Evaluating Recommender Systems: Survey and Framework

EVA ZANGERLE, Universität Innsbruck, Austria
CHRISTINE BAUER, Utrecht University, The Netherlands

The comprehensive evaluation of the performance of a recommender system is a complex endeavor: many facets need to be considered in configuring an adequate and effective evaluation setting. Such facets include, for instance, defining the specific goals of the evaluation, choosing an evaluation method, underlying data, and suitable evaluation metrics. In this article, we consolidate and systematically organize this dispersed knowledge on recommender systems evaluation. We introduce the Framework for Evaluating Recommender systems (FEVR), which we derive from the discourse on recommender systems evaluation. In FEVR, we categorize the evaluation space of recommender systems evaluation. We postulate that the comprehensive evaluation of a recommender system frequently requires considering multiple facets and perspectives in the evaluation. The FEVR framework provides a structured foundation to adopt adequate evaluation configurations that encompass this required multi-facetedness and provides the basis to advance in the field. We outline and discuss the challenges of a comprehensive evaluation of recommender systems and provide an outlook on what we need to embrace and do to move forward as a research community.

CCS Concepts: • **Information systems → Recommender systems**; **Evaluation of retrieval results**; • **Human-centered computing → HCI design and evaluation methods;**

Additional Key Words and Phrases: Survey, Framework for EValuating Recommender systems, FEVR

ACM Reference format:
Eva Zangerle and Christine Bauer. 2022. Evaluating Recommender Systems: Survey and Framework. *ACM Comput. Surv.* 55, 8, Article 170 (December 2022), 38 pages.
https://doi.org/10.1145/3556536

1 INTRODUCTION

Recommender systems (RS) have become important tools in people's everyday life, as they are efficient means to find and discover relevant, useful, and interesting items such as music tracks [41], movies [29, 50], or persons for social matching [44]. A RS elicits the interests and preferences of individual users (e.g., by explicit user input or via implicit information inferred from the user's interactions with the system) and tailors content and recommendations to these interests and needs [219]. As for most systems, the evaluation of RS demands attention in each and every phase throughout the system life cycle—in design and development as well as for continuous improvement while in operation. Delivering quality is a necessary factor for a system to be successful in

Eva Zangerle and Christine Bauer contributed equally to this work.
This research was funded in whole, or in part, by the Austrian Science Fund (FWF): P33526.
Authors' addresses: E. Zangerle, Universität Innsbruck, Technikerstr. 21A, Innsbruck, 6020, Austria; email: eva.zangerle@uibk.ac.at; C. Bauer, Utrecht University, Utrecht, 3584 CC, Princetonplein 5, The Netherlands; email: c.bauer@uu.nl.

0360-0300/2022/12-ART170
https://doi.org/10.1145/3556536

practice [8]. The evaluation may assess the core performance of a system in its very sense or may embrace the entire context in which the system is used [23, 101, 115, 184, 189]. Research on RS typically differentiates system-centric and user-centric evaluation, where the former refers to the evaluation of algorithmic aspects (e.g., the predictive accuracy of recommendation algorithms). The latter targets the users' perspective and evaluates how users perceive its quality or the user experience when interacting with the RS. In other words, the evaluation of a RS may cover system- or user-centric aspects concerning the system's context of use; a comprehensive evaluation essentially needs to address both as, for instance, provided recommendations that are adequate in terms of system-centric measures—for instance, the predictive accuracy of recommendation algorithms—do not necessarily meet a user's expectations [138, 157].

As we will demonstrate in this paper, there is an extensive number of dimensions that need to be considered when assessing the performance of a RS [93]. Besides the various facets of system configurations and the multitude of tasks that users aim to address with a RS (for instance, finding good items to getting a recommendation for a sequence of items) [101], there are multiple stakeholders involved who may have varying perspectives on a RS' goals [18]. There is a rich evaluation design space (e.g., evaluation setup, data collection, employed metrics) to draw from, and we have to specify evaluation configurations that meet the respective evaluation objectives. Such objectives may relate to, for instance, improving rating prediction accuracy, increasing user satisfaction and experience, or increasing click-through rates and revenue. As a consequence, the comprehensive evaluation of a RS is a very complex task. As the ultimate goal is that a RS functions well as a whole in various contexts (e.g., for different user groups, for different kinds of tasks and purposes), the evaluation needs to assess the various dimensions that make up a RS' performance. What is more, frequently, we might need to shed light on a single dimension from various angles. For instance, Kamehkhosh and Jannach [124] could reproduce—and, thus, confirm—the results of their offline evaluation in an online evaluation on users' perceived quality of recommendations. Matt et al. [155] evaluated several recommender algorithms for diversity effects from various angles; in taking these different perspectives, they found that the level of recommendation diversity perceived by users does not always reflect the factual diversity.

While the knowledge about system evaluation—and RS evaluation in particular—is continuously growing, empirical evidence, insights, and lessons learned are scattered across papers and research communities. To fill this research gap, this paper's main objective and major contribution is to consolidate and systematically organize this dispersed knowledge on RS evaluation. Therefore, we introduce the **Framework for Evaluating Recommender systems (FEVR)**, which we derive from the discourse on RS evaluation. We categorize the evaluation design space—i.e., the space that spans all required design decisions when conducting comprehensive RS evaluations. With FEVR, we provide a systematic overview of the essential facets of RS evaluation and their application. As FEVR encompasses a wide variety of facets to be considered in an evaluation configuration, it can accommodate comprehensive evaluations that address the various multi-faceted dimensions that make up a RS' performance. Besides guiding novices to RS research and evaluation, FEVR is a profound source for orientation for scientists and practitioners concerned with designing, implementing, or assessing RS. In addition, FEVR provides a structured basis for systematic RS evaluation that the RS research community can build on. We expect FEVR to serve as a guide to facilitate and foster the repeatability and reproducibility of RS research for researchers and practitioners, from novices to experts. Yet, comprehensive evaluation comes with challenges. Thus, to date, RS literature seems to concentrate on accuracy-driven offline evaluations and does not reflect the existing knowledge about what a comprehensive evaluation requires [39, 109, 111]. We outline and discuss the challenges of comprehensive RS evaluation, and provide an outlook on what we need to embrace and do to move forward as a research community.

		Items				
		i_1	i_2	i_3	i_4	i_5
Users	u_1	0	3			3
	u_2	2				
	u_3			2	4	
	u_4		3		1	
	u_5	5		1		1

Fig. 1. Exemplary user-item matrix M with 0–5 star ratings for items i_1–i_5 by users u_1–u_5.

2 CONCEPTUAL BASIS

In the following, we briefly describe the foundations of recommender systems (Section 2.1) and their evaluation (Section 2.2).

2.1 Recommender Systems

Recommender systems aim to help users to deal with information and choice overload [9] by providing them with recommendations for items that might be interesting to the user [178, 181]. In the following, we give a brief overview of the foundational recommendation approaches: collaborative filtering, content-based RS, and more recent advances.

The most dominant approach for computing recommendations is collaborative filtering [192, 193], which is based on the collective behavior of a system's users. The underlying assumption is that users who had similar preferences in the past will also have similar preferences in the future. Hence, recommendations are typically computed based on the users' past interactions with the items in the system [32, 67, 101, 192, 193]. These interactions are recorded in a user-item rating matrix, where the users' ratings for items are stored. Such ratings may either refer to explicit ratings where users assign scores on a scale of, e.g., 0-5, to items, or implicit ratings. Figure 1 shows an example of such a user-item matrix. Note that user-item matrices are highly sparse, as users only rate a small fraction of items available in the system. The algorithmic task of a RS is that of matrix completion—i.e., predicting the missing ratings in the matrix. This prediction of ratings can be performed using various methods: from traditional matrix completion methods, over neighborhood-based methods to matrix factorization, machine or deep learning-based approaches. For further information on these approaches, we refer the interested reader to the existing literature on these topics (e.g., References [67, 140, 161, 192, 193, 206, 226]).

For user-based collaborative filtering RS that leverage the neighborhood of users in the two-dimensional space of the matrix, the most similar users to the current user are detected (the so-called neighborhood) by comparing their interactions with the system. Analogously, in item-based (item-item) collaborative filtering [192], the most similar items to the ones the user has previously rated highly are recommended, where the similarities are again computed based on the user-item matrix. Subsequently, items the user has not interacted with are sorted by their predicted ratings and the top-n items are then recommended to the user.

For collaborative filtering tasks, **Matrix Factorization (MF)** [139] aims to find latent factors in a joint, lower-dimensional space that explain user ratings for a given item. Specifically, latent representations for users and items are computed such that user-item interactions can be modeled as the inner product of user and item representations. This is often performed by applying optimization approaches to decompose the user-item matrix into two lower-dimensional matrices (e.g., stochastic gradient descent or alternating least squares), mostly relying on a regularized model to avoid overfitting (e.g., References [83, 105]). Furthermore, learning-to-rank approaches model the computation of recommendations as a ranking task and apply machine learning to

model the ranking of recommendations. In principle, we differentiate three types of learning-to-rank approaches: (i) point-wise (compute a score for each item for ranking; used in traditional CF approaches), (ii) list-wise (compute an optimal order of a given list), and (iii) pair-wise (consider pairs of items to approximate the optimal ordering of the list). Bayesian Personalized Ranking is a popular example for learning-to-rank models; it is a generic, model-agnostic learning algorithm for predicting a personalized ranking [176] based on training pairs that incorporate positive and negative feedback.

In contrast, the central idea of content-based approaches is to recommend items that share characteristics with items that the user has previously liked (for instance, items that have a similar description or genre) [4, 160, 168]. Based on these characteristics of the user's previously liked items, a user model (often referred to as user profile) is built that represents the user's preferences. For the computation of recommendations, the user model is matched against item characteristics, and the most similar and, hence, relevant items are subsequently recommended to the user. Hybrid RS aim to combine collaborative and content-based filtering to leverage the advantages of both [33].

Similar to many other fields, a multitude of machine and deep learning models have been adapted for use in recommender systems. These include, for instance, deep neural networks for collaborative filtering, where the user-item interactions are modeled by a neural network [99], deep factorization machines [95], or (variational) autoencoders [148]. Convolutional Neural Networks (CNN) are mostly used for learning features from (multimedia) sources. For instance, learning representations from audio signals and incorporating them in a CF approach [212], or extracting and modeling latent features from user reviews and items [229]. Recurrent Neural Networks (RNN) allow modeling sequences and, hence, are applied for sequential recommendation tasks such as playlist generation or next-item recommendation [102, 175]. The use of reinforcement learning models for recommendation tasks is often performed by formulating the task as a multi-armed bandit problem (contextual bandits) [146, 156, 228]. Here, the bandit sequentially provides recommendations to users by also incorporating their contexts while continuously updating and optimizing the recommendation model based on user feedback. Furthermore, Graph Convolutional Networks (GCN) model users, items, and potential side information in a graph. Based on this information, latent representations for nodes are learned by aggregating feature information from local neighbors (e.g., References [98, 167]). This allows using these representations for candidate generation by nearest-neighbor lookups [222] or performing link-prediction tasks [25]. For a survey on deep learning for recommender systems, please refer to Zhang et al. [226]. In the context of evaluating deep learning recommender systems, it is noteworthy that evaluation metrics (cf. Section 3.4.4) are frequently used as loss functions (i.e., during the training phase).

Besides traditional recommendation approaches, there are several important extensions and specialized recommender systems that allow to deal with further input data or adapt to more specific use cases. These include, amongst others, context-aware recommender systems [3], where further contextual factors that describe, e.g., the user's situation (for instance, time, location, weather) are leveraged to compute recommendations that are suitable for a given user in a given context. Sequential (or sequence-aware) recommender systems [174] analyze the sequence of user interactions to compute sequences of recommendations (e.g., recommending the next song to listen to, given a sequence of songs the user has just listened to). Conversational recommender systems provide more sophisticated interaction paradigms for preference elicitation, item presentation, or user feedback [113]. All of these approaches go beyond traditional recommender systems and user interactions and, hence, also require more complex evaluation methods and setups. We refer the interested reader to the respective survey articles [3, 113, 174] for details on such evaluations.

2.2 Evaluation of Recommender Systems

An evaluation is a set of research methods and associated methodologies with the distinctive purpose of assessing quality [205]. In their book, Jannach et al. [117] state that evaluations are "methods for choosing the best technique based on the specifics of the application domain, identifying influential success factors behind different techniques, or comparing several techniques based on an optimality criterion" and that considering all these aspects is all required for effective evaluation research.

One of the early works on evaluating RS by Herlocker et al. [101] focuses on the evaluation of collaborative filtering RS. The authors stress that evaluating RS is inherently difficult as (i) algorithms may perform differently on different datasets, (ii) evaluation goals may differ, and (iii) choosing the right metrics to compare individual approaches is a complex task. Gunawardana et al. [93] provide a general overview of evaluation methods for RS.

Beel et al. [21, 23] investigated evaluation approaches in the field of research paper recommender systems. They find that 21% of all approaches do not include an evaluation and that 69% are evaluated using an offline evaluation. Furthermore, they also looked into baseline usage and the datasets utilized for the evaluation. The authors note that the wide usage of no or weak baselines, as well as the usage of very different datasets, makes it difficult to compare the performance of the individual approaches, which in turn severely hinders advancing research in the field. Dehghani Champiri et al. [57] performed a systematic literature review on evaluation methods and metrics for context-aware scholarly recommender systems. In a meta-analysis, they reviewed 67 studies and find that offline evaluations are the most popular experiment type.

Comparing a RS' performance results to existing approaches and to competitive, strong baselines is also an important aspect for assessing and contextualizing the performance of the system. In this regard, Rendle et al. [177] show that several widely-used baseline approaches, when carefully set up and tuned, outperform many recently published algorithms on the MovieLens 10M benchmark [97]. Along the same lines, Ferrari Dacrema et al. [77, 78] investigated the performance of deep learning recommendation approaches published at major venues between 2015 and 2018, particularly, when compared to well-tuned, established, non-neural baseline methods. They found that the majority of approaches were compared to poorly-tuned, weak baselines and that only one of twelve neural methods was consistently outperforming well-tuned learning-based techniques.

Complimentary to existing works on RS evaluation, we consolidate and systematically organize this knowledge in the proposed FEVR systems.

3 RECOMMENDER SYSTEMS EVALUATION: A REVIEW

Figure 2 presents an overview of the components and general factors to be considered for recommender systems evaluation. Along this framework, we present the conceptual basis and paradigms used in recommender system evaluations. We term the framework *FEVR: Framework for Evaluating Recommender systems* and emphasize that not necessarily all of these components and factors might be required to conduct a comprehensive evaluation of RS (this particularly holds for the proposed evaluation aspects). We consider this framework a collection and overview of potentially relevant components; it is meant to provide researchers and practitioners with an overview of the choices to be made when setting up the evaluation design and procedure.

