborative Recommendations. For techniques that are rather based on the automated detection of behavioral patterns in a larger community than on explicit rules, knowledge-based approaches are not applicable. Thus, many approaches from the literature focus on alternative methods to extract meaningful information from the available data on which explanations can be built.

Vig et al. [189] and Ben-Elazar and Koenigstein [17], for example, search for correlations between social tags and users to justify the recommended items (often "post-hoc") using tags that the users are assumed to be interested in. Analogously, content and metadata features of the items can be exploited, e.g., actors in movie recommendation scenarios [182] or keywords in research paper recommendation scenarios [136]. Even more complex methods try to extract sentiment information from the items to find more meaningful keywords [206]. In all the mentioned cases the results can then be displayed as a simple explanatory text like "*Because you were interested in … in the past.*"

A variety of explanation styles and corresponding visualizations for collaborative recommendation approaches was designed and evaluated with respect to their *persuasiveness* in [80]. Figure 10 shows the best-performing method from their study. The explanation method uses information about the internal reasoning of the algorithm—in that case the rating behavior of the user's neighbors for the explained item. The downside of this method is that the explanatory approach is very complex in comparison to what typical real-life systems offer. However, also black-box explanations—which actually do not even have to correspond to the reality—were surprisingly successful in their study. An example of such an explanation is "MovieLens has predicted correctly for you in 80% of the time in the past."

Persuasiveness, i.e., convincing users to adopt a recommendation, however, is not the only goal that one can try to achieve with explanations. Tintarev and Masthoff [186] identify seven possible goals: transparency, scrutability, trust, effectiveness, persuasiveness, efficiency, and effectiveness. When designing an interface to communicate an explanation to users, it is therefore important to keep the explanation goal in mind as different styles of presenting the information may be particularly suited for one goal but not for another. Which goal a designer aims to achieve in turn depends on the intended utility of the recommendation system. This can be, for example, providing improved decision support or driving the short-term shopping behavior of users.

Gedikli et al. [67] conducted a user study to analyze the effects of different types of explanations with respect to several of these goals. The study included 10 different explanation approaches. Among them was a subset of the techniques explored in [80], two "content-based" approaches which rely on tag clouds as a means for visualization, and one baseline method which displays the average community rating as an explanation. The results show that content-based approaches, which not only use numerical information like neighbor ratings but textual information like keywords [19], require more cognitive effort from the user (decreased *efficiency*) but seem to be more suited to help users make better decisions (increased *effectiveness*) and are often perceived to be more transparent. Some of the simple methods lead to the effect that users consistently overestimate the quality of a recommended item (increased *persuasiveness*), which might result in negative long-term effects in terms of user trust. The choice of the explanation method again depends on the importance of the individual goals in a given application setting.

In particular, in the context of collaborative filtering recommendations, an open challenge is that explaining the underlying reasoning often seems impossible given the complex machine-learning models that are used in the literature. Alternative methods are therefore needed that, e.g., "reconstruct" a reasonable explanation for users based on the item features and the user profile as done in [203].

3.3.4 Interactive Explanations.

Academic Approaches. The explanation approaches discussed so far aim at providing additional, "static" information to be displayed along with the recommended items. Schaffer et al. [171] go one step further and propose a more interactive explanation style where users can further inspect the provided explanations by looking at the users in their assumed neighborhood and their associated item preferences. As a result, the operating principle and the data source for the recommendations is made transparent for the user. A similar approach was presented in an earlier work by O'Donovan et al. [145], who proposed a visualization and explanation tool that supports the genre-based manipulation of the neighbors that should be considered in the recommendation process.

Another interactive explanation interface was proposed by Lamche et al. [117], where the explanation mechanism is used to collect feedback in order to improve the recommendations. Their system can, for example, explain a recommendation by stating "*Because you currently like shoes*" and give the user the option to mark this assumption as being incorrect. The user is then guided to a preference revision page, where they can provide more feedback regarding the system's incorrect assumptions. In contrast to the feedback mechanism of Amazon.com (see next paragraph), their approach allows users to express that they are not only not interested in a particular pair of shoes but that they are not interested in shoes in general.

Scrutable Explanations in Real-World Systems. On real-world websites explanations for collaborative recommendations are still rare and limited to the choice of the label of the recommendation list (e.g., "*Customers who bought … also bought*") or the presentation of comparably simple additional information such as the average community rating, a predicted rating, or—in the case of friend recommendation on social networks—information about joint contacts. The e-commerce site of Amazon.com is one of the few exceptions. As of 2016, item recommendations are often accompanied by an explanation in terms of one or more "supporting" items that the user has recently viewed, added to their wish list, or purchased. This additional item is labeled as "*Because you purchased …*" or a similar statement as a justification. Similar to the work by Lamche et al. [117], users can give feedback to the system in this context and state that this additional item should not serve as a basis for recommendations in the future. A screenshot of Amazon's explanation interface is shown in Figure 11.

Another example is the already mentioned Netflix user interface (see Figure 2), where users can state that they are "*Not interested*" in a certain movie recommendation. Additionally, the system explains recommendations in the form of "*Based on your interest in: … .*"

Fig. 11. Explanation and feedback features of Amazon.com. The explanation suggests that the item in question was recommended because of two other items that the user has purchased or added to the wish list.

Discussion. Overall, interactive explanation approaches, like the mentioned ones, demonstrate that explanations have the potential to become more than just static decision aids. Users can employ such facilities to explore details about the recommender algorithm's inner workings and put the user into control of the recommendations, as will be discussed next in Section 3.4. However, research on this topic is still quite limited and it is, in particular, unclear how to design interactive interfaces in a way that they remain intuitive to understand and operate for the average user. Furthermore, the recommendation lists should in the best case immediately reflect the feedback that is provided through the explanations, which appears to be challenging for complex machine-learning models.

3.4 Feedback on Recommendations and Putting the User into Control

Amazon.com's explanation feature shown in Figure 11 in fact serves multiple purposes besides providing the user with information why a certain item is recommended. Users are also stimulated to rate the recommended and the "supporting" item and state whether they already own the recommended item or if they are not interested in the recommendation. Also, users can explicitly declare that the supporting item should no longer be considered for recommendations.

In this section, we will review research approaches that allow users to give feedback on the recommendations as well as methods that put the user more into control of the recommendations.

3.4.1 Gathering and Incorporating User Feedback. Allowing the user to give feedback on the recommendations is not uncommon in practice. Friend recommendations on social networks as well as music or video recommendations on media streaming sites can be skipped; Question-And-Answers (Q&A) sites and review sites often let their users rate the helpfulness of an article. In most of these cases, however, it remains unclear for the user if this feedback will have an impact on future recommendations. In research approaches, giving feedback on recommendations—which is immediately taken into account—is the central interaction mechanism in critiquing-based systems (Section 2.3.2). In the following, we will, however, focus more on learning-based approaches to recommendation where the user feedback in many cases cannot be directly and unambiguously translated into constraints on item characteristics.

In [136, 137], for example, the users of a research paper recommender can provide feedback to the system by expressing their disinterest or by assigning importance weights for a set of keywords extracted from the recommendations on a continuous scale. An initial user study suggests that many users adopted these feedback mechanisms and that only a small minority of participants felt annoyed by them.

In the above-mentioned 3D visualization system by Kunkel et al. [114], users can literally "dig deeper" into the landscape or "raise terrain" to express their interest or dislike toward areas of similar items. As a response, the recommender system then immediately updates the recommendation list and the 3D visualization accordingly.

A special form of putting the user *back* into control in case of an automated learning system is proposed in [151]. The particularity of their proposed approach is that it combines a learned user model with a user-specified interest model to deal with cold-start situations. Both models are expressed in terms of "like-degrees" for certain item characteristics (e.g., movie genres). Whenever the user manually updates the interest profile, more weight is given to the user's explicit statement compared to the learned model. In contrast to other approaches, the user feedback is therefore not on the recommendations but on the resulting user model.

Finally, Hijikata et al. [82] conducted a study to investigate to what extent the presence of different user control and feedback mechanisms—such as the option to provide context information or to manipulate the user profile—can have a positive effect on user satisfaction. However, their results were not fully conclusive. In the end, it remained unclear if the participants who were provided with the feedback mechanisms were more satisfied because of the presence of the functionality or because of the resulting improvements in recommendation accuracy. A third group of participants was also provided with the feedback mechanisms, but their feedback was not taken into account in the process. It turned out that participants of this group were even less satisfied with the system than those who had no way of providing feedback to the system at all.

3.4.2 Dynamically Adjusting the Recommendations. In most research works, the resulting recommendations are assumed to be displayed in the order of the estimated relevance for the user (e.g., based on a rating prediction). Since the estimated relevance can deviate from the user's short-term preferences, different proposals were made to give the user more control over the process of filtering and adjusting the recommendation list.

In the basic approach by Schafer et al. [170], users can filter the recommendations based on certain features, e.g., genre or release year. A study indicated that users prefer to view a larger recommendation set when they are able to filter it by themselves. This suggests that they already appreciate this simple type of control. Similarly, in the approach by di Sciascio et al. [40] users can filter the results based on their tags. In addition, they can provide weights for the tags. Their approach not only allows users to manipulate the set of recommendations according to their short-term interests, it can also be used to improve the long-term profile.

Yet another form of allowing users to dynamically adjust the recommendations was proposed by Chau et al. [26], whose graph-based item space visualization allows users to re-arrange recommendations spatially, categorize them in custom groups, and mark items as favorites. The goal of their system was to help users understand complex item spaces. Unfortunately, no larger scale study has been conducted yet to evaluate the effectiveness of the system in that respect.

A different form of user control is implemented in the interactive explanation system by Schaffer et al. [171] (which was already mentioned as an explanation approach in Section 3.3). In their system, users can not only inspect why a certain item is recommended but also interactively change their profiles, e.g., by modifying or deleting ratings, and immediately view the effect on the recommendations. A similar approach was presented in [120], where users can express trust ratings toward their neighbors depending on the item category. Whether or not such approaches are applicable in practice, where neighborhoods are large and anonymous and where the computations cannot be applied in real-time, remains however, open.

In case of social network based recommendations, the situation can be different as users should know their friends and the neighborhoods are smaller. Bostandjiev et al. [22] implemented this idea of rating the social network neighbors and observed that this form of user control led to higher overall satisfaction and a higher degree of control than when using a non-interactive system.

Also in the context of social networks, Tintarev et al. [184] propose a system that automatically identifies meaningful user groups in Twitter feeds and allows the user to filter the feeds by group. They also enable users to retrieve additional posts that are n hops (conversation steps) away from groups that they like. However, participants of their preliminary user study were often confused by the hop filters, which emphasizes the need for intuitive user interface mechanisms when putting the user into control.

Finally, in the experiment by Harper et al. [78], users could interactively fine-tune certain desired characteristics of the recommended movies, e.g., in terms of their popularity or recency. The users were presented with a set of recommendations and could then use different knobs to shift the results in one or the other direction until they were satisfied. A survey at the end of the experiment showed that in the end users were more positive about the adapted recommendations than about the original ones, which is another indicator of the potential values of putting users into control.

3.4.3 Choosing or Influencing the Recommendation Strategy. A different method of putting users into control is to let them influence the strategy or algorithm based on which the recommendations are generated [44, 47, 146, 147, 179]. Most of these systems allow the user to choose between different pre-implemented strategies or let them assign importance weights.

In the system by [147], for example, users can weight comparably simple strategies like "*Most bookmarked*" or "*Similar to your favorites*," and their studies indicate that users like to use slider controls to manipulate the strategy mixture.

A common problem of such solutions is, however, that complex strategies cannot be easily explained to the user. Giving them, for example, the option to add more weight to a matrix factorization strategy, is most probably not very helpful. Ekstrand et al. [47] try to work around this problem by giving the algorithms arbitrary names and by only providing weak hints about their reasoning, hoping that users will work out the characteristics of the algorithms themselves. Their study indicates that users do indeed switch strategies, even when they are not told upfront how they work.

3.5 Persuasive User Interfaces

In the previous section, we have seen how explanation mechanisms can serve as a starting point for putting the user into control of the recommendations. Explanations are, however, also a possible means to *persuade* or convince users to adopt certain recommendations as described in Section 3.3. In this section, we will discuss explanations and other techniques to build persuasive recommender systems.

According to the Merriam-Webster Dictionary [212], persuasion is defined as "the act of causing people to do or believe something." In the context of recommender systems, persuasion aspects can play a role in two different forms, where in both cases the persuasive system has an *intent* to change the behavior of the users [59]. First, recommendation providers are typically interested in users considering the recommended items in general for their decision-making process. This can, for example, be achieved by displaying different forms of "persuasive cues" during the interaction that increase the users' trust in recommendations by the system [202]. Factors that can help increase the persuasion potential of an application, for example, include the *credibility* of the system, the level of user control, and the cognitive effort during the preference elicitation phase [76, 199].

Second, persuasion in recommender systems can also be related to the goal of *promoting one or more particular items* in the choice set, which is not necessarily the most useful option for the customer but maybe the most profitable option that still results in an acceptable level of user satisfaction (see, e.g., [89]). Whether or not this is desired, again depends on the business goal and the intended effects of the recommendation service.

3.5.1 Persuasive Explanations. Explanations are a key mechanism for persuasiveness in the RS literature and persuasiveness was correspondingly in the focus of the early study by Herlocker et al. [80]. Explanations can serve both above-mentioned goals. Bilgic and Mooney [19], for example, argue that transparency—as provided by explanations—in general has a persuasive effect because well-informed users are more likely to make a purchase decision. On the other hand, explanations can be used to push certain items, e.g., by deliberately omitting information, focusing on less relevant details, or presenting explanations that persuade users to choose an option that is not optimal for them but profitable for the provider [68, 185].

Gkika and Lekakos [68] investigated comparably simple forms of explanations which were not related to the inner workings of a recommendation strategy with respect to their persuasiveness. Their strategies included, e.g., the presentation of messages related to scarcity ("*Soon to be discontinued*") or commitment ("*You should try new things*"). In the end, however, the most persuasive types of explanations were those related to authority ("*Won the Oscar*") and social proof ("*Has a high community rating*"). In a similar research, Amin et al. [8] identified that social relations are the best strategy to promote an item ("*Popular with your friends*").

To some extent, the literature indicates that often very simple persuasive cues like displaying the average community rating can be effective, which was also observed by Herlocker et al. [80]. According to the results by Gedikli et al. [67], however, these forms of explanations can have the effect that users overestimate the true value of a recommended item. This can in turn lead to disappointed users and, in the long run, to limited trust in the recommendation system.

3.5.2 Persuasive Item Selection and Presentation. How a recommendation list is designed can be a key factor that influences the persuasiveness of a system. We have discussed different approaches that are based on Decision-Making principles in Section 3.1. One typical question in that context is related to the size of the choice set (list length), which has been extensively investigated over decades outside the field of computer science, e.g., in consumer research. An example of such research is the work of Sela et al. [176], who observed that users who were presented a larger product assortment—and a correspondingly more complex decision problem—were more likely to make more reasonable (utilitarian) and *justifiable* choices. Generally, their work connects questions of assortment size with justifications, which can both relate to the final decision satisfaction. In the RS literature, only a limited number of works in that direction exist. A practical question in that context could be which item attributes should be displayed in result lists, as consumers will use these attributes to derive justifications for their choices.

With respect to the question of how to persuade users to choose a certain item, a number of psychological phenomena exist, which are typically applied in the context of advertising and, more generally, marketing. Most of these psychological effects can be achieved through a specific selection and ordering of the items. As stated in Section 1.2, our work generally focuses on user interaction aspects and not on *what* is being recommended. For the sake of completeness, we will therefore only briefly discuss a few recent works here.

Felfernig et al. [56], for example, examined the role of *primacy* and *recency effects* in recommendations lists. Their observations indicate that users are more inclined toward the first and last option in a list of choices, while the purchase likelihood for the middle options was slightly lower.

Other phenomena observed in recommendation scenarios include *anchoring effects* and *decoy effects*. Anchoring effects, where users adjust their own evaluation of an object to a given reference point, were for example investigated in [1, 2, 32]. In these works, the authors for instance investigate to what extent displaying the average star rating of an item has an impact on the user's rating behavior (preference construction) or their economic behavior in terms of "willingness-to-pay."

Decoy effects in recommendation lists were explored by Teppan et al. [183]. The general idea of exploiting this effect to bias users in their decision is to purposely include so-called decoy items in the choice set. These items are dominated by the "target items," i.e., the ones to be promoted, in most dimensions, but they also slightly dominate other "premium" items. As a result, the user can easily identify the target items as the best possible choice, which in turn leads to higher satisfaction and confidence in the choice.

Finally, there are also other psychological phenomena like the "framing effect," where the decision of a subject depends on whether the choice situation is framed positively or negatively [187], which have not yet been studied in depth in the recommender systems literature and remain a topic for future research.

On a more general level, Cremonesi et al. [34] raise the question which of the typical quality characteristics for recommendation lists has the highest persuasive power. Their results indicate that in the examined domain, perceived accuracy has a smaller persuasive power than other characteristics like novelty. Prediction accuracy, therefore, does not necessarily translate to economic success of a recommender.

Note that the result presentation phase is not the only situation during which the user is susceptible to persuasion. In their study, Gretzel and Fesenmaier [76] discovered that the likelihood of a user buying an item can already be increased during the elicitation phase. They found out that shorter and more transparent elicitation sessions led to a higher enjoyment and perceived fit of the recommendations later on. The predicted relevance in contrast played only a minor part in the persuasion process.

3.6 Proactive Recommendations—"When to Recommend?"

The final aspect related to the presentation of recommendations that we review in this section is mostly orthogonal to the previously discussed issues. It concerns the design of "proactive" recommender systems, which actively notify users of relevant content and thereby potentially disrupt other activities of the user.

In typical online recommendation scenarios we commonly assume (a) that there is a predefined visual area on the web pages where the recommended items are displayed and (b) that the system *always* tries to display some suitable recommendations when a user navigates to such a page. There are, however, a number of application scenarios where systems provide proactive recommendations, i.e., where the recommendations are not the immediate response to a user's (navigation) action. Examples include the delivery of personalized newsletters or text messages that contain item suggestions ("push notifications") or recommendation popups, e.g., within smartphone applications.

In these cases, the recommendation problem is not only to find items that are suited for the user's current situation, the problem is also to determine a suitable time to issue the recommendations. Furthermore, it has to be estimated how many of such proactive recommendations will be tolerated by the users. Finally, as not much screen space might be available for the recommendations, a careful design is required when deciding which details of an item should be displayed.

In the Decision-Making literature, "intrusive decision aid scenarios" have been the topic of study for some time. For example, Fitzsimons and Lehmann [57] conducted multiple user studies that highlight the increased risk of unsolicited advice in contrast to traditional decision aid scenarios. Specifically, they observed that participants who received advice that contradicted their prior attitudes not only ignored the advice but—as a form of *reactance*—were often even more likely to stick to their initially preferred alternative. Therefore, extra care has to be taken when selecting items because of this potential reactance to recommendations.

In the recommender systems literature, a few works exist which address the problem of proactive recommendations. Höpken et al. [84], for example, devised a system that complements a traditional recommendation architecture for a tourism application with an e-mail/text message notification service. The decision about the right time for recommendations in their approach is taken based on the user's current geographical position and the availability of a relevant item to recommend for a specific location. A similar system is presented by Wörndl et al. [194], who base their recommendation decision on a slightly more complex two-stage scoring system. First, they try to assess if users in their current situation surpass a certain "attention threshold." In case this can be assumed, they evaluate the relevance of the items and then decide if a proactive recommendation should be made.