The framework contains two main components: the *evaluation objectives* and the *evaluation design space*. When designing RS evaluations, deciding upon the objectives of the evaluation (*What should be evaluated? How can we measure this?*) has to be the first step, because this directly influences the design decisions for the evaluation setup. The second component, the evaluation design space contains basic building blocks for the actual setup of the evaluation, which are assembled and configured based on the overall goal, the stakeholders involved, and the properties of the RS that

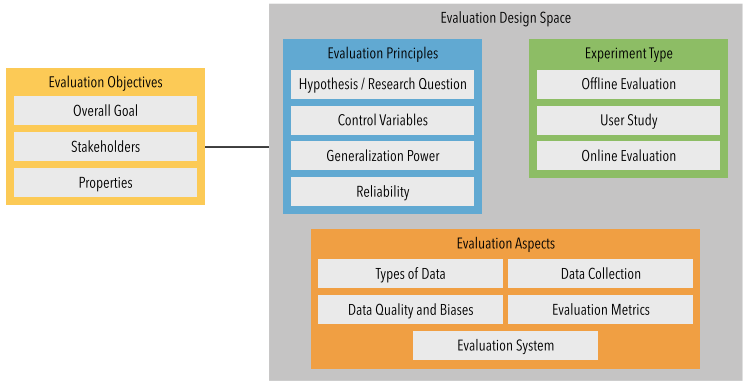

Fig. 2. Framework for Evaluating Recommender systems (FEVR): evaluation objectives and the design space (along the orthogonal dimensions of evaluation principles, experiment type, and evaluation aspects).

need to be evaluated. In the evaluation design space, we distinguish three design blocks. The so-called *evaluation principles* describe the guiding principles of the evaluation—from the definition of the hypothesis underlying the evaluation to the generalizability of the conducted evaluation. These principles are tightly connected and influenced by the defined objectives, because, for instance, the hypothesis to be evaluated needs to reflect the main objective of the evaluation (e.g., investigating whether algorithm A performs better than algorithm B). Given the objectives and principles, the *experiment type* can be considered a broad categorization of the type of experiment conducted to satisfy the objectives and principles (offline evaluation, user study, or online evaluation). *Evaluation aspects* can be considered a more fine-grained specification of the evaluation setup and, based on the defined requirements, control the detailed evaluation setup. They can be considered a set of configurations and decisions that do not necessarily all have to be considered for a RS evaluation; they should provide guidance for setting up and conducting comprehensive evaluations. We consider the choices of evaluation principles and experiment type rather high-level, whereas evaluation aspects cover more detailed and specific decisions regarding the evaluation setup.

In the following, we detail each of the framework's components and discuss their role in the activities of evaluating recommender systems.

3.1 Evaluation Objectives

At the heart of any evaluation activity is the comparison of the objectives (target performance) to the observed results (actual performance) [170]. Thus—whether explicitly or implicitly stated—evaluation is always based on one or more *evaluation objectives*. Evaluation objectives for evaluating a RS may take many forms. Essentially, objectives are shaped by the *overall goal* of academic and/or industry partners and the purpose of the system [109]. In this context, Herlocker et al. [101] underline that any RS evaluation has to be goal-driven. Schröder et al. [195] emphasize that setting the goal of an evaluation has to be executed with sufficient care and it should be the first step of any evaluation to "define its goal as precisely as possible."

The underlying premise of any RS evaluation—in academia and industry—is that a RS is supposed to create value in practice [115] and have an impact in the real world [111]—in the long run, or even in the short run. Thus, overall goals that are typically investigated by academia, as well as industry, include, among others, a RS' contribution to increasing the user satisfaction [181], increase an e-commerce website's revenue [93], increase the number of items sold [181], sell more diverse items [181], help users understand the item space [109, 116], and engage users to increase

Table 1. Overview of Papers on the Evaluation of RS Considering Different Stakeholders' Perspectives

Stakeholder	Examples
Consumer	[82, 129, 132, 134, 172, 173]
Consumer Groups	[76]
Platform Provider	[21, 109, 112]
Item Provider	[179]
Multiple Stakeholders	[1, 18, 35]

their visit duration on a website, or return to the website [218]. Although several goals and purposes of RS are addressed in RS research and evaluation, it is remarkable that this variety of user tasks and RS purposes is not widely reflected in literature; instead, the main interpretation of the purpose of a RS seems to be "help users find relevant items," while other recommendation purposes are largely underexplored in the literature [109].

Concerning setting an evaluation goal, Schröder et al. [195] provide a vivid example of a precise evaluation goal: "Find the recommendation algorithm and parameterization that leads to the highest overall turnover on a specific e-commerce website, if four product recommendations are displayed as a vertical list below the currently displayed product." Crook et al. [54] consider *prediction*, *ranking*, and *classification* as the most common tasks when viewed from the system's perspective. Considering the end consumers' perspective, Herlocker et al. [101] discuss various end consumer tasks that a RS might be able to support (e.g., finding good items, finding all good items, recommending a sequence, discovering new items). Such tasks essentially describe the end consumers' overall goals that a RS might be evaluated for. A RS may, thus, be evaluated for their ability to find good items, find all good items, recommend a sequence, or discover new items. When describing pitfalls and lessons learned from their evaluation activities, Crook et al. [54] emphasize the importance of choosing an overall evaluation goal that truly reflects business goals.

In general, evaluation objectives are shaped by the perspective that is taken in terms of the recommender's *stakeholders*. Beyond the end consumers, there are typically multiple stakeholders involved in and affected by recommender systems [1] with varying goals and potentially conflicting interests [18], which may manifest in different evaluation objectives. Currently, academic RS research tends to take the perspective of the end consumer [110], whereas research in industry is naturally built around the platform or system provider's perspective [225]. The item providers are a relatively new concern in RS research (e.g., References [71, 79, 96]). To date, RS research that takes multiple stakeholders into account is scarce [1, 18, 62]. Table 1 provides an overview of evaluation papers that take different stakeholders' perspectives.

While evaluating a recommender's overall goals (e.g., for an increase in a website's revenue) can be helpful, Gunawardana et al. [93] point out that it can be most useful to evaluate how recommenders perform in terms of specific *properties*. This allows focusing on improving specific properties where they fall short (e.g., usage prediction accuracy, sales diversity, confidence in the recommendation, privacy level). The challenge is to identify the properties that are indeed relevant for a recommender's performance and show that it affects the users' experience [93], or the interests of other stakeholders. As different domains, applications, and consumer tasks have different needs, it is essential to decide on the most important properties to evaluate for the concrete RS at hand [93]. As already pointed out, Schröder et al. [195] emphasize the importance to define the evaluation goal as precisely as possible. Accordingly, specifying the relevant properties will provide the necessary fine granularity in defining the evaluation objective. As there might be

trade-offs between sets of properties, it is often difficult to anticipate how these trade-offs affect the overall performance of the system [93]; this has to be considered in finding an appropriate evaluation design.

The evaluation objectives—including the overall goal, the stakeholder(s) being addressed, and the properties in the loop—are central to any evaluation effort and are, thus, the main drivers for configuring the evaluation design. We emphasize that poorly defined objectives will inevitably result in a poor evaluation.

3.2 Evaluation Design Space: Evaluation Principles

Closely related to the previously described evaluation objectives is a set of guiding principles for conducting evaluations [93]. These principles are pivotal in the process of designing and conducting RS evaluations, because they lay the foundation of the evaluation procedure and provide the foundation of the setup. Hence, they should be considered and fixed early on in the process of evaluating a RS to shape the method and setup of the evaluation.

The first evaluation principle concerns hypotheses (or research questions) that capture the evaluation objectives. Depending on the overall goal and whether a problem can be clearly defined, the evaluation's overall goal may be translated to one or more *a priori* formulated *hypotheses* that are grounded on prior knowledge (e.g., observations or theory) [217], or to one ore more exploratory-driven (broader) *research questions*.

Confirmatory evaluation involves testing one or more *a priori* formulated *hypotheses*. Hence, a central starting point for confirmatory evaluation is the formulation of one or multiple *hypotheses* regarding the outcome of the evaluation. Defining a concise hypothesis is a highly important step as it allows to precisely define the evaluation's goal—the more precise the hypothesis, the clearer the evaluation setup as the hypothesis (in line with the evaluation objectives) shapes the evaluation design.[1] An example of a hypothesis for RS evaluation in the field of content-based video recommendations is "Our recommendation algorithm based on visual features leads to a higher recommendation accuracy in comparison with conventional genre-based recommender systems" [59]. Another example is Knijnenburg and Willemsen [131]'s hypothesis regarding preference elicitation (PE): "Novices have a higher satisfaction and perceive the system as more useful when they use the case-based PE method (compared to the attribute-based PE method), while experts have a higher satisfaction and perceive the system as more useful when they use the attribute-based PE method (compared to the case-based PE method)." Jannach and Bauer [111] claim that algorithmic RS research frequently comes without (appropriate) hypothesis development; they call for more theory-guided research with clear pointers to underlying theory (e.g., from psychology) that support the hypotheses.[2]

Yet, sometimes the evaluation objectives address a problem where little is known about the phenomenon. In such situations, the problem cannot be clearly defined at this state of research and the evaluation might, thus, be of exploratory nature (e.g., to get a better understanding of a problem or explore patterns). In such cases, it is not possible or suitable to formulate hypotheses. Instead, the evaluation's overall goal can be addressed by formulating research questions. For instance, Liang and Willemsen [149] seek to understand the effects of defaults in music genre exploration

[1]For a discourse on the issues related to hypothesis-testing if a field is prone to produce "pseudo-empirical hypotheses" see Smedslund [199]. Smedslund [199] particularly emphasizes the problem that there is a prevailing belief (i.e., the current paradigm centered on the notion of probability) that "hypotheses that make sense are true, and hypotheses that do not make sense are false." For a discussion on the role of confirmation bias in making progress in research see Greenwald et al. [91] or Wagenmakers et al. [217].

[2]As an example, Jannach and Bauer [111] state that many works build on underlying assumptions such as "higher diversity is better" without providing any pointers to underlying research that would support such an assumption.

for which they formulate three research questions. Concerning author gender distribution in book recommendations, and Ekstrand and Kluver [71] explore how individual users' preference profiles propagate into the recommendations that they receive.

In hypothesis testing, all variables in the RS ecosystem that are not evaluated should be held fixed. Also in exploratory evaluation, the researcher exercises some control over the research conditions to explore the phenomenon of interest. The second evaluation principle, *control variables* (or short: controls) minimize the confounding variables, and we eliminate potential external influences on the evaluation result [93, 216]. This allows a targeted evaluation and comparison of different algorithms and configurations by ensuring that only variables that are evaluated can be changed and that differences in the evaluation results are not due to some further, external factors. Going back to the previous example hypothesis regarding preference elicitation, the authors tested the hypothesis by utilizing the PE method, user expertise, and commitment as independent variables and measured satisfaction with the system, perceived usefulness, understandability, and satisfaction with the chosen measures as dependent variables, while fixing all other variables. Jannach et al. [117] refer to these controlled test conditions as the "internal validity" [37] of experiments.

The third important principle is the *generalization power* of evaluations, the extent to which the conclusions of the evaluation are generalizable beyond the current evaluation setup and experiments. The generalization power is tightly connected to the evaluation setup as, e.g., varying the experimental setup, conducting experiments with different datasets, or extending the experiments to cover further application domains, user groups or stakeholders typically increases the generalization power of the evaluation [190]. Jannach et al. [117] refer to this as "external validity" [37], namely, the "extent to which results are generalizable to other user groups or situations [169]."

Reliability [117] is the fourth cornerstone of research evaluations as it demands evaluations to be consistent and free of errors (in both data and measurements). Particularly the consistency of multiple evaluation runs is crucial as this demonstrates the highly desirable repeatability of experiments, i.e., the ability to observe similar results of experiments conducted successively under the same (documented) settings and configurations, allowing consistent results describing the RS' performance. Tightly connected to repeatability is reproducibility, which refers to the ability "to duplicate the results of a prior study using the same materials as were used by the original investigator. That is, a second researcher might use the same raw data to build the same analysis files and implement the same statistical analysis in an attempt to yield the same results . . . Reproducibility is a minimum necessary condition for a finding to be believable and informative" [90]. Reproducible results require either access to the source code or a detailed description of the algorithm such that it can be re-implemented as well as having access to the dataset that was originally used. In this context, it is important to differentiate between reproducibility and replicability, which can be defined as the ability "to duplicate the results of a prior study if the same procedures are followed but new data are collected" [90]. In a nutshell, the three key concepts here can be defined as follows: reproducibility (different team, different experimental setup), repeatability (same team, same experimental setup), and replicability (different team, same experimental setup) as stated by the Association for Computing Machinery's badging initiative.[3]

The ACM Conference on **Recommender Systems (RecSys)** has introduced a specific reproducibility track in 2020, which calls for "algorithmic papers that repeat and analyze prior work."[4] Notably, this track calls for replicability as well as reproducibility papers. To further stress the importance of reproducibility, the best paper award of RecSys 2019 was awarded to Ferrari Dacrema

[3]https://www.acm.org/publications/policies/artifact-review-badging, also following Hong [103].

[4]Call for Papers (Reproducibility Track) for RecSys 2022: https://recsys.acm.org/recsys22/call/#content-tab-1-1-tab.

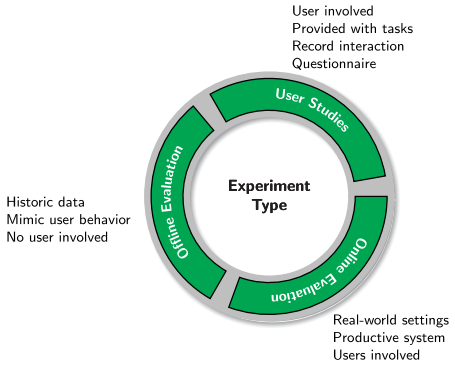

Fig. 3. Spectrum of experiment types.

et al. [78], in which the authors aim to reproduce the results of 18 papers from the field of deep learning recommender algorithms. In an extended version of that study, Ferrari Dacrema et al. [77] find that only twelve out of the 26 evaluations had a reproducible setup, corresponding to a total of 46% of all systems. Here, the authors considered a paper to have a reproducible setup if (i) a working version of the source code is available or the code only has to be modified in minimal ways to work correctly, and (ii) at least one dataset used in the original paper is available (this also includes the train-test splits to be available or at least be reconstructible based on the description in the paper). The importance of documenting train/test splits (among other factors) is also highlighted by Cañamares et al. [36], who show that different splitting methods and factors can lead to diverse evaluation results. On a similar note, Bellogín and Said [24] make the case for accountability and transparency in RS research and argue that only if the conducted research and evaluation is reproducible, it is also accountable. They discuss the requirements for accountable RS research and derive a framework that allows for reproducible and, hence, accountable RS evaluation.

3.3 Evaluation Design Space: Experiment Type

In RS research, we distinguish three experiment types: offline evaluations, user studies, and online evaluations [20, 22, 81, 93, 101]. These different types describe the general experimental setup; Gunawardana and Shani [92] also refer to these types as "evaluation protocols." The characteristics of these types include, among others, aspects of user involvement, utilized and obtainable data, or the type of insight that can be gained when using a specific experiment type. Please note that experiments of more than one type may be necessary to obtain a full picture of the performance of a RS. Offline evaluations are often the first step in conducting evaluations and there is a "logical evolution from offline evaluations, through user studies to online analyses" [81]. Figure 3 shows an overview of the three experiment types, emphasizing that they represent a contrasting spectrum of experiments, covering diverse and different aspects of RS performance, where each type comprises a wide variety of evaluation setups and configurations.

Table 2 features an overview and comparison of the three established experiment types utilized in the RS research community. In the following, we further elaborate on their characteristics, goals, usage scenarios, and differences. Offline evaluations aim to compare different recommendation algorithms and settings; they do not require any user interaction and may be considered system-centric. In contrast, both, user studies and online evaluations, involve users and can be considered user-centric. Still, user involvement in evaluation does not necessarily target or capture the user experience, as discussed in Knijnenburg and Willemsen [132]. Also, for instance, Celma and Herrera [40] refer to leave-*n*-out methods, a typical offline evaluation method, as user-centric;

Table 2. Overview of Experiment Types

Type	Description
Offline	Method: simulation of user behavior based on past interactions Task: defined by the researcher, purely algorithmic Repeatability: evaluation of an arbitrary number of experiments (e.g., algorithmic settings, models) possible at low cost Scale: large dataset, large number of users Insights: quantitative, narrow (focused on the predictive performance of algorithms)
User Study	Method: user observation in live or laboratory setting Task: defined by the researcher, carried out by the user Repeatability: expensive (recruitment of users) Scale: small cohort of users Insights: quantitative and/or qualitative (live user data, logging of user actions, eye tracking, questionnaires before/during/after task)
Online	Method: real-world user observation, online field experiment Task: self-selected by the user, carried out by the user Repeatability: expensive (requires full system and users) Scale: size of the cohort of users depending on evaluation system and user base Insights: quantitative and/or qualitative (live user data, logging of user actions, questionnaires before/during/after exposure to the system)

while at the same time, they state that those evaluations measure accuracy and neglect (user-perceived) efficiency of recommendations.

Orthogonal to the distinction between online and offline experiments and user studies, Said et al. [189] and Knijnenburg and Willemsen [132] distinguish system- and user-centric evaluations and emphasize the different objectives of the adopted evaluation methods: system-centric evaluation methods evaluate the system, while user-centric evaluation methods target the user experience when interacting with the system.

3.3.1 Offline Evaluation. In research literature, the most frequently used experiment type for RS evaluation are so-called "offline evaluations."[5] An offline evaluation uses a pre-collected dataset that contains users' explicit feedback on items (e.g., ratings of items) or implicit feedback on items (e.g., the items purchased, viewed, or consumed) [93]. User behavior is then mimicked and simulated based on this historical data, no real users (and their interactions with the system) are involved in the actual experiments. For the experiments, parts of the rating information are removed (at random) from the given dataset's user-item matrix (so-called leave-*n*-out evaluation [52]) and, subsequently, the recommender algorithms are analyzed regarding their ability to recommend (i.e., predict) the missing information [20, 22]—assessing whether the given recommender is apt to simulate user behavior to predict ratings that are reflected in the previously hidden data. Typically, offline evaluations are used to compare two or more RS algorithms (offline A/B testing [88]). Offline evaluations are meant to identify promising recommendation approaches by using metrics such as algorithmic accuracy and precision [20, 22, 132], and evaluating the predictive power of the approaches in regards to user preferences and opinions [81]—thus, the scope of evaluation objectives that can be evaluated with an offline evaluation is rather narrow [93] and focused on algorithmic tasks. It is, however, easy to repeat offline experiments as each evaluation run can be repeated any number of times using different recommender setups, algorithm parameters,

[5] According to Jannach and Bauer [111], more than 92% of the 117 RS papers published at AAAI and IJCAI in 2018 and 2019 relied exclusively on offline experiments. At ACM RecSys 2018 and 2019, three of four papers only used offline evaluations. For the years 2006–2011, more than two-thirds of papers relied on offline experiments Jannach et al. [118].

datasets, users and also, at an arbitrary scale regarding the input dataset and the number of users evaluated.

Temporal aspects of data can be critical in the design of such an evaluation. Burke [34] suggests a "temporal leave-one-out approach," where the timestamps are considered in selecting which part of the data is used for training the model and which part for testing. Gunawardana et al. [93] emphasize that selecting data based on timestamps allows for simulating what a recommender's predictions would have been if it had been running at the time when the data was available. Starting with no available prior data for computing predictions and stepping through user and interaction data in temporal order may be ideal in terms of simulating the system's behavior along the timeline; however, for large data sets, such an approach is computationally expensive [93].

While offline evaluations are widely used to obtain insights into the predictive performance of different recommendation algorithms, there are also disadvantages to offline evaluations. Given the described setup that relies on historic data, offline evaluation does not involve (current) real users. There is no interaction of users with the given (to be evaluated) RS algorithm in an actual system and the performance of the algorithm in a real-world scenario cannot be assessed. Hence, the generalizability (external validity) of the findings obtained by offline experiments is limited, and frequently questioned [183]. For instance, a recent study [123] showed that offline experiments on historical data for a destination recommender system did show higher predictive accuracy than a subsequent user study. In another study [183], offline experiments underestimated the precision results of online evaluations.

Counterfactual learning methods [6, 207] overcome one of the key problems in offline evaluation; namely, that the dataset was logged from a real-world platform where a particular RS was active (i.e., logged policy) while the offline evaluation has the objective to evaluate another RS algorithm (i.e., target policy). With counterfactual learning methods, one can address the question of how well a new RS algorithm would have performed if it had been used instead of the policy that logged the historical data. This counterfactual approach also reduces the effect of selection biases (i.e., biases introduced into the data through the actions selected by the logging policy) [122].

3.3.2 User Study. A user study is conducted by recruiting a (small) set of human test subjects who perform several pre-defined tasks that require them to interact with the RS [93]. The goal here is to observe user interaction with the system and to distill real-time feedback on the system's performance and the user's perceived value of the system. This observation can either be conducted in a laboratory or live setting. Thereby, the user study may be conducted in a way to compare two or more systems in, for instance, an experimental setup (controlled experiment[6]); a user study may also focus on exploring a particular phenomenon without comparing specific RS approaches (exploratory study) [173]. The subject's interaction behavior with the system is recorded and based on these records, various quantitative measures may be computed (e.g., time to complete a task, click-through rate, recommendation acceptance). In addition, the setting of a user study allows for asking subjects closed or open-ended questions during, before, and after the task potentially also providing qualitative feedback [93]. Further, user studies allow for integrating various forms of measurements such as eye-tracking or think-aloud-testing [163]. Hence, user studies allow for the most comprehensive feedback compared to the other experiment types, enabling answers to the widest set of questions. Notably, user studies measure user experience at the time of recommendation.

It is important to note that user studies may lead to costs [22, 57]—both in user time and financial costs, often limiting the number of users being involved in the study or the number of different

[6]Although an experimental setup may compare two or more variants of a RS, the term A/B testing is typically not used in the context of user studies.

system dimensions and configurations that can be investigated and evaluated [81]. This also involves recruiting a set of participants that are willing to participate in the experiment. These participants should be representative of the actual users of the system and have access to a running recommender system. Furthermore, users who know that they are part of a study often tend to behave differently (called "Hawthorne effect" [144]). Generally, user studies need extensive preparation and planning as repeating is expensive. Besides, a wide range of sensors and detailed observations of user behavior need to be installed to make sure to not miss any vital information during the study as a potential rerun of the experiment may be expensive. These factors can be regarded as causes of the low adaption of user studies in the field of RS research [20, 22].

3.3.3 Online Evaluation. In online evaluations, the RS is deployed in a real-world, live setting [93]. In contrast to user studies, users are not presented with specific tasks, but use the system to perform self-selected real-world tasks (also referred to as "live user studies" [81]). Hence, online evaluations allow for the most realistic evaluation scenario as users are self-motivated and use the system in the most natural and realistic manner [135, 136]. Accordingly, online evaluations provide feedback on the system's performance for users with a real information need [93]. Similar to user studies, user behavior is logged and recorded and subsequently used to distill performance metrics such as recommendation accuracy. Typically, this also involves measuring the acceptance of recommendations using click-through rates (CTR).

While the real-world setting is an advantage of online evaluations [135, 136], this very setting limits this experiment type to collecting user behavior on the platform (e.g., purchases, clicks, dwell, time). When inferring user satisfaction from user behavior [20, 22], care has to be taken, because user behavior (e.g., consumption activity) may also have different or additional causes such as integrated nudges [121], closing an app interpreted as negative feedback for an item [31], or biases due to interruptions or distractions [160].

We note that online evaluations require access to a RS and its implementation. Typically, online evaluations are carried out in the form of A/B testing [135] to compare the adapted system/algorithm to the original system. In so-called online field experiments [47], a small number of users are randomly assigned and exposed to different alternative RS configurations (instantiations) without their knowledge, and the users' interactions with the systems are recorded and analyzed. These instantiations may include different recommendation algorithms, and algorithm configurations, but also different interaction, presentation, or preference elicitation strategies.

Furthermore, online evaluations are performed for recommender systems that require a high amount of interaction with the user or where specifically the interaction with the user needs to be evaluated (e.g., critiquing systems [180], conversational recommender systems [48, 113], or novel interfaces and interaction strategies [30, 130]) that cannot be simulated are often evaluated in online field experiments. Traditionally, this includes A/B testing [135].

3.4 Evaluation Design Space: Evaluation Aspects

In this section, we provide an overview of individual aspects that are to be considered in the evaluation design space. Many of these aspects are interwoven, and their characteristics might have interdependencies or may be mutually exclusive. For instance, synthetic datasets come—by definition—without any user involvement. Experiments with random assignment of user groups to treatments (e.g., different RS algorithms) may be implemented in user studies (in randomized control trials or laboratory experiments) or online evaluation (in online field experiments) alike. Furthermore, trade-offs between RS performance indicators have been observed; for instance, a trade-off between accuracy and diversity is frequently reported [125, 143], and diversity may not necessarily be perceived by users [104, 133] or differently across users [143]. Moreover, situational factors may influence user experience due to varying user needs or preferences [61, 182].

Table 3. Characteristics of Explicit and Implicit Feedback (Adapted from Jawaheer et al. [120])

Dimension	Explicit Feedback	Implicit Feedback
Accuracy	High	Low
Abundance	Low	High
Expressivity of user preference	Positive and negative	Positive
Measurement reference	Absolute	Relative

Consequently, frameworks are an effective means to organize this complexity. For instance, Knijnenburg et al. [133]'s *framework for the user-centric evaluation of recommender systems* models this complexity for studies addressing the user experience.

In the following, we describe the individual aspects of the evaluation design space.

3.4.1 Types of Data. The essential basis for the evaluation of recommender systems is data. The characteristics of data can be manifold and may depend on the type of data used for computing the actual recommendations, among other factors. In the following, we give a brief overview of the different characteristics of data that may be used when evaluating RS.

Implicit and Explicit Rating Data. User ratings are usually collected by user behavior observations, which may, for instance, include records on the items that a user consumed, purchased, rated, viewed, or explored (e.g., pre-listening of songs), where the source may be an existing dataset or one that is collected for the respective study. When relying on the observation of user behavior when interacting with a RS, we typically distinguish between explicit and implicit feedback [105, 120]. Explicit feedback is provided directly by the user and the data unequivocally captures the user's perception of an item. Platforms that employ recommender systems frequently integrate mechanisms that allow users to explicitly express their interests in or preference for a specific item via rating scales (e.g., five-star rating scale, likes, thumbs-up, or thumbs-down). The rating scales used for providing explicit feedback usually allow for expressing both, positive and negative preferences (e.g., a scale from "I like it a lot" to "I do not like it").

Implicit feedback, in contrast, is inferred from a user's observable and measurable behavior when interacting with a RS (e.g., purchases, clicks, dwell time). When relying on implicit feedback, evaluations presume that, for instance, a consumed item is a quality choice, while all other items are considered irrelevant [15]. Hence, implicit feedback is typically positive only (e.g., purchase, click), while the absence of such information does not imply that the user does not like an item (e.g., a user not having listened to a track does not imply that the user does not like the track). Some scenarios also allow for opportunities for negative implicit feedback such as, for instance, the skipping of songs. Furthermore, implicit feedback can be used to infer relative preferences (for example, if a user watched one movie ten times whereas other movies typically only once or play counts of songs for a music RS). Thus, implicit feedback may be mapped to a degree of preferences, thereby ranging on a continuous scale to its positive extremity [120]. When interpreting implicit feedback, the assumption is that specific behavior is an indication of quality, regardless of whether the behavior may have other causes; thus, for example, closing a music streaming app may be mistakenly interpreted as a skip (i.e., negative feedback) [31] or the behavior is influenced by interruptions or distractions [56].

Most of the research in RS has focused on either explicit or implicit data [120], while comparably few have combined these two heterogeneous types of feedback (e.g., References [147, 151, 152]). Table 3 summarizes the characteristics of explicit and implicit feedback. Explicit feedback provides higher accuracy than implicit feedback inferred from behavior based on assumptions (e.g., the

assumption that users only click on items they are interested in). Typically, when users navigate through a platform that employs a RS, an abundance of data about user behavior is logged. In contrast, users are reluctant to explicitly rate items [100, 128], which leads to comparably little explicit feedback data. Note that explicit feedback tends to concentrate on either side of the rating scale, because users are more likely to express their preferences if they feel strongly in favor or against an item [12].

Although explicit and implicit feedback are heterogeneous types of feedback [120], research investigating the relations between implicit and explicit feedback for preference elicitation has shown that using implicit feedback is a viable alternative [165]. Still, implicit measures may reveal aspects that explicit measures do not [211]—particularly when user self-reports are not consistent with the actual user behavior. Integrating both, observation of actual user behavior and users' self-reports on intentions and perceptions, may deliver rich insights for which each approach in isolation would be insufficient.

Note that many evaluation designs presume that a consumed item is a viable option also in other contexts (e.g., another time, location, or activity) and consider item consumption as a generally valid positive implicit feedback. What the user indeed experiences, however, remains unclear. The validity of the feedback for other contexts depends on the design of the feedback mechanism. For instance, an item rated with five stars may be the user's lifetime favorite, but still not suitable for a certain occasion (e.g., a ballad for a workout, or a horror movie when watching with kids).

User, Item Information. RS algorithms typically heavily rely on rating data for the computation of recommendations, where the computations are mostly based solely on the user-item matrix. However, these approaches have been shown to suffer from sparsity and also, the cold-start problem, where recommendations for new items or users cannot be computed accurately as there is not enough information on the user or item, respectively. Therefore, metadata on the user, items, or context can also be incorporated to further enhance recommendations (this information is often referred to as "side information") [75, 162]. For instance, keywords describing the item may be extracted from, e.g., reviews on the item [10] or social ties between users can be extracted from relationships in social networks [154, 210]. Furthermore, when working toward business-oriented goals and metrics (cf. Section 3.4.4), data such as revenue information or click-through rates also have to be logged and analyzed [112]. In addition, context information is useful when users are expected to have different preferences in different contexts (e.g., watching a movie in a cinema or at home [187]).

Qualitative and Quantitative Data. Besides collecting behavioral user data (e.g., implicit feedback logged during user interactions with the system), evaluations may also rely on qualitative or quantitative evidence where data is gathered directly from the user. Quantitative data collection methods are highly-structured instruments—such as scales, tests, surveys, or questionnaires—which are typically standardized (e.g., same questions, same scales). This standardization facilitates validity and comparability across studies. Quantitative evidence allows for a deductive mode of analysis using statistical methods; answers may be compared and interrelated and allow for generalization to the population. Qualitative evidence is frequently deployed to understand the sample studied. Commonly used data collection methods include interviews, focus groups, and participant observations, where data is collected in the form of notes, videos, audio recordings, images, or text documents [80].

Natural and Synthetic Data. Herlocker et al. [101] distinguish between natural and synthetic datasets. While natural datasets capture user interactions with a RS or are directly derived from those, synthetic datasets are artificially created (e.g., References [58, 223]). Natural datasets contain (historical) data that may capture previous interactions of users with a RS (e.g., user behavior

such as clicks or likes), or data that may be associated with those (e.g., data that reflects users attitudes and feelings while interacting with a RS), or are derived from user interactions (e.g., turnover attributed to recommendations). In cases where a natural real-world dataset that would be sufficiently suitable for developing, training, and evaluating a RS is not available, a synthesized dataset may be used. In such cases, a synthesized dataset would allow for particularly modeling specific critical aspects that should be evaluated. For instance, a synthesized dataset may be created to reflect out-of-the-norm behavior. Herlocker et al. [101] stress that a synthetic dataset should only be used in the early stages of developing a RS and that synthesized datasets cannot simulate and represent real user behavior. Yet, not only user-behavior-related data can be synthesized. For instance, Jannach and Adomavicius [110] use fictitious profit values to investigate profitability aspects of RS.

3.4.2 Data Collection. Data collection methods may be distinguished based on their focus on considering contemporary and historical events, where methods may rely on past events (e.g., existing datasets, data retrieved from social media) or investigate contemporary events (e.g., observations, laboratory experiments) [221]. In the following, we give an overview of data collection aspects.