In contrast, Lacerda et al. [115] employ an explore-exploit strategy with the aim of avoiding to send the same unhelpful recommendation to many users. In their daily deals recommendations scenario, they first test new deal recommendations with only a few users. Based on the feedback of these users they then decide if the items should be pushed to more users or not. Additionally, they evaluate different strategies to determine the users to select in the exploration phase, which should be those that (a) are likely to give feedback and (b) are representative for a group of other users.

The question of how much detailed information should be included in proactive recommendations has, for example, been explored by Bader et al. [13]. They design an explanation interface for an in-car recommendation scenario, where the problem is to find a configuration such that sufficient explanatory and persuasive information is provided to the driver without distracting them from the driving task. In another study, Sabic and Zanker [167] found out that the *nature of a notification* has a strong effect on the users' likelihood to become annoyed. In their study, smartphone users stated that proactive content updates, e.g., on breaking news, disturbed them far more than, for example, system notifications or text message alerts.

In practice, unsolicited personalized recommendations are common. Many smartphone apps, for example, proactively notify the user of breaking news or new content in the social network. The use of "content pop-overs" that point readers to additionally relevant items while they are browsing the site is also common today on many websites. Research in proactive recommendations, however, still seems to be comparably limited, despite the fact that providing recommendations proactively seems to be more risky than traditional recommendations, e.g., due to a possibly reactant user behavior. One possible reason for the limited amount of research on this topic could be that the evaluation of such proactive scenarios through traditional laboratory studies is challenging.

3.7 Discussion

In the second part of the article, we have reviewed various possible forms of supporting the user once the initial set of recommendations is determined, e.g., by allowing the user to inspect additional information or to give feedback on the recommendations. We have also discussed different design alternatives regarding the size of the recommendation list or the use of multiple lists and have presented alternative visualization approaches that support additional user interactions.

Our review shows that a number of proposals were made over the years in the research literature that do not consider the presentation of a ranked list of items as the end of the recommendation process. Generally, however, the amount of research on these topics seems vanishingly small when compared to the number of papers that are published on the problem of rating prediction and item ranking [99].

Different forms of user interactions are also supported in real-world applications. Some of them include comparably powerful forms of putting the user into control of the recommendations, as done in the case of Amazon.com. Others like Spotify or Netflix implement mechanisms for the user to interactively give feedback on the recommendations, to explore the item space, and to discover new items Steck et al. [215]. As for any system with a broad and heterogeneous user population, finding user interaction mechanisms that do not overwhelm a large fraction of the users, often remains an open issue.

A number of practical problems are barely discussed in the literature at all. Examples of such practical problems include the use of multiple lists and their organization on the screen. This research question is, for example, discussed by Gomez-Uribe and Hunt [71], who review various aspects of the current Netflix recommendation system. Another practical issue is that without randomization, contextualization, or short-term adaptation users will be presented with the same set of recommendations every time they visit the site. Not using randomization in the top ranks might not only lead to monotonous and boring recommendations but also to blockbuster effects where the "rich get richer" [58].

Another typical practical problem of media streaming providers like Netflix is that the choice of the movie to watch can be the result of a group decision process. Generally, limited research also exists on the design of user interfaces that support such group decisions. Existing research works on group recommendation like [144] or [37] include user interface enhancements to support the decision-making process, but only few works like [196] explicitly focus on questions of the user interface design and propose more elaborate interaction mechanisms.

Finally, a number of methodological challenges related to the evaluation of novel interaction mechanisms for recommender systems remain open. Some general frameworks to assess the user perception of recommender system have been proposed in the past years, including [153, 193] or [110]. Such general frameworks, however, cannot cover the particularities of specific interaction mechanisms. Assessing the level of persuasiveness of an explanation interface or the efficiency of critiquing-based approaches requires problem-specific evaluation procedures and measures. Different evaluation approaches exist for these specific problems in the literature, but for many problems no agreed-upon standards exist today.

Table 4 summarizes some key research challenges that were identified in this section in the context of interaction mechanisms for enhanced result presentation, explanations, and user feedback.

4 CONCLUSIONS

The field of recommender systems has its roots in various domains including Information Filtering and Information Retrieval, Human-Computer Interaction (HCI), and Artificial Intelligence. However, when we look at the research landscape in particular in the years after the Netflix Prize, we see that the field is dominated by technical approaches that aim to optimize abstract accuracy metrics like the RMSE or Precision and Recall [92].

This focus of the field on accuracy measures has already been considered as potentially harmful before the Netflix Prize [132], and that the community should put more emphasis on the user experience of recommender systems has also been advocated more recently, for example, in [113] and [97]. Clearly, research on HCI aspects can be more challenging from a methodological perspective[6]

[6]Typical problems include questions of reproducibility, the validity of the experimental designs, the robustness of the applied statistical methods, and the generalizability of the observations.

Table 4. Selected Challenges in the Area of Result Presentation and User Feedback

Category	Specific challenges
List design	Determining the optimal choice set size for a given user and application domain
	Organizing user interfaces with multiple recommendation lists
	Avoiding boredom and creating diversified lists, e.g., by shuffling or re-ranking top-*n* item lists
Visualization	Helping users understand relationships between items (and other users) through interactive visualizations
	Designing easy-to-comprehend visualization approaches that can be integrated into real-world systems
Explanations	Explaining the differences between choices to the users
	Generating interpretable and persuasive explanations from complex machine-learning models
User control	Allowing users to give feedback on the recommendations in an intuitive way
Timing	Deciding when to recommend in proactive recommendation scenarios
Methodology	Developing standardized evaluation methodologies for novel interaction mechanisms

than optimizing an abstract performance measure and assessing the outcomes with commonly agreed evaluation procedures. Nonetheless, as more and more reports by companies are published that emphasize the importance of user experience aspects—including the recent work by Gomez-Uribe and Hunt [71] on the Netflix recommender—we hope that the already existing strands in HCI research on RS will be continued in the future.

In addition to the tenet that more emphasis should be put on the user experience of recommender systems [113], we argue that the intended utility of the recommender should be kept in mind when designing the user interaction mechanisms for the system. This, in turn, might require the consideration of domain-specific and application-specific solutions and probably in many cases a multi-disciplinary research approach that goes beyond the algorithmic perspective of selecting and ranking items.

REFERENCES

[1] Gediminas Adomavicius, Jesse C. Bockstedt, Shawn Curley, and Jingjing Zhang. 2012. Effects of online recommendations on consumers' willingness to pay. In *Proceedings of the 2nd Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'12)*. 40–45.

[2] Gediminas Adomavicius, Jesse C. Bockstedt, Shawn Curley, and Jingjing Zhang. 2013. Do recommender systems manipulate consumer preferences? A study of anchoring effects. *Information Systems Research* 24, 4 (2013), 956–975.

[3] Gediminas Adomavicius and YoungOk Kwon. 2007. New recommendation techniques for multicriteria rating systems. *Intelligent Systems* 22, 3 (2007), 48–55.

[4] Gediminas Adomavicius and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *Transactions on Knowledge and Data Engineering* 17, 6 (2005), 734–749.

[5] Muhammad Aljukhadar, Sylvain Senecal, and Charles-Etienne Daoust. 2012. Using recommendation agents to cope with information overload. *International Journal of Electronic Commerce* 17, 2 (2012), 41–70.

[6] Xavier Amatriain, Josep M. Pujol, and Nuria Oliver. 2009. I like it... I like it not: Evaluating user ratings noise in recommender systems. In *Proceedings of the 17th International Conference on User Modeling, Adaptation, and Personalization (UMAP'09)*. 247–258.

[7] Xavier Amatriain, Josep M. Pujol, Nava Tintarev, and Nuria Oliver. 2009. Rate it again: Increasing recommendation accuracy by user re-rating. In *Proceedings of the 3rd Conference on Recommender Systems (RecSys'09)*. 173–180.

[8] Mohammad Shafkat Amin, Baoshi Yan, Sripad Sriram, Anmol Bhasin, and Christian Posse. 2012. Social referral: Leveraging network connections to deliver recommendations. In *Proceedings of the 6th Conference on Recommender Systems (RecSys'12)*. 273–276.

[9] Taiwo Amoo and Hershey H. Friedman. 2000. Overall evaluation rating scales: An assessment. *International Journal of Market Research* 42, 3 (2000), 301–310.

[10] Taiwo Amoo and Hershey H. Friedman. 2001. Do numeric values influence subjects' responses to rating scales? *Journal of International Marketing and Marketing Research* 26 (2001), 41–46.

[11] Sarabjot Singh Anand and Nathan Griffiths. 2011. A market-based approach to address the new item problem. In *Proceedings of the 5th Conference on Recommender Systems (RecSys'11)*. 205–212.

[12] Ivana Andjelkovic, Denis Parra, and John O'Donovan. 2016. Moodplay: Interactive mood-based music discovery and recommendation. In *Proceedings of the 24th Conference on User Modeling, Adaptation, and Personalization (UMAP'16)*. 275–279.

[13] Roland Bader, Wolfgang Wörndl, Andreas Karitnig, and Gerhard Leitner. 2011. Designing an explanation interface for proactive recommendations in automotive scenarios. In *Proceedings of the Workshop on Advances in User Modeling at the 19th Conference on User Modeling, Adaptation, and Personalization (UMAP'11)*. 92–104.

[14] Sam Banks, Rachael Rafter, and Barry Smyth. 2015. The recommendation game: Using a game-with-a-purpose to generate recommendation data. In *Proceedings of the 9th Conference on Recommender Systems (RecSys'15)*. 305–308.