User Involvement. Evaluation methods may be distinguished with respect to user involvement. While offline studies do not require user interaction with a RS, user-centric evaluations need users to be involved, which is typically more expensive in terms of time and money [93, 189]—which is especially true for online evaluations with large user samples (cf. Section 3.3).

Randomized control trials are often considered the gold standard in behavioral science and related fields. In terms of RS evaluation, this means that users are recruited for the trial and randomly allocated to the RS to be evaluated (i.e., intervention) or to a standard RS (i.e., baseline) as the control. This procedure is also referred to as A/B-testing (e.g., References [54, 135, 136]). Randomized group assignment minimizes selection bias, keeping the participant groups that encounter an intervention or the baseline as similar as possible. Presuming that the environment can control for all the remaining variables (i.e., keeping the variables constant), the different groups allow for comparing the proposed system to the baselines. For instance, randomized control trials that are grounded on prior knowledge (e.g., observations or theory) [217] and where the factors measured (and the instruments used for measuring these factors) are carefully selected may help determine whether an intervention was effective [42]; explaining presumed causal links in real-world interventions is often too complex for experimental methods.

While randomized control trials are conducted in laboratory settings, experiments in field settings are typically referred to as "social experiments." Thereby, the term social experiment covers research in a field setting where investigators treat whole groups of people in different ways [221]. In online environments, this is referred to as online field experiment [47]. In field settings, the investigator's control is only partly possible. Field settings have the advantage that outcomes are observed in a natural, real-world environment rather than in an artificial laboratory environment—in the field, people are expected to behave naturally. Overall, though, field experiments are always less controlled than laboratory experiments, and field experiments are more difficult to replicate [150]. For RS evaluation, an online field experiment [47] very often requires collaboration with a RS provider from industry, who is commercially oriented and may not be willing to engage in risky interventions that may cause losing users and/or revenues. However, e.g., for the 2017 RecSys Challenge,[7] the best job recommendation approaches (determined by offline experiments) were also rolled out in XING's productive systems for online field experiments. Besides collaborating

[7] http://2017.recsyschallenge.com/.

with industry, a number of online field experiments have been carried out using research systems (e.g., MovieLens) (e.g., References [47, 227]). However, when carrying out a study with a research system, one also has to build a user community for it. Generally, this is often too great an investment just to carry out an experiment. This is why many researchers have argued for funding shared research infrastructure (in both Europe and the USA) including a system with actual users [137].

It is important to note that it is rarely feasible to repeat studies with user involvement for a substantially different set of algorithms and settings. System-centric (offline) evaluations are, in contrast, easily repeatable with varying algorithms [93, 101, 189]. However, offline evaluations have several weaknesses. For instance, data sparsity limits the coverage of items that can be evaluated. Also, the evaluation does not capture any explanations why a particular system or recommendation is preferred by a user (e.g., recommendation quality, aesthetics of the interface) [101]. Knijnenburg et al. [132, 133] propose a theoretical framework for user-centric evaluations that describes how users' personal interpretation of a system's critical features influences their experience and interaction with a system. In addition, Herlocker et al. [101] describe various dimensions that may be used to further differentiate user study evaluations. Examples for user-centric evaluations can, for instance, be found in References [51, 63, 70, 188].

Overall, while system-centric methods without user involvement typically aim to evaluate the RS from an algorithmic perspective (e.g., in terms of accuracy of predictions), user involvement opens up possibilities for evaluating user experience [189].

User Feedback Elicitation. At the core of many recommender systems are user preference models. Building such models requires eliciting feedback from users, for which—at runtime—data is typically collected while users interact with the RS. For evaluation purposes, we can leverage a wider variety of methods for data collection. For instance, besides considering interaction logs, observation [127] may be used to elicit users' behavior. An alternative method is to ask users for their behavior or intentions in a particular scenario. Such self-reports may be directed to reports on what they have done in the past or what users intend to do in a certain context. However, self-reports may not be consistent with user behavior [43, 141, 211], because the link between an individual's attitude and behavior is generally not very strong [7]. Furthermore, the process of reporting on one's behavior may itself induce reflection and actual change of behavior, which is known as the question-behavior effect [201]. It is, thus, good practice to combine self-report data with other information or to apply adjustment methods, because such an assessment considering several perspectives is more likely to provide an accurate picture [11].

For the elicitation of feedback on user experience, Pu et al. [172] propose an evaluation framework, called ResQue (Recommender systems' Quality of user experience) that aims to evaluate a comprehensive set of features of a RS: the system's usability, usefulness, interaction qualities, influence of these qualities on users' behavioral intentions, aspects influencing the adoption, and so on. ResQue provides specific questionnaire items and is, thus, considered highly operational. Knijnenburg et al. [133]'s framework for the user-centric evaluation of recommender systems takes a more abstract approach. It describes the structural relationships between the higher-level concepts without tying the concepts to specific questionnaire items. Therefore, it provides the flexibility to use and adapt the framework for various RS purposes and contextual settings and allows researchers to define and operationalize a set of specific, lower-level constructs. Both frameworks (i.e., Knijnenburg et al. [133] and Pu et al. [172]) may be integrated in user studies and online evaluations alike.

Existing Datasets. One advantage of relying on existing datasets is that (offline) evaluations can be conducted early in a project. In comparison to soliciting and evaluating contemporary events, it is frequently "easier" and less expensive in terms of money and time to rely on historical data [93].

Table 4. Widely Used Datasets for Evaluating RS

Dataset	Domain	Size
MovieLens20M[9] [97]	Movie ratings	20,000,263 ratings; range [0.5,5]
MovieLens1M[10] [97]	Movie ratings	1,000,209 ratings; range [1,5]
BookCrossing[11] [231]	Book ratings	1,157,112 ratings; range [1,10]
Yelp[12]	Business ratings	8,021,122 ratings; range [0,5]
MovieTweetings[13] [64]	Movie ratings	871,272 ratings; range [0,10]

Also, by utilizing popular datasets (e.g., the MovieLens dataset [97]), results can be compared with similar research. However, such an evaluation is restricted to the past. For instance, the goal of a leave-*n*-out analysis [32] is to analyze to which extent recommender algorithms can reconstruct past user interactions. Hence, such an evaluation can only serve as a baseline evaluation measure, because it only considers items that a user has already used in the past; assuming that unused items would not be used even if they were actually recommended [93]. Additional items that users might still consider useful are not considered in the evaluation, because ratings for these items are not contained in the dataset [224]. This is also stressed by Gunawardana et al. [93] by the following scenario: "For example, a user may not have used an item because she was unaware of its existence, but after the recommendation exposed that item the user can decide to select it. In this case, the number of false positives is overestimated."

Another risk is that the dataset chosen might not be (sufficiently) representative—the more realistic and representative the dataset is for real user behavior, the more reliable the results of the offline experiments are [93]. In fact, the applicability of the findings gained in an evaluation based on a historic dataset is highly impacted by the "quality, volume and closeness of the evaluation dataset to the data which would be collected by the intended recommender system" [81].

Table 4 lists datasets widely used for evaluating recommender systems and their main characteristics such as the domain, size, rating type, and examples of papers that have utilized the dataset in the evaluation of their system. There are different MovieLens datasets, differing in the number of ratings contained (from 100K ratings in the ML100K dataset to 20M ratings in the ML20M dataset; we list ML1M and ML20M in the table). Alternatively, the yearly conducted RecSys-Challenge[8] also provides datasets from a yearly changing application domain and task (including job, music, or accommodation (hotel) recommendation).

3.4.3 Data Quality and Biases. An important factor for RS evaluation is the quality of the data underlying the evaluations. This also includes potential biases that may be contained in the data used for the evaluation. Such biases may occur in the distributions of users, items, or ratings that are selected to be part of the evaluation dataset. As Gunawardana et al. [93] note, a typical example of a bias that is introduced when assembling the evaluation dataset is excluding users or items with low rate counts from the dataset. Careful curation of datasets by, e.g., using random sampling methods for limiting the size of the dataset to reduce the experimentation time is crucial to avoid such biases. Another aspect that may influence data biases is the collection method [93], where users do not provide feedback that is evenly distributed among items as, for instance, users

[8] http://www.recsyschallenge.com/.
[9] available for download at https://grouplens.org/datasets/movielens/.
[10] available for download at https://grouplens.org/datasets/movielens/.
[11] available for download at http://www2.informatik.uni-freiburg.de/~cziegler/BX/.
[12] available for download at https://www.yelp.com/dataset.
[13] available for download at https://github.com/sidooms/MovieTweetings.

tend to rate items that they particularly like or dislike. However, methods such as resampling or reweighting may be used for correcting such biases [203, 204].

Adomavicius and Zhang [5] investigated the characteristics of rating data and their impact on the overall recommendation performance. The characteristics they used for describing rating datasets are (i) overall rating density (i.e., the degree to which the user-item matrix is filled), (ii) rating frequency distribution (i.e., how ratings are distributed among items; rating data often exhibits a long-tail distribution [13, 164]), and (iii) the variance of rating values. In a set of experiments, the authors find that the recommendation performance is highly impacted by the structural characteristics of the dataset, where rating density and variance exhibit the highest impact.

3.4.4 Evaluation Metrics. There is an extensive number of facets of RS that may be considered when assessing the performance of a recommendation algorithm [92, 93]. Consequently, also the evaluation of RS relies on a diverse set of metrics, which we briefly summarize in the following. The presented metrics can be utilized for different experiment types, however, we note that due to the dominance of offline experiments, most of the presented metrics stem from offline settings.

In their early work on RS evaluation, Herlocker et al. [101] differentiate metrics for quantifying predictive accuracy, classification accuracy, rank accuracy, and prediction-rating correlation. Along the same lines, Gunawardana and Shani [92] investigate accuracy evaluation metrics and distinguish metrics based on the underlying task (rating prediction, recommending good items, optimizing utility, recommending fixed recommendation lists). Said et al. [189] classify the available metrics into classification metrics, predictive metrics, coverage metrics, confidence metrics, and learning rate metrics. In contrast, Avazpour et al. [16] provide a more detailed classification, distinguishing 15 classes of evaluation dimensions; these range, for instance, from correctness to coverage, utility, robustness, and novelty. Gunawardana et al. [93] distinguish prediction accuracy (rating prediction accuracy, usage prediction, ranking measures), coverage, novelty, serendipity, diversity, and confidence.[14] Chen and Liu [45] review evaluation metrics from four different perspectives (or rather, disciplines): machine learning (e.g., mean absolute error), information retrieval (e.g., recall or precision), human-computer interaction (e.g., diversity, trust, or novelty), and software engineering (e.g., robustness or scalability).

In the following, we discuss the most widely used categories of evaluation metrics. Table 5 gives an overview of these metrics, which we classify along the lines of previous classifications. For an extensive overview of evaluation metrics in the context of recommender systems, we refer to References [45, 87, 92, 93, 101, 166, 195]. Several works [185, 209] have shown that the metrics implemented in different libraries for RS evaluation (Section 3.4.5) sometimes use the same name while measuring different things, which leads to different results given the same input. Similarly, Bellogín and Said [24] report that papers present different variations of metrics (e.g., normalized vs non-normalized; computed over the entire dataset or on user-basis and then averaged); and sometimes the details of the evaluation protocol are not reported in papers [24, 36]. Tamm et al. [209] conclude that the more complex a metric is, the more room there is for different interpretations of the metric, leading to different variations of metric implementations. As a result, this might lead to misinterpretations of results within an evaluation [209], and limits the comparability across evaluations [24, 36, 185, 209]. In line with previous works [24, 36], we urge for a more detailed description of evaluation protocols as this will strengthen reproducibility and improve accountability [24].

[14] Gunawardana et al. [93] list further aspects that need to be evaluated, such as trust and risk, which are typically assessed via questionnaires. We do not cover these aspects here and kindly refer the interested reader to the original manuscript.

Table 5. Overview of Evaluation Metrics

Category	Metrics	References
Prediction accuracy	Mean absolute error (MAE)	[101, 197]
	(Root) Mean squared error ((R)MSE)	[101, 197]
Usage prediction	Recall, precision, F-score	[49, 213]
	Receiver operating characteristic curve (ROC)	[208]
	Area under ROC curve (AUC)	[17]
Ranking	Normalized discounted cumulative gain (NDCG)	[119]
	Mean reciprocal rank (MRR)	[215]
Novelty	Item novelty	[38]
	Global long-tail novelty	[40, 125]
Diversity	intra-list similarity (ILS)	[231]
Coverage	Item coverage	[87, 101]
	User space coverage	[87, 93]
	Gini index	[93]
Serendipity	Unexpectedness	[101]
	Serendipity	[125, 159]
Fairness across users	Value unfairness	[220]
	Absolute unfairness	[220]
	Over/underestimation of fairness	[220]
Fairness across items	Pairwise fairness	[26]
	Disparate treatment ratio (DTR)	[198]
	Equal expected exposure	[60]
	Equity of amortized attention	[27]
	Disparate impact ratio (DIR)	[198]
	Viable-Λ test	[191]
Business-oriented	Click-through rate (CTR)	[55, 86, 89]
	Adoption and conversion rate	[55, 89]
	Sales and revenue	[46, 145]
Articles providing an overview of metrics: [45, 87, 92, 93, 101, 166, 195].		

Fundamentally, we emphasize that it is important to evaluate a RS with a suite of metrics, because a one-metric evaluation will—in most cases—be one-sided and cannot characterize the broad performance of a RS. When optimizing a RS for one metric, it is crucial to also evaluate whether this optimization sacrifices performance elsewhere in the process [87, 101]. For instance, it is doubtful whether a RS algorithm optimized for prediction accuracy while sacrificing performance in terms of diversity, novelty, or coverage is overall desirable. Similarly, a RS that performs equally across various user groups but for all groups with similarly low accuracy and low diversity will not likely reach a good user experience for any user. It is, thus, crucial to measure—and report—a set of complementary metrics. In many cases, it will be key to find a good balance across metrics.

Prediction accuracy refers to the extent to which the RS can predict user ratings [93, 101]. These include error metrics that quantify the error of the rating prediction performed by the RS (i.e., the difference between the predicted rating and the actual rating in a leave-*n*-out setting). The most widely used prediction accuracy metrics are mean absolute error and root mean squared error.

Usage prediction metrics can be seen as classification metrics that capture the rate of correct recommendations—in a setting where each recommendation can be classified as relevant or non-relevant [92, 93, 101]. This involves binarizing ratings such as, e.g., on a rating scale of 1–5 considering ratings of 1–3 as non-relevant and ratings of 4 and 5 as relevant. The most popular usage prediction metrics are recall, precision, and the F-score, which combines recall

and precision. Precision is the fraction of recommended items that are also relevant. In contrast, recall measures the fraction of relevant items that are indeed recommended. Often, this includes restricting relevant items to the k most relevant items, where the system's ability to identify the k most suitable items for a user is captured as opposed to evaluating all recommendations (often referred to as recall@k or precision@k, respectively) [101]. Alternatively, the receiver operating characteristic curve can also be used to measure usage prediction, where the true positive rate is plotted against the false positive rate for various recommendation list lengths k. These curves can also be aggregated into a single score by computing the area under the ROC curve (AUC).

Ranking metrics are used to quantify the quality of the ranking of recommendation candidates [92, 166]. Relevant recommendations that are ranked higher are scored higher, whereas relevant documents that are ranked lower are provided a discounted score. Typical ranking metrics include normalized discounted cumulative gain (NDCG) [119], or mean reciprocal rank (MRR) [215].

Diversity refers to the dissimilarity of the items recommended [38, 125, 143, 214], where low similarity values mean high diversity. Diversity is often measured by computing the intra-list diversity [200, 231] and thereby, aggregating the pairwise similarity of all items on the recommendation list. Here, similarity can be computed, e.g., by Jaccard or cosine similarity [125].