[15] Patrick Baudisch and Lars Brueckner. 2002. TV scout: Lowering the entry barrier to personalized TV program recommendation. In *Proceedings of the 2nd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'02)*. 58–68.

[16] Claudia Becerra, Fabio Gonzalez, and Alexander Gelbukh. 2011. Visualizable and explicable recommendations obtained from price estimation functions. In *Joint Proceedings of the RecSys 2011 Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'11) and User-Centric Evaluation of Recommender Systems and Their Interfaces-2 (UCERSTI 2) affiliated with the 5th Conference on Recommender Systems (RecSys'11)*. 27–34.

[17] Shay Ben-Elazar and Noam Koenigstein. 2014. A hybrid explanations framework for collaborative filtering recommender systems. In *Poster Proceedings of the 8th Conference on Recommender Systems (RecSys'14)*.

[18] Nofar Dali Betzalel, Bracha Shapira, and Lior Rokach. 2015. "Please, not now!": A model for timing recommendations. In *Proceedings of the 9th Conference on Recommender Systems (RecSys'15)*. 297–300.

[19] Mustafa Bilgic and Raymond Mooney. 2005. Explaining recommendations: Satisfaction vs. promotion. In *Proceedings of Beyond Personalization 2005: A Workshop on the Next Stage of Recommender Systems Research at the 10th International Conference on Intelligent User Interfaces (IUI'05)*. 13–18.

[20] Dirk G. F. M. Bollen, Mark P. Graus, and Martijn C. Willemsen. 2012. Remembering the stars?: Effect of time on preference retrieval from memory. In *Proceedings of the 6th Conference on Recommender Systems (RecSys'12)*. 217–220.

[21] Dirk G. F. M. Bollen, Bart P. Knijnenburg, Martijn C. Willemsen, and Mark P. Graus. 2010. Understanding choice overload in recommender systems. In *Proceedings of the 4th Conference on Recommender Systems (RecSys'10)*. 63–70.

[22] Svetlin Bostandjiev, John O'Donovan, and Tobias Höllerer. 2012. TasteWeights: A visual interactive hybrid recommender system. In *Proceedings of the 4th Conference on Recommender Systems (RecSys'10)*. 35–42.

[23] Derek G. Bridge. 2002. Towards conversational recommender systems: A dialogue grammar approach. In *Proceedings of the 6th European Conference on Case Based Reasoning (ECCBR'02)*. 9–22.

[24] Robin Burke. 2000. Knowledge-based recommender systems. *Encyclopedia of Library and Information Science* 69, 32 (2000), 180–200.

[25] Robin D. Burke, Kristian J. Hammond, and Benjamin C. Young. 1996. Knowledge-based navigation of complex information spaces. In *Proceedings of the 13th National Conference on Artificial Intelligence (AAAI'96)*. 462–468.

[26] Duen Horng Chau, Aniket Kittur, Jason I. Hong, and Christos Faloutsos. 2011. Apolo: Interactive large graph sensemaking by combining machine learning and visualization. In *Proceedings of the 17th SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'11)*. 739–742.

[27] Li Chen. 2009. Adaptive tradeoff explanations in conversational recommenders. In *Proceedings of the 3rd Conference on Recommender Systems (RecSys'09)*. 225–228.

[28] Li Chen and Pearl Pu. 2006. Evaluating critiquing-based recommender agents. In *Proceedings of the 21st National Conference on Artificial Intelligence (AAAI'06)*. 157–162.

[29] Li Chen and Pearl Pu. 2008. A cross-cultural user evaluation of product recommender interfaces. In *Proceedings of the 2nd Conference on Recommender Systems (RecSys'08)*. 75–82.

[30] Li Chen and Ho Keung Tsoi. 2011. Users' decision behavior in recommender interfaces: Impact of layout design. In *Joint Proceedings of the RecSys 2011 Workshop on Human Decision Making in Recommender Systems*

(Decisions@RecSys'11) and User-Centric Evaluation of Recommender Systems and Their Interfaces-2 (UCERSTI 2) affiliated with the 5th Conference on Recommender Systems (RecSys'11). 21–26.

[31] Yu Chen and Pearl Pu. 2012. CoFeel: An interface for providing emotional feedback in mobile group recommender systems. In *Joint Proceedings of the 1st International Workshop on Recommendation Technologies for Lifesytle Change (LIFESTYLE'12) and the 1st International Workshop on Interfaces for Recommender Systems (InterfaceRS'12).* 48–55.

[32] Dan Cosley, Shyong K. Lam, Istvan Albert, Joseph A. Konstan, and John Riedl. 2003. Is seeing believing?: How recommender system interfaces affect users' opinions. In *Proceedings of the 2003 SIGCHI Conference on Human Factors in Computing Systems (CHI'03).* 585–592.

[33] Paolo Cremonesi, Antonio Donatacci, Franca Garzotto, and Roberto Turrin. 2012. Decision-making in recommender systems: The role of user's goals and bounded resources. In *Proceedings of the 2nd Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'12).* 1–7.

[34] Paolo Cremonesi, Franca Garzotto, and Roberto Turrin. 2012. Investigating the persuasion potential of recommender systems from a quality perspective: An empirical study. *Transactions on Interactive Intelligent Systems* 2, 2 (2012), 11:1–11:41.

[35] Paolo Cremonesi, Franca Garzottto, and Roberto Turrin. 2012. User effort vs. accuracy in rating-based elicitation. In *Proceedings of the 6th Conference on Recommender Systems (RecSys'12).* 27–34.

[36] Elizabeth M. Daly, Adi Botea, Akihiro Kishimoto, and Radu Marinescu. 2014. Multi-criteria journey aware housing recommender system. In *Proceedings of the 8th Conference on Recommender Systems (RecSys'14).* 325–328.

[37] Toon De Pessemier, Simon Dooms, and Luc Martens. 2012. Design and evaluation of a group recommender system. In *Proceedings of the 6th Conference on Recommender Systems (RecSys'12).* 225–228.

[38] Lihi Naamani Dery, Meir Kalech, Lior Rokach, and Bracha Shapira. 2014. Preference elicitation for narrowing the recommended list for groups. In *Proceedings of the 8th Conference on Recommender Systems (RecSys'14).* 333–336.

[39] Ravi Dhar and Itamar Simonson. 2003. The effect of forced choice on choice. *Journal of Marketing Research* 40, 2 (2003), 146–160.

[40] Cecilia di Sciascio, Vedran Sabol, and Eduardo E. Veas. 2015. *uRank*: Exploring document recommendations through an interactive user-driven approach. In *Proceedings of the Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS'15) co-located with the 9th Conference on Recommender Systems (RecSys'15).* 29–36.

[41] Kristin Diehl and Cait Poynor. 2010. Great expectations?! Assortment size, expectations, and satisfaction. *Journal of Marketing Research* 47, 2 (2010), 312–322. arXiv:http://dx.doi.org/10.1509/jmkr.47.2.312

[42] Kristin Diehl, Erica van Herpen, and Cait Lamberton. 2015. Organizing products with complements versus substitutes: Effects on store preferences as a function of effort and assortment perceptions. *Journal of Retailing* 91, 1 (2015), 1–18.

[43] Tim Donkers, Benedikt Loepp, and Jürgen Ziegler. 2015. Merging latent factors and tags to increase interactive control of recommendations. In *Poster Proceedings of the 9th Conference on Recommender Systems (RecSys'15).*

[44] Simon Dooms, Toon De Pessemier, and Luc Martens. 2014. Improving IMDb movie recommendations with interactive settings and filters. In *Poster Proceedings of the 8th Conference on Recommender Systems (RecSys'14).*

[45] Sara Drenner, Shilad Sen, and Loren G. Terveen. 2008. Crafting the initial user experience to achieve community goals. In *Proceedings of the 2nd Conference on Recommender Systems (RecSys'08).* 187–194.

[46] Greg Dunn, Jurgen Wiersema, Jaap Ham, and Lora Aroyo. 2009. Evaluating interface variants on personality acquisition for recommender systems. In *Proceedings of the 17th International Conference on User Modeling, Adaptation, and Personalization (UMAP'09).* 259–270.

[47] Michael D. Ekstrand, Daniel Kluver, F. Maxwell Harper, and Joseph A. Konstan. 2015. Letting users choose recommender algorithms: An experimental study. In *Proceedings of the 9th Conference on Recommender Systems (RecSys'15).* 11–18.

[48] Mehdi Elahi, Matthias Braunhofer, Francesco Ricci, and Marko Tkalcic. 2013. Personality-based active learning for collaborative filtering recommender systems. In *Proceedings of the 25th International Conference of the Italian Association for Artificial Intelligence (AI*IA'13).* 360–371.

[49] Mehdi Elahi, Mouzhi Ge, Francesco Ricci, Ignacio Fernández-Tobías, Shlomo Berkovski, and David Massimo. 2015. Interaction design in a mobile food recommender system. In *Proceedings of the Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS'15) Co-located with the 9th Conference on Recommender Systems (RecSys'15).* 49–52.

[50] Mehdi Elahi, Francesco Ricci, and Neil Rubens. 2013. Active learning strategies for rating elicitation in collaborative filtering: A system-wide perspective. *Transactions on Intelligent Systems and Technology* 5, 1 (2013), 13:1–13:33.

[51] Barbara Fasolo, Ralph Hertwig, Michaela Huber, and Mark Ludwig. 2009. Size, entropy, and density: What is the difference that makes the difference between small and large real-world assortments? *Psychology & Marketing* 26, 3 (2009), 254–279.

[52] Alexander Felfernig, Gerhard Friedrich, Klaus Isak, Kostyantyn Shchekotykhin, Erich Teppan, and Dietmar Jannach. 2009. Automated debugging of recommender user interface descriptions. *Applied Intelligence* 31, 1 (2009), 1–14.