Novelty metrics aim at measuring to which extent recommended items are novel [38]. Item novelty [107, 230] refers to the fraction of recommended items that are indeed new to the user, whereas global long-tail novelty measures the global novelty of items—i.e., if an item is known by few users and, hence, is in the long tail of the item popularity distribution [32, 40].

Serendipity describes how surprising recommendations are to a user and, hence, is tightly related to novelty [125, 159]. However, as Gunawardana et al. [93] note, recommending a movie staring an actor that the user has liked in the past might be novel, but not necessarily surprising to the user. The so-called unexpectedness measure compares the recommendations produced by a serendipitous recommender to the recommendations computed by a baseline [159]. Building on the unexpectedness measure, serendipity can be measured by the fraction of relevant and unexpected recommendations in the list [125] or the unexpectedness measure [2].

Coverage metrics describe the extent to which items are actually recommended [4, 87]. This includes catalog coverage (i.e., the fraction of all available items that can be recommended; often referred to as item space coverage) [189], user space coverage [93] (i.e., the fraction of items that are recommended to a user; often also referred to as prediction coverage [87]), or measuring the distribution of items chosen by users (e.g., by using the Gini index or Shannon entropy) [93]. Coverage metrics are also used to measure fairness, because coverage captures the share of items or users that are served by the RS.

Fairness metrics concern both, fairness across users and across items. In both cases, fairness may be captured at the level of the individual or at group level. Individual fairness captures fairness (or unfairness) at the level of individual subjects [27] and implies that similar subjects (hence, similar users or similar items) are treated similarly [65]. Group fairness defines fairness on a group level and requires that salient subject groups (e.g., demographic groups) should be treated comparably [66]; in other words, group fairness is defined as the collective treatment received by all members of a group [27]. A major goal of group fairness is that protected attributes—for instance, demographic traits such as age, gender, or ethnicity—do not influence recommendation outcomes due to data bias or model inaccuracies and biases [27, 196].

Fairness across users is typically addressed at the group level. One way to address group fairness from the user perspective is to disaggregate the user-oriented metrics to measure and compare to which extent user groups are provided with lower-quality recommendations (e.g., References [69, 73, 74, 108, 142, 158, 196]). Yao and Huang [220] propose three (un-)fairness metrics:

value unfairness measures, whether groups of users receive constantly lower or higher predicted ratings compared to their true preference; absolute unfairness measures the absolute difference of the estimation error for groups, and under/overestimation of fairness measures inconsistency in the extent to which predictions under- or overestimate the true ratings.

Fairness across items addresses the fair representation of item groups [27] and it is addressed at group level and at the level of individual items, too. The goal of many metrics is to measure the exposure or attention [27, 198] an item group receives and assess the fairness of this distribution: in a ranked list of recommendations, lower ranks are assumed to get less exposure and, thus, less attention.[15] Beutel et al. [26] propose the concept of pairwise fairness, which aims to measure whether items of one group are consistently ranked lower than those of another group. Other metrics put exposure across groups and relevance of items into relation. The disparate treatment ratio (DTR) [198] is a statistical parity metric that measures exposure across groups proportional to relevance. Diaz et al. [60] consider the distribution over rankings instead of a single fixed ranking. The idea behind the principle of equal expected exposure is that "no item should receive more or less expected exposure than any other item of the same relevance grade" [60]. Biega et al. [27] capture unfairness at the level of individual items; they propose the equity of amortized attention, which indicates whether the attention is distributed proportionally to relevance when amortized over a sequence of rankings. The disparate impact ratio (DIR) [198] goes further than exposure and considers the impact of exposure: DIR measures across items groups, whether items obtain proportional impact in terms of the click-through rate. The viable-Λ test [191] accounts for varying user attention patterns through parametrization in the measurement of group fairness across items.

Business-oriented metrics are used by service providers to assess the business value of recommendations [112]. While service providers naturally are interested in user-centered metrics as positive user experience impacts revenue, business-oriented metrics allow to directly measure click-through-rates [55, 86, 89, 126], adoption and conversion rates [55, 89], and revenue [46, 145]. Click-through rates measure the number of clicks generated by recommendations, whereas adoption and conversion rates measure how many clicks actually lead to the consumption of recommended items. Therefore, adoption and conversion rates, and even more so, the sales and revenue generated by recommended items, more directly measure the generated business value of recommendations.

3.4.5 Evaluation System. Involving users in evaluations requires a (usually graphical) user interface to allow users to interact with the system. In RS evaluation, different options are available concerning the extent to which the evaluated system is incorporated in a real-world or industry environment. This aspect is highly interwoven with the choice of whether to involve users in the evaluation. For an offline algorithmic evaluation, there is no need to provide a user interface, as no users are involved. However, measuring user experience requires the involvement of users and, hence, a user interface. Konstan and Riedl [138] distinguish three designs of systems for evaluation: (i) systems dedicated to experimental use, which may range from interfaces for purely experimental research to more sophisticated systems; (ii) collaborating with operators of real-world (industry) systems for online field (real-world) experiments; and (iii) developing and maintaining a research system and (large) user community for (long-term) evaluations.

Also, "bad" user interface design may bias the assessment of RSs, because they affect the users' overall experience [50, 173]. Users may evaluate recommendations differently if they were

[15] While many approaches assume logarithmic discounting of attention [198], also other approaches exist, too (for example, using a geometric distribution [27] or parametrizing varying attention patterns [191]).

presented by an improved user interface. Putting effort into a good (or neutral) user interface design is expensive. Maintenance costs for a dedicated research system are high, too. Likewise, acquiring a large set of users may be challenging. All these issues contribute to the low adoption of non-offline evaluations.

Generally, there are several RS evaluation frameworks. Most of these libraries are primarily for offline evaluations and, hence, provide a set of recommender algorithms and an evaluation framework. These frameworks include, for instance, LensKit [68, 72], MyMediaLite [84], LibRec [94], Rival [185, 186], Surprise [106], or Elliot [14]. Recently, Beel et al. [19] proposed a "living-lab" for online evaluations of scholarly recommender systems that can be used on top of a production recommender system and logs all user actions (clicks, purchases, etc.) to evaluate the algorithms' effectiveness in online evaluations and user studies.

4 MAPPING A FICTITIOUS CASE TO FEVR

In the following, we first present a fictitious case as an example evaluation. Then, we showcase how this scenario can be mapped to the FEVR framework (Section 4.1) and discuss the limitations of this evaluation configuration (Section 4.2).

The context of the example is as follows: in an academic setting, a group of researchers has developed a novel recommendation algorithm, termed *RecAlg*, that aims to improve the item diversity of music recommendations by incorporating audio and lyrics features of tracks, while also improving (or, at least, maintaining) prediction accuracy. The goal is to support users in finding likable music by providing personalized music recommendations.

4.1 Mapping to FEVR

With this example case, we revisit the major components of the proposed FEVR framework and discuss the design components regarding the evaluation of the RS. We provide a compact overview of the design components of the example case in Table 6. Note that the evaluation principles component draws from the other components and is discussed at the end of this section.

As for the *evaluation objectives*, the overall goal is to evaluate whether users are indeed able to find likable music when provided with recommendations computed by the novel RecAlg algorithm. The stakeholders addressed are naturally the users of the system (algorithm), but—as the proposed algorithm aims to improve the diversity of recommended tracks—artists could also benefit from this increased item diversity as a more diverse set of artists may now be represented in the set of recommended tracks. As for the properties evaluated, the researchers aim to evaluate the diversity of recommendations; particularly, the change in catalog coverage (and, hence, the change to items in the long tail of the popularity curve). FEVR's evaluation design space (cf. Figure 2 for a graphical overview of the core components) encompasses the main evaluation principles, which we discuss in the following. As *experiment type*, the group of researchers chooses to perform an offline evaluation to assess the basic algorithmic performance of RecAlg (that still needs to be confirmed in later user-centric evaluations to evaluate whether users do indeed also perceive the provided recommendations as more diverse and accurate).

The *evaluation aspects* that need to be considered in this evaluation encompass the data to be used but also the evaluation system and the evaluation metrics applied. As this example is situated in a scientific setting, offline experiments can be performed using an existing evaluation framework. In this particular case, the Elliot [14] framework is chosen. As for the data used for the evaluation, the researchers rely on an extensive dataset of listening events, namely, the LFM-2b dataset [194]. This dataset contains 2 billion listening events (i.e., a user has listened to a particular song), which represent implicit feedback, as well as detailed side information on the music tracks contained. LFM-2b is the most extensive and recent public dataset in the domain. In the

Table 6. FEVR: Overview of Example Evaluation

FEVR Component	Brief Description
Evaluation Objectives	
Overall Goal	To evaluate whether users are able to find likable music in the recommendations computed by the novel RecAlg algorithm
Stakeholders	Users of the system (algorithm) Artists may also benefit from an increased item diversity as a more diverse set of artists may be represented
Properties	Item diversity in the recommendations; catalog coverage
Evaluation Principles	
Hypothesis/Research Question	H_1: RecAlg provides users (on average) with more diverse recommendations with respect to the intra-list diversity while maintaining prediction accuracy compared to the baseline algorithm.
Control Variables	Follow accountability framework by Bellogín and Said [24] (for randomization in dataset splitting to prevent selection bias)
Generalization Power	Limited due to lack of user involvement and dataset biases
Reliability	Follow accountability framework by Bellogín and Said [24]
Experiment Type	Offline Evaluation with A/B-testing
Evaluation Aspects	
Types of Data	Implicit ratings (listening events), side information for music tracks
Data Collection	LFM-2b dataset [194]
Data Quality and Biases	Platform bias, popularity bias, skewed gender distribution, imbalanced country distribution
Evaluation Metrics	Prediction accuracy with RMSE; intra-list similarity in terms of different unique artists
Evaluation System	Existing evaluation framework Elliot [14]

experimental setup, users for the training, test, and validation sets are chosen randomly to avoid introducing biases in this step. The metrics employed directly reflect the goals of our evaluation: for quantifying RecAlg's prediction accuracy, the group of researchers rely on RMSE and for measuring the diversity of recommendation lists, they rely on intra-list similarity. As for the *evaluation principles*, the main hypothesis is that the novel RecAlg approach provides users with more diverse recommendations concerning the intra-list similarity while maintaining prediction accuracy. The generalization power of this evaluation is limited in the sense that it does not involve users, the dataset used encompasses biases, and the fact that implicit feedback data was used. For a further discussion on the limitations of this evaluation, please refer to Section 4.2. To ensure the reliability of the conducted experiments and to control for confounding factors, the group of researchers follows the accountability framework by Bellogín and Said [24].

With this example evaluation scenario, we have illustrated how an evaluation configuration can be mapped to FEVR and demonstrated that FEVR can be used as a checklist for an evaluation configuration. However, we note that this described evaluation scenario is a very basic one and has limitations, which we discuss in the following (Section 4.2).

4.2 Limitations and Discussion

As for any kind of offline evaluation with a publicly available dataset, the generalization power is limited due to the inherent biases in the dataset. First and foremost, there is a platform bias and evaluation results would not generalize to other music streaming platforms or even to other application domains. This and other dataset biases (e.g., skewed gender distribution of users, imbalanced distribution across user countries) may be addressed by extending the evaluation by integrating

further datasets. Comparing evaluation results across datasets provides the opportunity to reason across all results and, thereby, increases the generalizability of findings.

Furthermore, the presented example evaluation builds on assumptions that are—at least in the scenario presented—not grounded on prior knowledge and not justified with respective pointers to underlying theory or observations. Many assumptions concern a user's need for diversity and their perceived diversity. The evaluation setting is built on a set of assumptions including that users indeed enjoy or even want artist diversity in their playlists, that all users have similar diversity needs, that an individual's diversity need is constant (i.e., context-independent), and that individuals perceive the provided recommendations as diverse as the intra-list similarity metric suggests. With a lack of literature on those topics, it is necessary to integrate additional methods in the evaluation to explore and clarify these assumptions (and to obtain a more comprehensive picture of the experiment's results). Frequently, this will require a mixed-methods research approach [53] where quantitative and qualitative research methods are combined. A recent tutorial at the ACM SIGKDD Conference on Knowledge Discovery and Data Mining 2021 [202][16] demonstrates how a mixed-methods approach is used in real-world (industry) settings (specifically, the tutorial presents case studies from Spotify) to analyze and justify assumptions, develop business-oriented metrics, so that further evaluation steps are valid and reliable.

Finally, the results of the presented evaluation will give direction about the next evaluation steps. Hence, the evaluation results will inform whether the novel RecAlg algorithm achieves sufficient performance to be further evaluated in, for instance, a user study (e.g., by particularly considering diversity perception). Unsatisfactory results will suggest revisiting the algorithm and exploring further opportunities. Again, FEVR can serve as a checklist for the configuration of the next evaluation step.

5 DISCUSSION, CONCLUSION, AND FUTURE DIRECTIONS

The review of literature on RS evaluation shows that finding an adequate configuration for the comprehensive evaluation of a RS is a complex endeavor; the evaluation design space is rich, and finding an adequate configuration may be challenging. In this article, we consolidate and systematically organize the dispersed knowledge on RS evaluation. With FEVR, we provide a basis, overview, and guidance for researchers as a profound source for orientation in evaluating RS.

Still, for RS to work in practice (i.e., in industry) as well as for the research community to advance, we have to engage in a more comprehensive evaluation of RS—an evaluation that embraces the entire RS and its context of use and does not only address single dimensions in isolation. Yet, to date, such a comprehensive evaluation approach is hardly adopted in RS research.

From a practical perspective, the reasons for the low adoption of comprehensive evaluation—and the excessive use of offline evaluation only—are manifold [39]: (a) identifying an adequate combination of evaluation designs and configurations (more broadly speaking, aspects that can and need to be addressed together) meeting the evaluation objectives may be a complex task (particularly for inexperienced researchers); (b) the costs for involving users in the evaluation process are high (compared to pure offline studies); (c) integrating results of multiple evaluation designs and configurations into an entire study is complex and drawing conclusions from components effectively across the entire study can be challenging; and (d) evaluations considering multiple methods require adequate skills in various (at least two) evaluation methods. Senior researchers tend to have a preference for one method [171] and apply methodologically what they are strong at, which also prevents young researchers from learning (and possibly adopting) additional methodical

[16]The slides of the tutorial can be found at https://github.com/kdd2021-mixedmethods. A similar case study has already been presented at the tutorial on "Mixed methods for evaluating user satisfaction" at ACM RecSys 2018 [85].

approaches. While these reasons for non-adoption are all plausible, we argue that the goal should be to use the most adequate evaluation setting for set evaluation objectives. In many cases, this will require an integration of multiple evaluation designs. This comes with several challenges:

- *Methodological issues.* Jannach et al. [114] point to methodological issues and research practices in RS evaluation where novel recommender approaches are compared to weak (e.g., non-optimized) baselines [77, 78, 153]. Showing "phantom progress," as Ludewig et al. [153] term it, hamper the progress of research and is of little value for evaluating the recommender approach under investigation. Along this line comes the need for good evaluation protocols that are documented in papers with sufficient detail [24, 36] to strengthen reproducibility. Yet, using many different metric variants—even if properly documented—hinders the comparability across works. Accordingly, the development and establishment of standardized protocols is a core issue that the community needs to address for advancing the field.
- *Methodological competencies.* Employing a comprehensive RS evaluation requires researchers to build competencies in a set of methods as expertise in only one method is insufficient. Furthermore, consolidating these methods' results into an integrated picture of the system's quality and the perceived quality of the RS is another skill set that has to be developed.
- *Datasets.* A crucial task is to find or elicit datasets that are sufficiently representative of the use case that the RS is evaluated for. Reducing biases inherent in real-world data is considered one of the key challenges [28]. Furthermore, Jannach and colleagues [114, 115] call for the evaluation of RS' longitudinal effects. One of the challenges involved is to obtain a rich dataset over a long period of time in a comparable manner. With the fast-paced progress in RS research, RS approaches are continually being updated and fine-tuned and datasets embracing a longer period do likely encompass dynamics of having different RS approaches active at different times.
- *Multi-stakeholder RS.* Research on multi-stakeholder RS is currently still in its infancy. For evaluation, we can observe "a diversity of methodological approaches and little agreement on basic questions of evaluation" [1].
- *Conversational RS.* Although conversational RS seem to advance at an accelerated pace, no consensus on how to evaluate such systems has evolved yet [113]. For instance, conversational RS rely on natural language processing (NLP), and evaluating language models and generation models is itself an inherently complex task [113]. Using and evaluating such models in task-oriented systems such as conversational RS might be even more challenging [113].
- *Domain-specifics.* The quality of recommendations depends on the particular domain or application. For a news recommender, the recency of items is important. In the music domain, recommenders are often considered useful when they support discovery of the back catalog. In tourism, the geographical vicinity might be relevant. The evaluation configuration has to take such domain-specifics into account [111]. This requires deep domain knowledge (and data), which frequently requires collaborating with domain experts in academia and industry. Evaluation without domain expertise bears the risk of being based on wrong assumptions.
- *Multi-* evaluations.* Comprehensive evaluations encompassing the required multi-facetedness (e.g., multi-method, multi-metrics, multi-stakeholder) appears to be an adequate and necessary pathway for RS evaluation. The key issue is that we need to establish evaluations that are apt to characterize the broad performance of a RS, which can only be accomplished with thoughtful integration of multiple methods. This requires an evaluation culture where a suite of metrics is evaluated and reported, and where the needs of the multiple stakeholders of RS are considered. The hurdles of such evaluations, including involved costs, required skills, and so on, are—undeniably—impediments we need to take

up and overcome these challenges to advance recommender systems and the field of recommender systems at large. Yet, this seems to require a paradigm shift in our research community's evaluation efforts [111].

While FEVR framework provides a structured basis to adopt adequate evaluation configurations, we—as a community—have to move forward together: it is on us to adopt, apply, and establish suitable practices.

REFERENCES

[1] Himan Abdollahpouri, Gediminas Adomavicius, Robin Burke, Ido Guy, Dietmar Jannach, Toshihiro Kamishima, Jan Krasnodebski, and Luiz Pizzato. 2020. Multistakeholder recommendation: Survey and research directions. *User Model. User-Adapt. Interact.* 30, 1 (2020), 127–158. https://doi.org/10.1007/s11257-019-09256-1

[2] Panagiotis Adamopoulos and Alexander Tuzhilin. 2015. On unexpectedness in recommender systems: Or how to better expect the unexpected. *ACM Trans. Intell. Syst. Technol.* 5, 4, Article 54 (2015). https://doi.org/10.1145/2559952

[3] Gediminas Adomavicius, Konstantin Bauman, Alexander Tuzhilin, and Moshe Unger. 2022. Context-aware recommender systems: From foundations to recent developments. In *Recommender Systems Handbook* (3rd ed.), Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, New York, NY, 211–250. https://doi.org/10.1007/978-1-0716-2197-4_6

[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. *IEEE Trans. Knowl. Data Eng.* 17, 6 (2005), 734–749. https://doi.org/10.1109/tkde.2005.99

[5] Gediminas Adomavicius and Jingjing Zhang. 2012. Impact of data characteristics on recommender systems performance. *ACM Trans. Manage. Info. Syst.* 3, 1, Article 3 (2012). https://doi.org/10.1145/2151163.2151166

[6] Aman Agarwal, Kenta Takatsu, Ivan Zaitsev, and Thorsten Joachims. 2019. A general framework for counterfactual learning-to-rank. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'19)*. ACM, 5–14. https://doi.org/10.1145/3331184.3331202

[7] Icek Ajzen and Martin Fishbein. 1977. Attitude-behavior relations: A theoretical analysis and review of empirical research. *Psychol. Bull. J.* 84, 5 (1977), 888–918. https://doi.org/10.1037/0033-2909.84.5.888

[8] Anas Bassam AL-Badareen, Mohd Hasan Selamat, Jamilah Din, Marzanah A. Jabar, and Sherzod Turaev. 2011. Software quality evaluation: User's view. *Int. J. Appl. Math. Info.* 5, 3 (2011), 200–207.

[9] Muhammad Aljukhadar, Sylvain Senecal, and Charles-Etienne Daoust. 2012. Using recommendation agents to cope with information overload. *Int. J. Electr. Comm.* 17, 2 (2012), 41–70. https://doi.org/10.2753/jec1086-4415170202

[10] Amjad Almahairi, Kyle Kastner, Kyunghyun Cho, and Aaron Courville. 2015. Learning distributed representations from reviews for collaborative filtering. In *Proceedings of the 9th ACM Conference on Recommender Systems (RecSys'15)*. ACM, 147–154. https://doi.org/10.1145/2792838.2800192

[11] Alaa Althubaiti. 2016. Information bias in health research: Definition, pitfalls, and adjustment methods. *J. Multidisc. Healthcare* 9 (2016), 211. https://doi.org/10.2147/jmdh.s104807

[12] 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 ACM Conference on Recommender Systems (RecSys'09)*. ACM, 247–258. https://doi.org/10.1145/1639714.1639744

[13] Chris Anderson. 2007. *The Long Tail: How Endless Choice is Creating Unlimited Demand*. Random House.

[14] Vito Walter Anelli, Alejandro Bellogín, Antonio Ferrara, Daniele Malitesta, Felice Antonio Merra, Claudio Pomo, Francesco Maria Donini, and Tommaso Di Noia. 2021. Elliot: A comprehensive and rigorous framework for reproducible recommender systems evaluation. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'21)*. ACM, 2405–2414. https://doi.org/10.1145/3404835.3463245

[15] Juan Arévalo, Juan Ramón Duque, and Marco Creatura. 2018. A missing information loss function for implicit feedback datasets. Retrieved from https://arXiv:1805.00121.

[16] Iman Avazpour, Teerat Pitakrat, Lars Grunske, and John Grundy. 2013. Dimensions and metrics for evaluating recommendation systems. In *Recommendation Systems in Software Engineering*. Springer, Berlin, 245–273. https://doi.org/10.1007/978-3-642-45135-5_10

[17] Donald Bamber. 1975. The area above the ordinal dominance graph and the area below the receiver operating characteristic graph. *J. Math. Psychol.* 12, 4 (1975), 387–415. https://doi.org/10.1016/0022-2496(75)90001-2

[18] Christine Bauer and Eva Zangerle. 2019. Leveraging multi-method evaluation for multi-stakeholder settings. In *Proceedings of the 1st Workshop on the Impact of Recommender Systems co-located with 13th ACM Conference on Recommender Systems (ImpactRS'19)*, Vol. 2462. CEUR-WS.org, Article 3. Retrieved from http://ceur-ws.org/Vol-2462/short3.pdf.

[19] Joeran Beel, Andrew Collins, Oliver Kopp, Linus W. Dietz, and Petr Knoth. 2019. Online evaluations for everyone: Mr. DLib's living lab for scholarly recommendations. In *Advances in Information Retrieval*. Springer, 213–219.

[20] Joeran Beel, Marcel Genzmehr, Stefan Langer, Andreas Nürnberger, and Bela Gipp. 2013. A comparative analysis of offline and online evaluations and discussion of research paper recommender system evaluation. In *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation (RepSys'13)*. ACM, 7–14. https://doi.org/10.1145/2532508.2532511

[21] Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. 2015. Research paper recommender systems: A literature survey. *Int. J. Dig. Libr.* 17, 4 (2015), 305–338. https://doi.org/10.1007/s00799-015-0156-0

[22] Joeran Beel and Stefan Langer. 2015. A comparison of offline evaluations, online evaluations, and user studies in the context of research-paper recommender systems. In *Proceedings of the International Conference on Theory and Practice of Digital Libraries*. Springer, 153–168. https://doi.org/10.1007/978-3-319-24592-8_12

[23] Joeran Beel, Stefan Langer, Marcel Genzmehr, Bela Gipp, Corinna Breitinger, and Andreas Nürnberger. 2013. Research paper recommender system evaluation: A quantitative literature survey. In *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation (RepSys'13)*. ACM, 15–22. https://doi.org/10.1145/2532508.2532512

[24] Alejandro Bellogín and Alan Said. 2021. Improving accountability in recommender systems research through reproducibility. *User Model. User-Adapt. Interact.* 31 (2021). https://doi.org/10.1007/s11257-021-09302-x

[25] Rianne van den Berg, Thomas N. Kipf, and Max Welling. 2018. Graph convolutional matrix completion. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'18)*. Retrieved from https://www.kdd.org/kdd2018/files/deep-learning-day/DLDay18_paper_32.pdf.

[26] Alex Beutel, Jilin Chen, Tulsee Doshi, Hai Qian, Li Wei, Yi Wu, Lukasz Heldt, Zhe Zhao, Lichan Hong, Ed H. Chi, and Cristos Goodrow. 2019. Fairness in recommendation ranking through pairwise comparisons. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'19)*. ACM, 2212–2220. https://doi.org/10.1145/3292500.3330745

[27] Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. 2018. Equity of attention: Amortizing individual fairness in rankings. In *Proceedings of the 41st International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'18)*. ACM, 405–414. https://doi.org/10.1145/3209978.3210063

[28] Toine Bogers, Marijn Koolen, Bamshad Mobasher, Casper Petersen, and Alexander Tuzhilin. 2020. Report on the fourth workshop on recommendation in complex environments (ComplexRec 2020). *SIGIR Forum* 54, 2, Article 14 (2020). https://doi.org/10.1145/3483382.3483397

[29] Philip Bonhard, Clare Harries, John McCarthy, and M. Angela Sasse. 2006. Accounting for taste: Using profile similarity to improve recommender systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'06)*. ACM, 1057–1066. https://doi.org/10.1145/1124772.1124930

[30] Svetlin Bostandjiev, John O'Donovan, and Tobias Höllerer. 2012. TasteWeights: A visual interactive hybrid recommender system. In *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys'12)*. ACM, 35–42. https://doi.org/10.1145/2365952.2365964

[31] Klaas Bosteels, Elias Pampalk, and Etienne E. Kerre. 2009. Evaluating and analysing dynamic playlist generation heuristics using radio logs and fuzzy set theory. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR'09)*. International Society for Music Information Retrieval, 351–356.

[32] John S. Breese, David Heckerman, and Carl Kadie. 1998. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI'98)*. Morgan Kaufmann Publishers, 43–52. Retrieved from https://dl.acm.org/doi/10.5555/2074094.2074100.

[33] Robin Burke. 2002. Hybrid recommender systems: Survey and experiments. *User Model. User-Adapt. Interact.* 12, 4 (2002), 331–370.

[34] Robin Burke. 2010. Evaluating the dynamic properties of recommendation algorithms. In *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys'10)*. ACM, 225–228. https://doi.org/10.1145/1864708.1864753

[35] Robin Burke, Himan Abdollahpouri, Bamshad Mobasher, and Trinadh Gupta. 2016. Towards multi-stakeholder utility evaluation of recommender systems. In *Proceedings of the 1st Workshop on Surprise, Opposition, and Obstruction in Adaptive and Personalized Systems (SOAP'16)*, Vol. 1618. CEUR-WS.org, Article 2. Retrieved from http://ceur-ws.org/Vol-1618/SOAP_paper2.pdf.

[36] Rocío Cañamares, Pablo Castells, and Alistair Moffat. 2020. Offline evaluation options for recommender systems. *Info. Retriev. J.* 23, 4 (2020), 387–410. https://doi.org/10.1007/s10791-020-09371-3

[37] Donald T. Campbell. 1957. Factors relevant to the validity of experiments in social settings. *Psychol. Bull.* 54, 4 (1957), 297–312. https://doi.org/10.1037/h0040950

[38] Pablo Castells, Neil Hurley, and Saúl Vargas. 2022. Novelty and diversity in recommender systems. In *Recommender Systems Handbook* (3rd ed.), Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, New York, NY, 603–646. https://doi.org/10.1007/978-1-0716-2197-4_16

[39] Ilknur Celik, Ilaria Torre, Frosina Koceva, Christine Bauer, Eva Zangerle, and Bart Knijnenburg. 2018. UMAP 2018 intelligent user-adapted interfaces: Design and multi-modal evaluation (IUadaptMe) workshop chairs' welcome and organization. In *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization (UMAP'18)*. ACM, 137–139. https://doi.org/10.1145/3213586.3226202

[40] Òscar Celma and Perfecto Herrera. 2008. A new approach to evaluating novel recommendations. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'08)*. ACM, 179–186. https://doi.org/10.1145/1454008.1454038

[41] Òscar Celma. 2010. *Music Recommendation and Discovery*. Springer, Berlin. https://doi.org/10.1007/978-3-642-13287-2

[42] Thomas C. Chalmers, Harry Smith, Bradley Blackburn, Bernard Silverman, Biruta Schroeder, Dinah Reitman, and Alexander Ambroz. 1981. A method for assessing the quality of a randomized control trial. *Control. Clin. Trials* 2, 1 (1981), 31–49. https://doi.org/10.1016/0197-2456(81)90056-8

[43] Yu-Long Chao and San-Pui Lam. 2009. Measuring responsible environmental behavior: Self-reported and other-reported measures and their differences in testing a behavioral model. *Environ. Behav.* 43, 1 (2009), 53–71. https://doi.org/10.1177/0013916509350849

[44] Chien Chin Chen, Shun-Yuan Shih, and Meng Lee. 2016. Who should you follow? Combining learning to rank with social influence for informative friend recommendation. *Decision Supp. Syst.* 90 (2016), 33–45. https://doi.org/10.1016/j.dss.2016.06.017

[45] Mingang Chen and Pan Liu. 2017. Performance evaluation of recommender systems. *Int. J. Perform. Eng.* 13, 8 (2017), 1246–1256. https://doi.org/10.23940/ijpe.17.08.p7.12461256

[46] Pei-Yu Chen, Yen-Chun Chou, and Robert J. Kauffman. 2009. Community-based recommender systems: Analyzing business models from a systems operator's perspective. In *Proceedings of the 42nd Hawaii International Conference on System Sciences (HICSS'09)*. IEEE. https://doi.org/10.1109/hicss.2009.117

[47] Yan Chen and Joseph Konstan. 2015. Online field experiments: A selective survey of methods. *J. Econ. Sci. Assoc.* 1, 1 (2015), 29–42. https://doi.org/10.1007/s40881-015-0005-3

[48] Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016. Towards conversational recommender systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'16)*. ACM, 815–824. https://doi.org/10.1145/2939672.2939746

[49] Cyril W. Cleverdon, Jack Mills, and E. Michael Keen. 1966. *Factors Determining the Performance of Indexing Systems, (Volume 1: Design)*. Technical Report, College of Aeronautics. Retrieved from http://hdl.handle.net/1826/861.

[50] 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 SIGCHI Conference on Human Factors in Computing Systems (CHI'03)*. ACM, 585–592. https://doi.org/10.1145/642611.642713

[51] Paolo Cremonesi, Franca Garzotto, Sara Negro, Alessandro Vittorio Papadopoulos, and Roberto Turrin. 2011. Looking for "good" recommendations: A comparative evaluation of recommender systems. In *Proceedings of the Conference on Human-Computer Interaction (INTERACT'11)*. Springer, Berlin, 152–168. https://doi.org/10.1007/978-3-642-23765-2_11

[52] Paolo Cremonesi, Roberto Turrin, Eugenio Lentini, and Matteo Matteucci. 2008. An evaluation methodology for collaborative recommender systems. In *Proceedings of the International Conference on Automated Solutions for Cross Media Content and Multi-Channel Distribution (AXMEDIS'08)*. IEEE, 224–231. https://doi.org/10.1109/axmedis.2008.13

[53] John W. Creswell. 2003. *Research Design: Qualitative, Quantitative, and Mixed Methods Approaches* (2nd ed.). SAGE.