[53] Alexander Felfernig, Gerhard Friedrich, Dietmar Jannach, and Markus Zanker. 2007. An integrated environment for the development of knowledge-based recommender applications. *International Journal of Electronic Commerce* 11, 2 (2007), 11–34.

[54] Alexander Felfernig, Gerhard Friedrich, Dietmar Jannach, and Markus Zanker. 2011. Developing constraint-based recommenders. In *Recommender Systems Handbook* (1st ed.), Francesco Ricci et al. (Ed.). Springer, 187–215. Springer.

[55] Alexander Felfernig and Bartosz Gula. 2006. An empirical study on consumer behavior in the interaction with knowledge-based recommender applications. In *Proceedings of the 8th International Conference on E-Commerce Technology (CEC'06)/3rd International Conference on Enterprise Computing, E-Commerce and E-Services (EEE'06)*. 37.

[56] Alexander Felfernig, Bartosz Gula, Gerhard Leitner, Marco Maier, Rudolf Melcher, and Erich Teppan. 2008. Persuasion in knowledge-based recommendation. In *Persuasive Technology*. Lecture Notes in Computer Science, Vol. 5033. Springer, 71–82.

[57] Gavan J. Fitzsimons and Donald R. Lehmann. 2004. Reactance to recommendations: When unsolicited advice yields contrary responses. *Marketing Science* 23, 1 (2004), 82–94.

[58] Daniel Fleder and Kartik Hosanagar. 2009. Blockbuster culture's next rise or fall: The impact of recommender systems on sales diversity. *Management Science* 55, 5 (2009), 697–712.

[59] Brian J. Fogg. 1998. Persuasive computers: Perspectives and research directions. In *Proceedings of the 1998 SIGCHI Conference on Human Factors in Computing Systems (CHI'98)*. 225–232.

[60] Hershey H. Friedman and Taiwo Amoo. 1999. Rating the rating scales. *Journal of Marketing Management* 9, 3 (1999), 114–123.

[61] Matthias Fuchs and Markus Zanker. 2012. Multi-criteria ratings for recommender systems: An empirical analysis in the tourism domain. In *Proceedings of the 13th International Conference on E-Commerce and Web Technologies (EC-Web'12)*. 100–111.

[62] Gayatree Ganu, Noemie Elhadad, and Amélie Marian. 2009. Beyond the stars: Improving rating predictions using review text content. In *Proceedings of the 12th International Workshop on the Web and Databases (WebDB'09)*.

[63] Rui Gao, Bibo Hao, Shuotian Bai, Lin Li, Ang Li, and Tingshao Zhu. 2013. Improving user profile with personality traits predicted from social media content. In *Proceedings of the 7th Conference on Recommender Systems (RecSys'13)*. 355–358.

[64] Damianos Gavalas, Charalampos Konstantopoulos, Konstantinos Mastakas, and Grammati Pantziou. 2014. Mobile recommender systems in tourism. *Journal of Network and Computer Applications* 39 (2014), 319–333.

[65] Mouzhi Ge, Dietmar Jannach, Fatih Gedikli, and Martin Hepp. 2012. Effects of the placement of diverse items in recommendation lists. In *Proceedings of the 14th International Conference on Enterprise Information Systems (ICEIS'12)*. 201–208.

[66] Fatih Gedikli and Dietmar Jannach. 2013. Improving recommendation accuracy based on item-specific tag preferences. *Transactions on Intelligent Systems and Technology* 4, 1 (2013), 11:1–11:19.

[67] Fatih Gedikli, Dietmar Jannach, and Mouzhi Ge. 2014. How should I explain? A comparison of different explanation types for recommender systems. *International Journal of Human-Computer Studies* 72, 4 (2014), 367–382.

[68] Sofia Gkika and George Lekakos. 2014. The persuasive role of explanations in recommender systems. In *Proceedings of the 2nd International Workshop on Behavior Change Support Systems (BCSS'14)*. 59–68.

[69] Mehmet Göker and Cynthia Thompson. 2000. The adaptive place advisor: A conversational recommendation system. In *Proceedings of the 8th German Workshop on Case Based Reasoning*. 187–198.

[70] Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins. 2001. Eigentaste: A constant time collaborative filtering algorithm. *Information Retrieval* 4, 2 (2001), 133–151.

[71] Carlos A. Gomez-Uribe and Neil Hunt. 2015. The Netflix recommender system: Algorithms, business value, and innovation. *Transactions on Management Information Systems* 6, 4 (2015), 13:1–13:19.

[72] Eduardo Graells-Garrido, Mounia Lalmas, and Ricardo Baeza-Yates. 2016. Data portraits and intermediary topics: Encouraging exploration of politically diverse profiles. In *Proceedings of the 21st International Conference on Intelligent User Interfaces (IUI'16)*. 228–240.

[73] Peter Grasch, Alexander Felfernig, and Florian Reinfrank. 2013. ReComment: Towards critiquing-based recommendation with speech interaction. In *Proceedings of the 7th Conference on Recommender Systems (RecSys'13)*. 157–164.

[74] Shirley Gregor and Izak Benbasat. 1999. Explanations from intelligent systems: Theoretical foundations and implications for practice. *MIS Quarterly* 23, 4 (1999), 497–530.

[75] Brynjar Gretarsson, John O'Donovan, Svetlin Bostandjiev, Christopher Hall, and Tobias Höllerer. 2010. SmallWorlds: Visualizing social recommendations. *Computer Graphics Forum* 29, 3 (2010), 833–842.

[76] Ulrike Gretzel and Daniel R. Fesenmaier. 2006. Persuasion in recommender systems. *International Journal of Electronic Commerce* 11, 2 (2006), 81–100.

[77] Francesca Guzzi, Francesco Ricci, and Robin D. Burke. 2011. Interactive multi-party critiquing for group recommendation. In *Proceedings of the 5th Conference on Recommender Systems (RecSys'11)*. 265–268.

[78] F. Maxwell Harper, Funing Xu, Harmanpreet Kaur, Kyle Condiff, Shuo Chang, and Loren G. Terveen. 2015. Putting users in control of their recommendations. In *Proceedings of the 9th Conference on Recommender Systems (RecSys'15)*. 3–10.

[79] Chen He, Denis Parra, and Katrien Verbert. 2016. Interactive recommender systems: A survey of the state of the art and future research challenges and opportunities. *Expert Systems with Applications* 56 (2016), 9–27.

[80] Jonathan L. Herlocker, Joseph A. Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the 2000 Conference on Computer Supported Cooperative Work (CSCW'00)*. 241–250.

[81] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *Transactions on Information Systems* 22, 1 (2004), 5–53.

[82] Yoshinori Hijikata, Yuki Kai, and Shogo Nishida. 2012. The relation between user intervention and user satisfaction for information recommendation. In *Proceedings of the 27th Annual Symposium on Applied Computing (SAC'12)*. 2002–2007.

[83] Daniel E. Ho and Kevin M. Quinn. 2008. Improving the presentation and interpretation of online ratings data with model-based figures. *The American Statistician* 62, 4 (2008), 279–288.

[84] Wolfram Höpken, Matthias Fuchs, Markus Zanker, and Thomas Beer. 2010. Context-based adaptation of mobile applications in tourism. *Information Technology & Tourism* 12, 2 (2010), 175–195.

[85] Rong Hu and Pearl Pu. 2010. A study on user perception of personality-based recommender systems. In *Proceedings of the 18th International Conference on User Modeling, Adaptation, and Personalization (UMAP'10)*. 291–302.

[86] Jon Espen Ingvaldsen, Jon Atle Gulla, and Özlem Özgöbek. 2015. User controlled news recommendations. In *Proceedings of the Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS'15) Co-located with the 9th Conference on Recommender Systems (RecSys'15)*. 45–48.

[87] Sheena S. Iyengar and Mark R. Lepper. 2000. When choice is demotivating: Can one desire too much of a good thing? *Journal of Personality and Social Psychology* 79, 6 (2000), 995–1006.

[88] Niklas Jakob, Stefan Hagen Weber, Mark-Christoph Müller, and Iryna Gurevych. 2009. Beyond the stars: Exploiting free-text user reviews to improve the accuracy of movie recommendations. In *Proceedings of the 1st International Workshop on Topic-Sentiment Analysis for Mass Opinion at the 2009 Conference on Information and Knowledge Management (CIKM'09)*. 57–64.

[89] Anthony Jameson, Silvia Gabrielli, Per Ola Kristensson, Katharina Reinecke, Federica Cena, Cristina Gena, and Fabiana Vernero. 2011. How can we support users' preferential choice?. In *Extended Abstracts of the 2011 Conference on Human Factors in Computing Systems (CHI'11)*. 409–418.

[90] Dietmar Jannach. 2004. Preference-based treatment of empty result sets in product finders and knowledge-based recommenders. In *Proceedings of the 27th Annual Conference on Artificial Intelligence (AI'04)*. 145–159.

[91] Dietmar Jannach. 2006. Finding preferred query relaxations in content-based recommenders. In *Proceedings of the 3rd International Conference on Intelligent Systems (IS'06)*. 355–360.

[92] Dietmar Jannach and Gedas Adomavicius. 2016. Recommendations with a purpose. In *Proceedings of the 10th Conference on Recommender Systems (RecSys'16)*. 7–10.

[93] Dietmar Jannach and Gerold Kreutler. 2005. Personalized user preference elicitation for e-services. In *Proceedings of the 2005 International Conference on e-Technology, e-Commerce, and e-Service (EEE'05)*. 604–611.

[94] Dietmar Jannach and Gerold Kreutler. 2007. Rapid development of knowledge-based conversational recommender applications with advisor suite. *Journal of Web Engineering* 6, 2 (2007), 165–192.

[95] Dietmar Jannach, Lukas Lerche, and Michael Jugovac. 2015. Item familiarity as a possible confounding factor in user-centric recommender systems evaluation. *i-com Journal for Interactive Media* 14, 1 (2015), 29–40.

[96] Dietmar Jannach, Lukas Lerche, and Markus Zanker. 2017. Social information access. In *"Recommending based on Implicit Feedback" (forthcoming)*, Peter Brusilovskyand Daqing He (Eds.). Springer.

[97] Dietmar Jannach, Paul Resnick, Alexander Tuzhilin, and Markus Zanker. 2016. Recommender systems—Beyond matrix completion. *Communications of the ACM* 59, 11 (2016).

[98] Dietmar Jannach, Markus Zanker, and Matthias Fuchs. 2014. Leveraging multi-criteria customer feedback for satisfaction analysis and improved recommendations. *Information Technology & Tourism* 14, 2 (2014), 119–149.

[99] Dietmar Jannach, Markus Zanker, Mouzhi Ge, and Marian Gröning. 2012. Recommender systems in computer science and information systems—A landscape of research. In *Proceedings of the 13th International Conference on E-Commerce and Web Technologies (EC-WEB'12)*. 76–87.

[100] Zhenhui Jiang, Weiquan Wang, and Izak Benbasat. 2005. Multimedia-based interactive advising technology for online consumer decision support. *Communications of the ACM* 48, 9 (2005), 92–98.

[101] Rong Jin and Luo Si. 2004. A Bayesian approach toward active learning for collaborative filtering. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence (UAI'04)*. 278–285.

[102] Thorsten Joachims, Laura Granka, Bing Pan, Helene Hembrooke, and Geri Gay. 2005. Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the 28th Annual International SIGIR Conference on Research and Development in Information Retrieval (SIGIR'05)*. 154–161.

[103] Hermann Kaindl, Elmar P. Wach, Ada Okoli, Roman Popp, Ralph Hoch, Werner Gaulke, and Tim Hussein. 2013. Semi-automatic generation of recommendation processes and their GUIs. In *Proceedings of the 18th International Conference on Intelligent User Interfaces (IUI'13)*. 85–94.

[104] Rasoul Karimi, Christoph Freudenthaler, Alexandros Nanopoulos, and Lars Schmidt-Thieme. 2012. Exploiting the characteristics of matrix factorization for active learning in recommender systems. In *Proceedings of the 6th Conference on Recommender Systems (RecSys'12)*. 317–320.

[105] Kristiina Karvonen, Sanna Shibasaki, Sofia Nunes, Puneet Kaur, and Olli Immonen. 2010. Visual nudges for enhancing the use and produce of reputation information. In *Proceedings of the Workshop on User-Centric Evaluation of Recommender Systems and Their Interfaces at the 4th Conference on Recommender Systems (RecSys'10)*. 1–8.

[106] Diane Kelly and Nicholas J. Belkin. 2001. Reading time, scrolling and interaction: Exploring implicit sources of user preferences for relevance feedback. In *Proceedings of the 24th Annual International SIGIR Conference on Research and Development in Information Retrieval (SIGIR'01)*. 408–409.

[107] Diane Kelly and Jaime Teevan. 2003. Implicit feedback for inferring user preference: A bibliography. *SIGIR Forum* 37, 2 (2003), 18–28.

[108] Noreen M. Klein and Manjit S. Yadav. 1989. Context effects on effort and accuracy in choice: An enquiry into adaptive decision making. *Journal of Consumer Research* 15, 4 (1989), 411–421.

[109] Daniel Kluver, Tien T. Nguyen, Michael D. Ekstrand, Shilad Sen, and John Riedl. 2012. How many bits per rating?. In *Proceedings of the 6th Conference on Recommender Systems (RecSys'12)*. 99–106.

[110] Bart P. Knijnenburg, Svetlin Bostandjiev, John O'Donovan, and Alfred Kobsa. 2012. Inspectability and control in social recommenders. In *Proceedings of the 6th Conference on Recommender Systems (RecSys'12)*. 43–50.

[111] Bart P. Knijnenburg, Niels J. M. Reijmer, and Martijn C. Willemsen. 2011. Each to his own: How different users call for different interaction methods in recommender systems. In *Proceedings of the 5th Conference on Recommender Systems (RecSys'11)*. 141–148.

[112] Alfred Kobsa, Jürgen Koenemann, and Wolfgang Pohl. 2001. Personalised hypermedia presentation techniques for improving online customer relationships. *The Knowledge Engineering Review* 16, 2 (2001), 111–155.

[113] Joseph Konstan and John Riedl. 2012. Recommender systems: From algorithms to user experience. *User Modeling and User-Adapted Interaction* 22, 1 (2012), 101–123.

[114] Johannes Kunkel, Benedikt Loepp, and Jürgen Ziegler. 2015. 3D-visualisierung zur eingabe von Präferenzen in empfehlungssystemen. In *Proceedings of the 2015 Conference on Mensch und Computer 2015*. De Gruyter Oldenbourg, 123–132.

[115] Anísio Lacerda, Adriano Veloso, and Nivio Ziviani. 2013. Exploratory and interactive daily deals recommendation. In *Proceedings of the 7th Conference on Recommender Systems (RecSys'13)*. 439–442.

[116] Joseph Lajos, Amitava Chattopadhyay, and Kishore Sengupta. 2009. When electronic recommendation agents backfire: Negative effects on choice satisfaction, attitudes, and purchase intentions. *Advances in Consumer Research* 36 (2009), 845–846.

[117] Béatrice Lamche, Ugur Adigüzel, and Wolfgang Wörndl. 2014. Interactive explanations in mobile shopping recommender systems. In *Proceedings of the Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS'14) co-located with the 8th Conference on Recommender Systems (RecSys'14)*. 14–21.

[118] Béatrice Lamche, Enrico Pollok, Wolfgang Wörndl, and Georg Groh. 2014. Evaluating the effectiveness of stereotype user models for recommendations on mobile devices. In *Proceedings of the 4th International Workshop on Personalization Approaches in Learning Environments (PALE'14), held in conjunction with the 22nd International Conference on User Modeling, Adaptation, and Personalization (UMAP'14)*. 33–41.

[119] Béatrice Lamche, Yannick Rödl, Claudius Hauptmann, and Wolfgang Wörndl. 2015. Context-aware recommendations for mobile shopping. In *Proceedings of the Workshop on Location-Aware Recommendations (LocalRec'15) co-located with the 9th Conference on Recommender Systems (RecSys'15)*. 21–27.

[120] Danielle Hyunsook Lee. 2008. PITTCULT: Trust-based cultural event recommender. In *Proceedings of the 2nd Conference on Recommender Systems (RecSys'08)*. 311–314.

[121] Wei-Po Lee. 2004. Towards agent-based decision making in the electronic marketplace: Interactive recommendation and automated negotiation. *Expert Systems with Applications* 27, 4 (2004), 665–679.

[122] Cane Wing-ki Leung, Stephen Chi-fai Chan, and Fu-lai Chung. 2006. Integrating collaborative filtering and sentiment analysis: A rating inference approach. In *Proceedings of the 2006 Workshop on Recommender Systems in Conjunction with the 17th European Conference on Artificial Intelligence (ECAI'06)*. 62–66.

[123] Greg Linden, Steve Hanks, and Neal Lesh. 1997. Interactive assessment of user preference models: The automated travel assistant. In *Proceedings of the 6th International Conference on User Modeling (UM'97)*, Vol. 383. 67–78.

[124] Benedikt Loepp, Tim Hussein, and Jürgen Ziegler. 2014. Choice-based preference elicitation for collaborative filtering recommender systems. In *Proceedings of the 2014 SIGCHI Conference on Human Factors in Computing Systems (CHI'14)*. 3085–3094.

[125] Tariq Mahmood, Francesco Ricci, and Adriano Venturini. 2009. Improving recommendation effectiveness: Adapting a dialogue strategy in online travel planning. *Information Technology & Tourism* 11, 4 (2009), 285–302.

[126] Benjamin M. Marlin, Richard S. Zemel, Sam T. Roweis, and Malcolm Slaney. 2007. Collaborative filtering and the missing at random assumption. In *Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence (UAI'07)*. 267–275.

[127] Riccardo Mazza. 2009. *Introduction to Information Visualization* (1st ed.). Springer.

[128] Kevin McCarthy, James Reilly, Lorraine McGinty, and Barry Smyth. 2004. On the dynamic generation of compound critiques in conversational recommender systems. In *Proceedings of the 3rd International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'04)*. 176–184.

[129] Kevin McCarthy, James Reilly, Lorraine McGinty, and Barry Smyth. 2004. Thinking positively—Explanatory feedback for conversational recommender systems. In *Proceedings of the Explanation Workshop at the European Conference on Case-Based Reasoning (ECCBR'04)*. 115–124.

[130] Robert McCrae and Paul Costa. 1996. Toward a new generation of personality theories: Theoretical contexts for the five-factor model. In *The Five-Factor Model of Personality: Theoretical Perspectives*, Jerry S. Wiggins (Ed.). Guildford Press, 51–87. Guildford Press.

[131] Sean M. McNee, Shyong K. Lam, Joseph A. Konstan, and John Riedl. 2003. Interfaces for eliciting new user preferences in recommender systems. In *Proceedings of the 9th International Conference on User Modeling (UM'03)*. 178–187.

[132] Sean M. McNee, John Riedl, and Jospeh A. Konstan. 2006. Being accurate is not enough: How accuracy metrics have hurt recommender systems. In *Extended Abstracts of the 2006 Conference on Human Factors in Computing Systems (CHI'06)*. 1097–1101.

[133] David McSherry. 2004. Incremental relaxation of unsuccessful queries. In *Proceedings of the 7th European Conference on Case-Based Reasoning (ECCBR'04)*. 331–345.

[134] Amine Naak, Hicham Hage, and Esma Ameur. 2009. A multi-criteria collaborative filtering approach for research paper recommendation in papyres. In *E-Technologies: Innovation in an Open World*. Lecture Notes in Business Information Processing, Vol. 26. Springer, 25–39.