[54] Thomas Crook, Brian Frasca, Ron Kohavi, and Roger Longbotham. 2009. Seven pitfalls to avoid when running controlled experiments on the web. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09)*. ACM, 1105–1114. https://doi.org/10.1145/1557019.1557139

[55] James Davidson, Benjamin Liebald, Junning Liu, Palash Nandy, Taylor Van Vleet, Ullas Gargi, Sujoy Gupta, Yu He, Mike Lambert, Blake Livingston, and Dasarathi Sampath. 2010. The YouTube video recommendation system. In *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys'10)*. ACM, 293–296. https://doi.org/10.1145/1864708.1864770

[56] Marco de Gemmis, Pasquale Lops, Cataldo Musto, Fedelucio Narducci, and Giovanni Semeraro. 2015. Semantics-aware content-based recommender systems. In *Recommender Systems Handbook* (2nd ed.). Springer US, New York, NY, 119–159. https://doi.org/10.1007/978-1-4899-7637-6_4

[57] Zohreh Dehghani Champiri, Adeleh Asemi, and Salim Siti Salwah Binti. 2019. Meta-analysis of evaluation methods and metrics used in context-aware scholarly recommender systems. *Knowl. Info. Syst.* 61, 2 (2019), 1147–1178. https://doi.org/10.1007/s10115-018-1324-5

[58] María del Carmen-Rodríguez-Hernández, Sergio Ilarri, Ramón Hermoso, and Raquel Trillo-Lado. 2017. DataGen-CARS: A generator of synthetic data for the evaluation of context-aware recommendation systems. *Pervas. Mobile Comput.* 38 (2017), 516–541. https://doi.org/10.1016/j.pmcj.2016.09.020

[59] Yashar Deldjoo, Mehdi Elahi, Paolo Cremonesi, Franca Garzotto, Pietro Piazzolla, and Massimo Quadrana. 2016. Content-based video recommendation system based on stylistic visual features. *J. Data Semant.* 5, 2 (2016), 99–113. https://doi.org/10.1007/s13740-016-0060-9

[60] Fernando Diaz, Bhaskar Mitra, Michael D. Ekstrand, Asia J. Biega, and Ben Carterette. 2020. Evaluating stochastic rankings with expected exposure. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM'20)*. ACM, 275–284. https://doi.org/10.1145/3340531.3411962

[61] Peter R. Dickson. 1982. Person-situation: Segmentation's missing link. *J. Market.* 46, 4 (1982), 56. https://doi.org/10.2307/1251362

[62] Karlijn Dinnissen and Christine Bauer. 2022. Fairness in music recommender systems: A stakeholder-centered mini review. *Front. Big Data* 5, Article 913608 (2022). https://doi.org/10.3389/fdata.2022.913608

[63] Simon Dooms, Toon De Pessemier, and Luc Martens. 2011. A user-centric evaluation of recommender algorithms for an event recommendation system. In *Proceedings of the Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'11) and User-Centric Evaluation of Recommender Systems and Their Interfaces-2 (UCERSTI'11)*, Vol. 811. CEUR-WS.org, 67–73. Retrieved from https://hdl.handle.net/1854/LU-2040145.

[64] Simon Dooms, Toon De Pessemier, and Luc Martens. 2013. MovieTweetings: A movie rating dataset collected from Twitter. In *Proceedings of the Workshop on Crowdsourcing and Human Computation for Recommender Systems (CrowdRec'13)*. http://hdl.handle.net/1854/LU-4284240.

[65] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proceedings of the 3rd Innovations in Theoretical Computer Science Conference (ITCS'12)*. ACM, 214–226. https://doi.org/10.1145/2090236.2090255

[66] Cynthia Dwork and Christina Ilvento. 2018. Group fairness under composition. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT*'18)*. ACM.

[67] Michael D. Ekstrand. 2011. Collaborative filtering recommender systems. *Found. Trends Hum.–Comput. Interact.* 4, 2 (2011), 81–173. https://doi.org/10.1561/1100000009

[68] Michael D. Ekstrand. 2020. LensKit for python: Next-generation software for recommender systems experiments. In *Proceedings of the 29th ACM International Conference on Information and Knowledge Management (CIKM'20)*. ACM, New York, NY, 2999–3006. https://doi.org/10.1145/3340531.3412778

[69] Michael D. Ekstrand, Robin Burke, and Fernando Diaz. 2019. Fairness and discrimination in recommendation and retrieval. In *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys'19)*. ACM, 576–577. https://doi.org/10.1145/3298689.3346964

[70] Michael D. Ekstrand, F. Maxwell Harper, Martijn C. Willemsen, and Joseph A. Konstan. 2014. User perception of differences in recommender algorithms. In *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys'14)*. ACM, 161–168. https://doi.org/10.1145/2645710.2645737

[71] Michael D. Ekstrand and Daniel Kluver. 2021. Exploring author gender in book rating and recommendation. *User Model. User-Adapt. Interact.* 31, 3 (2021), 377–420. https://doi.org/10.1007/s11257-020-09284-2

[72] Michael D. Ekstrand, Michael Ludwig, Joseph A. Konstan, and John T. Riedl. 2011. Rethinking the recommender research ecosystem: Reproducibility, openness, and LensKit. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys'11)*. ACM, 133–140. https://doi.org/10.1145/2043932.2043958

[73] Michael D. Ekstrand and Vaibhav Mahant. 2017. Sturgeon and the cool kids: Problems with random decoys for top-N recommender evaluation. In *Proceedings of the 30th International Florida Artificial Intelligence Research Society Conference (FLAIRS'17)*. AAAI, 639–644.

[74] Michael D. Ekstrand, Mucun Tian, Ion Madrazo Azpiazu, Jennifer D. Ekstrand, Oghenemaro Anuyah, David McNeill, and Maria Soledad Pera. 2018. All the cool kids, how do they fit in? Popularity and demographic biases in recommender evaluation and effectiveness. In *Proceedings of the 1st Conference on Fairness, Accountability and Transparency*, Vol. 81. PMLR, 172–186. Retrieved from https://proceedings.mlr.press/v81/ekstrand18b.html.

[75] Yi Fang and Luo Si. 2011. Matrix co-factorization for recommendation with rich side information and implicit feedback. In *Proceedings of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec'11)*. ACM, 65–69. https://doi.org/10.1145/2039320.2039330

[76] Alexander Felfernig, Ludovico Boratto, Martin Stettinger, and Marko Tkalčič. 2018. Evaluating group recommender systems. In *SpringerBriefs in Electrical and Computer Engineering*. Springer International, Chapter Evaluating Group Recommender Systems, 59–71. https://doi.org/10.1007/978-3-319-75067-5_3

[77] Maurizio Ferrari Dacrema, Simone Boglio, Paolo Cremonesi, and Dietmar Jannach. 2021. A troubling analysis of reproducibility and progress in recommender systems research. *ACM Trans. Info. Syst.* 39, 2, Article 20 (2021). https://doi.org/10.1145/3434185

[78] Maurizio Ferrari Dacrema, Paolo Cremonesi, and Dietmar Jannach. 2019. Are we really making much progress? A worrying analysis of recent neural recommendation approaches. In *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys'19)*. ACM, 101–109. https://doi.org/10.1145/3298689.3347058

[79] Andres Ferraro, Xavier Serra, and Christine Bauer. 2021. Break the loop: Gender imbalance in music recommenders. In *Proceedings of the Conference on Human Information Interaction and Retrieval (CHIIR'21)*. ACM, 249–254. https://doi.org/10.1145/3406522.3446033

[80] Uwe Flick. 2014. *An Introduction to Qualitative Research*. SAGE.

[81] Jill Freyne and Shlomo Berkovsky. 2013. Evaluating recommender systems for supportive technologies. In *Human–Computer Interaction Series*. Springer, London, 195–217. https://doi.org/10.1007/978-1-4471-4778-7_8

[82] Mathias Funk, Anne Rozinat, Evangelos Karapanos, Ana Karla Alves de Medeiros, and Aylin Koca. 2010. In situ evaluation of recommender systems: Framework and instrumentation. *Int. J. Hum.-Comput. Studies* 68, 8 (2010), 525–547. https://doi.org/10.1016/j.ijhcs.2010.01.002

[83] Simon Funk. 2006. Try this at home. Retrieved from http://sifter.org/~simon/journal/2006.

[84] Zeno Gantner, Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2011. MyMediaLite: A free recommender system library. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys'11)*. ACM, 305–308. https://doi.org/10.1145/2043932.2043989

[85] Jean Garcia-Gathright, Christine Hosey, Brian St. Thomas, Ben Carterette, and Fernando Diaz. 2018. Mixed methods for evaluating user satisfaction. In *Proceedings of the 12th ACM Conference on Recommender Systems (RecSys'18)*. ACM, 541–542. https://doi.org/10.1145/3240323.3241622

[86] Florent Garcin, Boi Faltings, Olivier Donatsch, Ayar Alazzawi, Christophe Bruttin, and Amr Huber. 2014. Offline and online evaluation of news recommender systems at swissinfo.ch. In *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys'14)*. ACM, 169–176. https://doi.org/10.1145/2645710.2645745

[87] Mouzhi Ge, Carla Delgado-Battenfeld, and Dietmar Jannach. 2010. Beyond accuracy: Evaluating recommender systems by coverage and serendipity. In *Proceedings of the 4th ACM Conference on Recommender Systems (RecSys'10)*. ACM, 257–260. https://doi.org/10.1145/1864708.1864761

[88] Alexandre Gilotte, Clément Calauzènes, Thomas Nedelec, Alexandre Abraham, and Simon Dollé. 2018. Offline A/B testing for recommender systems. In *Proceedings of the 11th ACM International Conference on Web Search and Data Mining (WSDM'18)*. ACM, 198–206. https://doi.org/10.1145/3159652.3159687

[89] Carlos A. Gomez-Uribe and Neil Hunt. 2016. The Netflix recommender system: Algorithms, business value, and innovation. *ACM Trans. Manage. Info. Syst.* 6, 4, Article 13 (2016). https://doi.org/10.1145/2843948

[90] Steven N. Goodman, Daniele Fanelli, and John P. A. Ioannidis. 2016. What does research reproducibility mean? *Sci. Translat. Med.* 8, 341 (2016), 341ps12. https://doi.org/10.1126/scitranslmed.aaf5027

[91] Anthony G. Greenwald, Anthony R. Pratkanis, Michael R. Leippe, and Michael H. Baumgardner. 1986. Under what conditions does theory obstruct research progress? *Psychol. Rev.* 93, 2 (1986), 216–229. https://doi.org/10.1037/0033-295x.93.2.216

[92] Asela Gunawardana and Guy Shani. 2009. A survey of accuracy evaluation metrics of recommendation tasks. *J. Mach. Learn. Res.* 10 (Dec. 2009), 2935–2962.

[93] Asela Gunawardana, Guy Shani, and Sivan Yogev. 2022. Evaluating recommender systems. In *Recommender Systems Handbook* (3rd ed.), Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, New York, NY, 547–601. https://doi.org/10.1007/978-1-0716-2197-4_15

[94] Guibing Guo, Jie Zhang, Zhu Sun, and Neil Yorke-Smith. 2015. LibRec: A Java library for recommender systems. In *Proceedings of the 23rd Conference on User Modeling, Adaptation and Personalization (UMAP'15)*, Vol. 1388. CEUR-WS.org. Retrieved from http://ceur-ws.org/Vol-1388/demo_paper1.pdf.

[95] Huifeng Guo, Ruiming Tang, Yunming Ye, Zhenguo Li, and Xiuqiang He. 2017. DeepFM: A factorization-machine based neural network for CTR prediction. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI'17)*. AAAI, 1725–1731. https://doi.org/10.5555/3172077.3172127

[96] Elizabeth Gómez, Carlos Shui Zhang, Ludovico Boratto, Maria Salamó, and Guilherme Ramos. 2022. Enabling cross-continent provider fairness in educational recommender systems. *Future Gen. Comput. Syst.* 127 (2022), 435–447. https://doi.org/10.1016/j.future.2021.08.025

[97] F. Maxwell Harper and Joseph A. Konstan. 2016. The MovieLens datasets: History and context. *ACM Trans. Interact. Intell. Syst.* 5, 4, Article 19 (2016). https://doi.org/10.1145/2827872

[98] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, YongDong Zhang, and Meng Wang. 2020. *LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation*. ACM, 639–648. https://doi.org/10.1145/3397271.3401063

[99] Xiangnan He, Lizi Liao, Hanwang Zhang, Liqiang Nie, Xia Hu, and Tat-Seng Chua. 2017. Neural collaborative filtering. In *Proceedings of the 26th International Conference on World Wide Web (WWW'17)*. IW3C2, 173–182. https://doi.org/10.1145/3038912.3052569

[100] Jonathan L. Herlocker, Joseph A. Konstan, and John Riedl. 2000. Explaining collaborative filtering recommendations. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW'00)*. ACM, 241–250. https://doi.org/10.1145/358916.358995

[101] Jonathan L. Herlocker, Joseph A. Konstan, Loren G. Terveen, and John T. Riedl. 2004. Evaluating collaborative filtering recommender systems. *ACM Trans. Info. Syst.* 22, 1 (2004), 5–53. https://doi.org/10.1145/963770.963772

[102] Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. 2015. Session-based recommendations with recurrent neural networks. Retrieved from arXiv:1511.06939.

[103] Neil P. Chue Hong. 2021. Reproducibility badging and definitions: A recommended practice of the national information standards organization. Retrieved from https://www.niso.org/publications/rp-31-2021-badging. https://doi.org/10.3789/niso-rp-31-2021

[104] Rong Hu and Pearl Pu. 2011. Helping users perceive recommendation diversity. In *Proceedings of the Workshop on Novelty and Diversity in Recommender Systems (DiveRS'11), at the 5th ACM International Conference on Recommender Systems (RecSys'11)*. CEUR-WS.org, Article 6, 43–50. Retrieved from http://ceur-ws.org/Vol-816/paper6.pdf.

[105] Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *Proceedings of the 8th IEEE International Conference on Data Mining (ICDM'08)*. IEEE, 263–272. https://doi.org/10.1109/icdm.2008.22

[106] Nicolas Hug. 2017. Surprise, A Python Library for Recommender Systems. Retrieved from http://surpriselib.com.

[107] Neil Hurley and Mi Zhang. 2011. Novelty and diversity in Top-N recommendation—Analysis and evaluation. *ACM Trans. Internet Technol.* 10, 4, Article 14 (2011). https://doi.org/10.1145/1944339.1944341

[108] Ben Hutchinson and Margaret Mitchell. 2019. 50 years of test (Un)fairness: Lessons for machine learning. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT*'19)*. ACM, 49–58. https://doi.org/10.1145/3287560.3287600

[109] Dietmar Jannach and Gediminas Adomavicius. 2016. Recommendations with a purpose. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys'16)*. ACM, 7–10. https://doi.org/10.1145/2959100.2959186

[110] Dietmar Jannach and Gediminas Adomavicius. 2017. Price and profit awareness in recommender systems. In *Proceedings of the 1st International Workshop on Value-Aware and Multistakeholder Recommendation (VAMS'17)*. Retrieved from https://arXiv:1707.08029.