[135] Theodora Nanou, George Lekakos, and Konstantinos G. Fouskas. 2010. The effects of recommendations' presentation on persuasion and satisfaction in a movie recommender system. *Multimedia Systems* 16, 4 (2010), 219–230.

[136] Dario De Nart, Felice Ferrara, and Carlo Tasso. 2013. Personalized access to scientific publications: From recommendation to explanation. In *Proceedings of the 21st International Conference on User Modeling, Adaptation, and Personalization (UMAP'13)*. 296–301.

[137] Dario De Nart and Carlo Tasso. 2014. A personalized concept-driven recommender system for scientific libraries. In *Proceedings of the 10th Italian Research Conference on Digital Libraries (IRCDL'14)*. 84–91.

[138] Julia Neidhardt, Leonhard Seyfang, Rainer Schuster, and Hannes Werthner. 2015. A picture-based approach to recommender systems. *Information Technology & Tourism* 15, 1 (2015), 49–69.

[139] Quang Nhat Nguyen and Francesco Ricci. 2007. Replaying live-user interactions in the off-line evaluation of critique-based mobile recommendations. In *Proceedings of the 7th Conference on Recommender Systems (RecSys'13)*. 81–88.

[140] Tien T. Nguyen, Daniel Kluver, Ting-Yu Wang, Pik-Mai Hui, Michael D. Ekstrand, Martijn C. Willemsen, and John Riedl. 2013. Rating support interfaces to improve user experience and recommender accuracy. In *Proceedings of the 7th Conference on Recommender Systems (RecSys'13)*. 149–156.

[141] Stephen M. Nowlis, Barbara E. Kahn, and Ravi Dhar. 2002. Coping with ambivalence: The effect of removing a neutral option on consumer attitude and preference judgments. *Journal of Consumer Research* 29, 3 (2002), 319–334.

[142] Maria Augusta S. N. Nunes and Rong Hu. 2012. Personality-based recommender systems: An overview. In *Proceedings of the 6th Conference on Recommender Systems (Recsys'12)*. 5–6.

[143] Douglas W. Oard and Jinmook Kim. 1998. Implicit feedback for recommender systems. In *Proceedings of the 1998 AAAI Workshop on Recommender Systems*. 81–83.

[144] Mark O'Connor, Dan Cosley, Joseph A. Konstan, and John Riedl. 2001. PolyLens: A recommender system for groups of users. In *Proceedings of the 7th Conference on European Conference on Computer Supported Cooperative Work (ECSCW'01)*. 199–218.

[145] John O'Donovan, Barry Smyth, Brynjar Gretarsson, Svetlin Bostandjiev, and Tobias Höllerer. 2008. PeerChooser: Visual interactive recommendation. In *Proceedings of the 2008 SIGCHI Conference on Human Factors in Computing Systems (CHI'08)*. 1085–1088.

[146] Denis Parra and Peter Brusilovsky. 2015. User-controllable personalization: A case study with setfusion. *International Journal of Human-Computer Studies* 78 (2015), 43–67.

[147] Denis Parra, Peter Brusilovsky, and Christoph Trattner. 2014. See what you want to see: Visual user-driven approach for hybrid recommendation. In *Proceedings of the 19th International Conference on Intelligent User Interfaces (IUI'14).* 235–240.

[148] Evelien Perik, Boris de Ruyter, Panos Markopoulos, and Berry Eggen. 2004. The sensitivities of user profile information in music recommender systems. In *Proceedings of the 2nd Annual Conference on Privacy, Security, and Trust (PST'04).* 137–141.

[149] Carolin Plate, Nathalie Basselin, Alexander Kröner, Michael Schneider, Stephan Baldes, Vania Dimitrova, and Anthony Jameson. 2006. Recomindation: New functions for augmented memories. In *Proceedings of the 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'06).* 141–150.

[150] Alina Pommeranz, Joost Broekens, Pascal Wiggers, Willem-Paul Brinkman, and Catholijn M. Jonker. 2012. Designing interfaces for explicit preference elicitation: A user-centered investigation of preference representation and elicitation process. *User Modeling and User-Adapted Interaction* 22, 4 (2012), 357–397.

[151] Verus Pronk, Wim F. J. Verhaegh, Adolf Proidl, and Marco Tiemann. 2007. Incorporating user control into recommender systems based on naive Bayesian classification. In *Proceedings of the 1st Conference on Recommender Systems (RecSys'07).* 73–80.

[152] Pearl Pu and Li Chen. 2007. Trust-inspiring explanation interfaces for recommender systems. *Knowledge-Based Systems* 20, 6 (2007), 542–556.

[153] Pearl Pu, Li Chen, and Rong Hu. 2011. A user-centric evaluation framework for recommender systems. In *Proceedings of the 5th Conference on Recommender Systems (RecSys'11).* 157–164.

[154] Pearl Pu, Maoan Zhou, and Sylvain Castagnos. 2009. Critiquing recommenders for public taste products. In *Proceedings of the 3rd Conference on Recommender Systems (RecSys'09).* 249–252.

[155] Al Mamunur Rashid, Istvan Albert, Dan Cosley, Shyong K. Lam, Sean M. McNee, Joseph A. Konstan, and John Riedl. 2002. Getting to know you: Learning new user preferences in recommender systems. In *Proceedings of the 7th International Conference on Intelligent User Interfaces (IUI'02).* 127–134.

[156] Al Mamunur Rashid, George Karypis, and John Riedl. 2008. Learning preferences of new users in recommender systems: An information theoretic approach. *SIGKDD Explorations Newsletter* 10, 2 (2008), 90–100.

[157] James Reilly, Kevin McCarthy, Lorraine McGinty, and Barry Smyth. 2004. Dynamic critiquing. In *Advances in Case-Based Reasoning*, Peter Funk et al. (Ed.). Springer, 763–777.

[158] James Reilly, Kevin McCarthy, Lorraine McGinty, and Barry Smyth. 2005. Incremental critiquing. *Knowledge-Based Systems* 18, 4–5 (2005), 143–151.

[159] Elena Reutskaja and Robin M. Hogarth. 2009. Satisfaction in choice as a function of the number of alternatives: When "goods satiate". *Psychology & Marketing* 26, 3 (2009), 197–203.

[160] Francesco Ricci. 2011. Mobile recommender systems. *Information Technology and Tourism* 12, 3 (2011), 205–231.

[161] Francesco Ricci and Fabio Del Missier. 2004. Supporting travel decision making through personalized recommendation. In *Designing Personalized User Experiences in eCommerce*, Clare-Marie Karat et al. (Ed.). Human-Computer Interaction Series, Vol. 5. Springer, 231–251. Springer.

[162] Francesco Ricci and Quang Nhat Nguyen. 2007. Acquiring and revising preferences in a critique-based mobile recommender system. *Intelligent Systems* 22, 3 (2007), 22–29.

[163] Neil Rubens, Dain Kaplan, and Masashi Sugiyama. 2011. Active learning in recommender systems. In *Recommender Systems Handbook* (1st ed.), Francesco Ricci et al. (Ed.). Springer, 735–767.

[164] Lloyd Rutledge, Natalia Stash, Yiwen Wang, and Lora Aroyo. 2008. Accuracy in rating and recommending item features. In *Proceedings of the 5th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'08).* 163–172.

[165] Thomas L. Saaty. 1977. A scaling method for priorities in hierarchical structures. *Journal of Mathematical Psychology* 15, 3 (1977), 234–281.

[166] Thomas L. Saaty. 1988. What is the analytic hierarchy process? In *Mathematical Models for Decision Support*, Gautam Mitra et al. (Ed.). NATO ASI Series, Vol. 48. Springer, 109–121.

[167] Adem Sabic and Markus Zanker. 2015. Investigating user's information needs and attitudes towards proactivity in mobile tourist guides. In *Information and Communication Technologies in Tourism 2015*, Iis Tussyadiah et al. (Ed.). Springer, 493–505.

[168] Lara Quijano Sánchez, Juan A. Recio-García, and Belén Díaz-Agudo. 2011. Using personality to create alliances in group recommender systems. In *Proceedings of the 19th International Conference on Case-Based Reasoning (ICCBR'11).* 226–240.

[169] J. Ben Schafer, Joseph A. Konstan, and John Riedl. 2001. E-commerce recommendation applications. In *Applications of Data Mining to Electronic Commerce*, Ron Kohavi et al. (Ed.). Springer, 115–153.

[170] J. Ben Schafer, Joseph A. Konstan, and John Riedl. 2002. Meta-recommendation systems: User-controlled integration of diverse recommendations. In *Proceedings of the 11th International Conference on Information and Knowledge Management (CIKM'02).* 43–51.

[171] James Schaffer, Tobias Höllerer, and John O'Donovan. 2015. Hypothetical recommendation: A study of interactive profile manipulation behavior for recommender systems. In *Proceedings of the 28th International Florida Artificial Intelligence Research Society Conference (FLAIRS'15)*. 507–512.

[172] Christian Schmitt, Dietmar Dengler, and Mathias Bauer. 2002. The MAUT-machine: An adaptive recommender system. In *Proceedings of the 10th Workshop on Adaptivität und Benutzermodellierung in interaktiven Softwaresystemen (ABIS'02)*. 83–90.

[173] Michael Scholz and Verena Dorner. 2012. Estimating optimal recommendation set sizes for individual consumers. In *Proceedings of the International Conference on Information Systems (ICIS'12)*. 2440–2459.

[174] Barry Schwartz. 2004. *The Paradox of Choice: Why More is Less*. Harper Perennial.

[175] Barry Schwartz, Andrew Ward, John Monterosso, Sonja Lyubomirsky, Katherine White, and Darrin R. Lehman. 2002. Maximizing versus satisficing: Happiness is a matter of choice.*Journal of Personality and Social Psychology* 83, 5 (2002), 1178.

[176] Aner Sela, Jonah Berger, and Wendy Liu. 2009. Variety, vice, and virtue: How assortment size influences option choice. *Journal of Consumer Research* 35, 6 (2009), 941–951.

[177] Shilad Sen, Jesse Vig, and John Riedl. 2009. Tagommenders: Connecting users to items through tags. In *Proceedings of the 18th International Conference on World Wide Web (WWW'09)*. 671–680.

[178] Avni M. Shah and George Wolford. 2007. Buying behavior as a function of parametric variation of number of choices. *Psychological Science* 18, 5 (2007), 369–370.

[179] Amit Sharma. 2013. PopCore: A system for network-centric recommendation. In *Proceedings of the 2013 Conference on Computer Supported Cooperative Work (CSCW'13)*. 31–34.

[180] Barry Smyth and Paul McClave. 2001. Similarity vs. diversity. In *Case-Based Reasoning Research and Development*. Lecture Notes in Computer Science, Vol. 2080. Springer, 347–361.

[181] E. Isaac Sparling and Shilad Sen. 2011. Rating: How difficult is it?. In *Proceedings of the 5th Conference on Recommender Systems (RecSys'11)*. 149–156.

[182] Panagiotis Symeonidis, Alexandros Nanopoulos, and Yannis Manolopoulos. 2009. MoviExplain: A recommender system with explanations. In *Proceedings of the 5th Conference on Recommender Systems (RecSys'11)*. 317–320.

[183] Erich Teppan, Alexander Felfernig, and Klaus Isak. 2011. Decoy effects in financial service e-sales systems. In *Joint Proceedings of the RecSys 2011 Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'11) and User-Centric Evaluation of Recommender Systems and Their Interfaces-2 (UCERSTI 2) affiliated with the 5th Conference on Recommender Systems (RecSys'11)*. 1–8.

[184] Nava Tintarev, Byungkyu Kang, Tobias Höllerer, and John O'Donovan. 2015. Inspection mechanisms for community-based content discovery in microblogs. In *Proceedings of the Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS'15) co-located with the 9th Conference on Recommender Systems (RecSys'15)*. 21–28.

[185] Nava Tintarev and Judith Masthoff. 2007. A survey of explanations in recommender systems. In *Proceedings of the 23rd International Conference on Data Engineering Workshops (ICDE'07)*. 801–810.

[186] Nava Tintarev and Judith Masthoff. 2011. Designing and evaluating explanations for recommender systems. In *Recommender Systems Handbook* (1st ed.), Francesco Ricci et al. (Ed.). Springer, 479–510. Springer.

[187] Amos Tversky and Daniel Kahneman. 1981. The framing of decisions and the psychology of choice. *Science* 211, 4481 (1981), 453–458. arXiv:http://science.sciencemag.org/content/211/4481/453.full.pdf.

[188] Katrien Verbert, Denis Parra, and Peter Brusilovsky. 2014. The effect of different set-based visualizations on user exploration of recommendations. In *Proceedings of the Joint Workshop on Interfaces and Human Decision Making for Recommender Systems (IntRS'14) co-located with the 8th Conference on Recommender Systems (RecSys'14)*. 37–44.

[189] Jesse Vig, Shilad Sen, and John Riedl. 2009. Tagsplanations: Explaining recommendations using tags. In *Proceedings of the 14th International Conference on Intelligent User Interfaces (IUI'09)*. 47–56.

[190] Jesse Vig, Shilad Sen, and John Riedl. 2011. Navigating the tag genome. In *Proceedings of the 16th International Conference on Intelligent User Interfaces (IUI'11)*. 93–102.

[191] Michail Vlachos and Daniel Svonava. 2012. Graph embeddings for movie visualization and recommendation. In *Joint Proceedings of the 1st International Workshop on Recommendation Technologies for Lifesytle Change (LIFESTYLE'12) and the 1st International Workshop on Interfaces for Recommender Systems (InterfaceRS'12)*. 56–59.

[192] Wesley Waldner and Julita Vassileva. 2014. Emphasize, don't filter!: Displaying recommendations in Twitter timelines. In *Proceedings of the 8th Conference on Recommender Systems (RecSys'14)*. 313–316.

[193] Martijn C. Willemsen, Bart P. Knijnenburg, Mark P. Graus, Linda C. M. Velter-Bremmers, and Kai Fu. 2011. Using latent features diversification to reduce choice difficulty in recommendation list. In *Joint Proceedings of the RecSys 2011 Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'11) and User-Centric Evaluation of Recommender Systems and Their Interfaces-2 (UCERSTI 2) affiliated with the 5th Conference on Recommender Systems (RecSys'11)*. 14–20.

[194] Wolfgang Wörndl, Johannes Huebner, Roland Bader, and Daniel Gallego-Vico. 2011. A model for proactivity in mobile, context-aware recommender systems. In *Proceedings of the 5th Conference on Recommender Systems (RecSys'11)*. 273–276.

[195] Wolfgang Wörndl and Béatrice Lamche. 2015. User interaction with context-aware recommender systems on smartphones. *i-com Journal for Interactive Media* 14, 1 (2015), 19–28.

[196] Wolfgang Wörndl and Part Saelim. 2014. Voting operations for a group recommender system in a distributed user interface environment. In *Poster Proceedings of the 8th Conference on Recommender Systems (RecSys'14)*.

[197] Wolfgang Wörndl, Jan Weicker, and Béatrice Lamche. 2013. Selecting gestural user interaction patterns for recommender applications on smartphones. In *Proceedings of the 3rd Workshop on Human Decision Making in Recommender Systems (Decisions@Recsys'13)*. 17–20.

[198] Julie Wulff and Daniel Hardt. 2014. Can you trust online ratings? Evidence of systematic differences in user populations. In *Proceedings of the 22nd European Conference on Information Systems (ECIS'14)*.

[199] Bo Xiao and Izak Benbasat. 2007. E-commerce product recommendation agents: Use, characteristics, and impact. *MIS Quarterly* 31, 1 (2007), 137–209.

[200] Alexandre Yahi, Antoine Chassang, Louis Raynaud, Hugo Duthil, and Duen Horng (Polo) Chau. 2015. Aurigo: An interactive tour planner for personalized itineraries. In *Proceedings of the 20th International Conference on Intelligent User Interfaces (IUI'15)*. 275–285.

[201] Chun-Ya Yang and Soe-Tsyr Yuan. 2010. Color imagery for destination recommendation in regional tourism. In *Proceedings of the 14th Pacific Asia Conference on Information Systems (PACIS'10)*.

[202] Kyung-Hyan Yoo, Ulrike Gretzel, and Markus Zanker. 2013. *Persuasive Recommender Systems: Conceptual Background and Implications*. Springer.

[203] Markus Zanker and Daniel Ninaus. 2010. Knowledgeable explanations for recommender systems. In *Proceedings of the International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT'10)*. 657–660.

[204] Jiyong Zhang, Nicolas Jones, and Pearl Pu. 2008. A visual interface for critiquing-based recommender systems. In *Proceedings of the 9th Conference on Electronic Commerce (EC'08)*. ACM, 230–239.

[205] Jiyong Zhang and Pearl Pu. 2006. A comparative study of compound critique generation in conversational recommender systems. In *Proceedings of the 4th International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'06)*. 234–243.

[206] Yongfeng Zhang, Guokun Lai, Min Zhang, Yi Zhang, Yiqun Liu, and Shaoping Ma. 2014. Explicit factor models for explainable recommendation based on phrase-level sentiment analysis. In *Proceedings of the 37th International Conference on Research and Development in Information Retrieval (SIGIR'14)*. 83–92.

[207] Shiwan Zhao, Michelle X. Zhou, Quan Yuan, Xiatian Zhang, Wentao Zheng, and Rongyao Fu. 2010. Who is talking about what: Social map-based recommendation for content-centric social websites. In *Proceedings of the 4th Conference on Recommender Systems (RecSys'10)*. 143–150.

[208] Hua Zhou and Kenneth Lange. 2009. Rating movies and rating the raters who rate them. *The American Statistician* 63, 4 (2009), 297–307.

WEB REFERENCES

[209] Chris Alvino and Justin Basilico. 2015. Learning a Personalized Homepage. Retrieved February, 2017 from http://techblog.netflix.com/2015/04/learning-personalized-homepage.html.

[210] Xavier Amatriain and Justin Basilico. 2012. Netflix Recommendations: Beyond the 5 stars (Part 1). Retrieved February, 2017 from http://techblog.netflix.com/2012/04/netflix-recommendations-beyond-5-stars.html.

[211] Nathan McAlone. 2016. Netflix wants to ditch its 5-star rating system. Retrieved February, 2017 from http://businessinsider.de/netflix-wants-to-ditch-5-star-ratings-2016-1.

[212] Merriam-Webster Dictionary. 2016. Persuasion. Retrieved February, 2017 from http://www.merriam-webster.com/dictionary/persuasion.

[213] Shiva Rajaraman. 2009. Five Stars Dominate Ratings. Retrieved February, 2017 from http://youtube-global.blogspot.de/2009/09/five-stars-dominate-ratings.html.

[214] Judy Robertson. 2011. Stats: We're Doing It Wrong. Retrieved February, 2017 from http://cacm.acm.org/blogs/blog-cacm/107125-stats-were-doing-it-wrong.

[215] Harald Steck, Roelof van Zwol, and Chris Johnson. 2015. Interactive Recommender Systems with Netflix and Spotify. Retrieved February, 2017 from http://de.slideshare.net/MrChrisJohnson/interactive-recommender-systems-with-netflix-and-spotify.

[216] Kirsten Swearingen and Rashmi Sinha. 2002. Interaction design for recommender systems. Retrieved February, 2017 from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.15.7347&rep=rep1&type=pdf.

Received March 2016; revised October 2016; accepted January 2017