[111] Dietmar Jannach and Christine Bauer. 2020. Escaping the McNamara fallacy: Towards more impactful recommender systems research. *AI Mag.* 41, 4 (2020), 79–95. https://doi.org/10.1609/aimag.v41i4.5312

[112] Dietmar Jannach and Michael Jugovac. 2019. Measuring the business value of recommender systems. *ACM Trans. Manage. Info. Syst.* 10, 4, Article 16 (2019). https://doi.org/10.1145/3370082

[113] Dietmar Jannach, Ahtsham Manzoor, Wanling Cai, and Li Chen. 2021. A Survey on Conversational Recommender Systems. https://doi.org/10.1145/3453154

[114] Dietmar Jannach, Bamshad Mobasher, and Shlomo Berkovsky. 2020. Research directions in session-based and sequential recommendation: A preface to the special issue. *User Model. User-Adapt. Interact.* 30, 4 (2020), 609–616. https://doi.org/10.1007/s11257-020-09274-4

[115] Dietmar Jannach, Oren Sar Shalom, and Joseph A. Konstan. 2019. Towards more impactful recommender systems research. In *Proceedings of the 1st Workshop on the Impact of Recommender Systems (ImpactRS'19)*, Vol. 2462. CEUR-WS.org. Retrieved from http://ceur-ws.org/Vol-2462/short6.pdf.

[116] Dietmar Jannach and Markus Zanker. 2022. Value and impact of recommender systems. In *Recommender Systems Handbook* (3rd ed.), Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, New York, NY, 519–546. https://doi.org/10.1007/978-1-0716-2197-4_14

[117] Dietmar Jannach, Markus Zanker, Alexander Felfernig, and Gerhard Friedrich. 2009. *Recommender Systems: An Introduction*. Cambridge University Press. https://doi.org/10.1017/cbo9780511763113

[118] 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 International Conference on Electronic Commerce and Web Technologies (ECWeb'12)*. Springer, 76–87. https://doi.org/10.1007/978-3-642-32273-0_7

[119] Kalervo Järvelin and Jaana Kekäläinen. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Info. Syst.* 20, 4 (2002), 422–446. https://doi.org/10.1145/582415.582418

[120] Gawesh Jawaheer, Martin Szomszor, and Patty Kostkova. 2010. Comparison of implicit and explicit feedback from an online music recommendation service. In *Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender (HetRec'10)*. ACM, 47–51. https://doi.org/10.1145/1869446.1869453

[121] Mathias Jesse and Dietmar Jannach. 2021. Digital nudging with recommender systems: Survey and future directions. *Comput. Hum. Behav. Rep.* 3, Article 100052 (2021). https://doi.org/10.1016/j.chbr.2020.100052

[122] Thorsten Joachims, Ben London, Yi Su, Adith Swaminathan, and Lequn Wang. 2021. Recommendations as treatments. *AI Mag.* 42, 3 (2021), 19–30. https://doi.org/10.1609/aaai.12014

[123] Soon-Gyo Jung, Joni Salminen, Shammur A. Chowdhury, Dianne Ramirez Robillos, and Bernard J. Jansen. 2020. Things change: Comparing results using historical data and user testing for evaluating a recommendation task. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA'20)*. ACM. https://doi.org/10.1145/3334480.3382945

[124] Iman Kamehkhosh and Dietmar Jannach. 2017. User perception of next-track music recommendations. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization (UMAP'17)*. ACM, 113–121. https://doi.org/10.1145/3079628.3079668

[125] Marius Kaminskas and Derek Bridge. 2017. Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Trans. Interact. Intell. Syst.* 7, 1, Article 2 (2017). https://doi.org/10.1145/2926720

[126] Jayasimha Katukuri, Tolga Könik, Rajyashree Mukherjee, and Santanu Kolay. 2014. Recommending similar items in large-scale online marketplaces. In *Proceedings of the IEEE International Conference on Big Data (Big Data'14)*. IEEE, 868–876. https://doi.org/10.1109/bigdata.2014.7004317

[127] Barbara B. Kawulich. 2005. Participant observation as a data collection method. *Forum Qualitative Sozialforschung/Forum: Qual. Soc. Res.* 6, 2 (2005). https://doi.org/10.17169/fqs-6.2.466

[128] Shah Khusro, Zafar Ali, and Irfan Ullah. 2016. Recommender systems: Issues, challenges, and research opportunities. In *Proceedings of the International Conference on Information Science and Applications (ICISA'16)*. Springer, 1179–1189. https://doi.org/10.1007/978-981-10-0557-2_112

[129] Bart Knijnenburg, Lydia Meesters, Paul Marrow, and Don Bouwhuis. 2010. User-centric evaluation framework for multimedia recommender systems. In *Proceedings of the International Conference on User Centric Media (UCMEDIA'09)*, Vol. 40. Springer, 366–369. https://doi.org/10.1007/978-3-642-12630-7_47

[130] 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 ACM Conference on Recommender Systems (RecSys'11)*. ACM, 141–148. https://doi.org/10.1145/2043932.2043960

[131] Bart P. Knijnenburg and Martijn C. Willemsen. 2009. Understanding the effect of adaptive preference elicitation methods on user satisfaction of a recommender system. In *Proceedings of the 3rd ACM Conference on Recommender Systems (RecSys'09)*. ACM, 381–384. https://doi.org/10.1145/1639714.1639793

[132] Bart P. Knijnenburg and Martijn C. Willemsen. 2015. Evaluating recommender systems with user experiments. In *Recommender Systems Handbook* (2nd ed.). Springer US, New York, NY, 309–352. https://doi.org/10.1007/978-1-4899-7637-6_9

[133] Bart P. Knijnenburg, Martijn C. Willemsen, Zeno Gantner, Hakan Soncu, and Chris Newell. 2012. Explaining the user experience of recommender systems. *User Model. User-Adapt. Interact.* 22, 4–5 (2012), 441–504. https://doi.org/10.1007/s11257-011-9118-4

[134] Bart P. Knijnenburg, Martijn C. Willemsen, and Alfred Kobsa. 2011. A pragmatic procedure to support the user-centric evaluation of recommender systems. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys'11)*. ACM, 321–324. https://doi.org/10.1145/2043932.2043993

[135] Ron Kohavi, Alex Deng, Brian Frasca, Toby Walker, Ya Xu, and Nils Pohlmann. 2013. Online controlled experiments at large scale. In *Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'13)*. ACM, 1168–1176. https://doi.org/10.1145/2487575.2488217

[136] Ron Kohavi, Roger Longbotham, Dan Sommerfield, and Randal M. Henne. 2008. Controlled experiments on the web: Survey and practical guide. *Data Min. Knowl. Discov.* 18, 1 (2008), 140–181. https://doi.org/10.1007/s10618-008-0114-1

[137] Joseph A. Konstan, Robin Burke, and Edward C. Malthouse. 2021. Towards an experimental news user community as infrastructure for recommendation research. In *Proceedings of the 9th International Workshop on News Recommendation and Analytics (INRA'21)*, Vol. 3143. CEUR-WS.org, Article 4, 43–46. Retrieved from http://ceur-ws.org/Vol-3143/paper4.pdf.

[138] Joseph A. Konstan and John Riedl. 2012. Recommender systems: From algorithms to user experience. *User Model. User-Adapt. Interact.* 22, 1–2 (2012), 101–123. https://doi.org/10.1007/s11257-011-9112-x

[139] Yehuda Koren, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42, 8 (2009), 30–37. https://doi.org/10.1109/mc.2009.263

[140] Yehuda Koren, Steffen Rendle, and Robert Bell. 2022. Advances in collaborative filtering. In *Recommender Systems Handbook* (3rd ed.), Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, New York, NY, 91–142. https://doi.org/10.1007/978-1-0716-2197-4_3

[141] Christine Kormos and Robert Gifford. 2014. The validity of self-report measures of proenvironmental behavior: A meta-analytic review. *J. Environ. Psychol.* 40 (2014), 359–371. https://doi.org/10.1016/j.jenvp.2014.09.003

[142] Dominik Kowald, Peter Müllner, Eva Zangerle, Christine Bauer, Markus Schedl, and Elisabeth Lex. 2021. Support the underground: Characteristics of beyond-mainstream music listeners. *EPJ Data Sci.* 10, 1, Article 14 (2021). https://doi.org/10.1140/epjds/s13688-021-00268-9

[143] Matevž Kunaver and Tomaž Požrl. 2017. Diversity in recommender systems—A survey. *Knowl.-Based Syst.* 123 (2017), 154–162. https://doi.org/10.1016/j.knosys.2017.02.009

[144] Henry A. Landsberger. 1958. *Hawthorne Revisited: Management and the Worker, Its Critics, and Developments in Human Relations in Industry*. Cornell University Press.

[145] Dokyun Lee and Kartik Hosanagar. 2014. Impact of recommender systems on sales volume and diversity. In *Proceedings of the International Conference on Information Systems (ICIS'14)*. AIS. Retrieved from https://aisel.aisnet.org/icis2014/proceedings/EBusiness/40/.

[146] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. 2010. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th International Conference on World Wide Web (WWW'10)*. IW3C2, 661–670. https://doi.org/10.1145/1772690.1772758

[147] Jianxun Lian, Xiaohuan Zhou, Fuzheng Zhang, Zhongxia Chen, Xing Xie, and Guangzhong Sun. 2018. xDeepFM: Combining explicit and implicit feature interactions for recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'18)*. ACM, 1754–1763. https://doi.org/10.1145/3219819.3220023

[148] Dawen Liang, Rahul G. Krishnan, Matthew D. Hoffman, and Tony Jebara. 2018. Variational autoencoders for collaborative filtering. In *Proceedings of the World Wide Web Conference (WWW'18)*. IW3C2, 689–698. https://doi.org/10.1145/3178876.3186150

[149] Yu Liang and Martijn C. Willemsen. 2021. The role of preference consistency, defaults and musical expertise in users' exploration behavior in a genre exploration recommender. In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys'21)*. ACM, 230–240. https://doi.org/10.1145/3460231.3474253

[150] Can Liu, Hamed S. Alavi, Enrico Costanza, Shumin Zhai, Wendy Mackay, and Wendy Moncur. 2019. Rigor, relevance and impact: The tensions and trade-offs between research in the lab and in the wild. In *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems (CHI EA'19)*. ACM, Article panel04. https://doi.org/10.1145/3290607.3311744

[151] Nathan N. Liu, Evan W. Xiang, Min Zhao, and Qiang Yang. 2010. Unifying explicit and implicit feedback for collaborative filtering. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management (CIKM'10)*. ACM, 1445–1448. https://doi.org/10.1145/1871437.1871643

[152] Siping Liu, Xiaohan Tu, and Renfa Li. 2017. Unifying explicit and implicit feedback for Top-N recommendation. In *Proceedings of the IEEE 2nd International Conference on Big Data Analysis (ICBDA'17)*. IEEE, 35–39. https://doi.org/10.1109/icbda.2017.8078860

[153] Malte Ludewig, Noemi Mauro, Sara Latifi, and Dietmar Jannach. 2019. Performance comparison of neural and non-neural approaches to session-based recommendation. In *Proceedings of the 13th ACM Conference on Recommender Systems (RecSys'19)*. ACM, 462–466. https://doi.org/10.1145/3298689.3347041

[154] Hao Ma, Haixuan Yang, Michael R. Lyu, and Irwin King. 2008. SoRec: Social recommendation using probabilistic matrix factorization. In *Proceedings of the 17th ACM Conference on Information and Knowledge Mining (CIKM'08)*. ACM, 931–940. https://doi.org/10.1145/1458082.1458205

[155] Christian Matt, Thomas Hess, and Christian Weiß. 2019. A factual and perceptional framework for assessing diversity effects of online recommender systems. *Internet Res.* 29, 6 (2019), 1526–1550. https://doi.org/10.1108/intr-06-2018-0274

[156] James McInerney, Benjamin Lacker, Samantha Hansen, Karl Higley, Hugues Bouchard, Alois Gruson, and Rishabh Mehrotra. 2018. Explore, exploit, and explain: Personalizing explainable recommendations with bandits. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'18)*. ACM, 31–39. https://doi.org/10.1145/3240323.3240354

[157] Sean M. McNee, John Riedl, and Joseph A. Konstan. 2006. Making recommendations better: An analytic model for human-recommender interaction. In *Extended Abstracts on Human Factors in Computing Systems (CHI EA'06)*. ACM, 1103–1108. https://doi.org/10.1145/1125451.1125660

[158] Rishabh Mehrotra, Ashton Anderson, Fernando Diaz, Amit Sharma, Hanna Wallach, and Emine Yilmaz. 2017. Auditing search engines for differential satisfaction across demographics. In *Proceedings of the 26th International Conference on World Wide Web Companion (WWW'17)*. IW3C2, 626–633. https://doi.org/10.1145/3041021.3054197

[159] Tomoko Murakami, Koichiro Mori, and Ryohei Orihara. 2008. Metrics for evaluating the serendipity of recommendation lists. In *New Frontiers in Artificial Intelligence*. Springer, 40–46. https://doi.org/10.1007/978-3-540-78197-4_5

[160] Cataldo Musto, Marco de Gemmis, Pasquale Lops, Fedelucio Narducci, and Giovanni Semeraro. 2022. Semantics and content-based recommendations. In *Recommender Systems Handbook* (3rd ed.), Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, New York, NY, 251–298. https://doi.org/10.1007/978-1-0716-2197-4_7

[161] Athanasios N. Nikolakopoulos, Xia Ning, Christian Desrosiers, and George Karypis. 2022. Trust your neighbors: A comprehensive survey of neighborhood-based methods for recommender systems. In *Recommender Systems Handbook* (3rd ed.), Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, New York, NY, 39–89. https://doi.org/10.1007/978-1-0716-2197-4_2

[162] Xia Ning and George Karypis. 2012. Sparse linear methods with side information for Top-N recommendations. In *Proceedings of the 6th ACM Conference on Recommender Systems (RecSys'12)*. ACM, 155–162. https://doi.org/10.1145/2365952.2365983

[163] Mie Nørgaard and Kasper Hornbæk. 2006. What do usability evaluators do in practice? An explorative study of think-aloud testing. In *Proceedings of the 6th ACM Conference on Designing Interactive Systems (DIS'06)*. ACM, 209–218. https://doi.org/10.1145/1142405.1142439

[164] Yoon-Joo Park and Alexander Tuzhilin. 2008. The long tail of recommender systems and how to leverage it. In *Proceedings of the ACM Conference on Recommender Systems (RecSys'08)*. ACM, 11–18. https://doi.org/10.1145/1454008.1454012

[165] Denis Parra and Xavier Amatriain. 2011. Walk the talk. In *Proceedings of the International Conference on User Modeling, Adaptation, and Personalization (UMAP'11)*. Springer, 255–268. https://doi.org/10.1007/978-3-642-22362-4_22

[166] Denis Parra and Shaghayegh Sahebi. 2013. Recommender systems: Sources of knowledge and evaluation metrics. In *Advanced Techniques in Web Intelligence-2: Web User Browsing Behaviour and Preference Analysis*. Springer, 149–175. https://doi.org/10.1007/978-3-642-33326-2_7

[167] Bibek Paudel, Fabian Christoffel, Chris Newell, and Abraham Bernstein. 2017. Updatable, accurate, diverse, and scalable recommendations for interactive applications. *ACM Trans. Interactive Intelligent Systems* 7, 1 (2017). https://doi.org/10.1145/2955101

[168] Michael J. Pazzani and Daniel Billsus. 2007. Content-based recommendation systems. In *The Adaptive Web*. Springer, Berlin, 325–341. https://doi.org/10.1007/978-3-540-72079-9_10

[169] Elazar J. Pedhazur and Liora Pedhazur Schmelkin. 2013. *Measurement, Design, and Analysis*. Psychology Press. https://doi.org/10.4324/9780203726389

[170] Ken Peffers, Tuure Tuunanen, Marcus A. Rothenberger, and Samir Chatterjee. 2007. A design science research methodology for information systems research. *J. Manage. Info. Syst.* 24, 3 (2007), 45–77. https://doi.org/10.2753/mis0742-1222240302

[171] Guo Chao, Alex Peng, and Fenio Annansingh. 2013. Experiences in applying mixed-methods approach in information systems research. In *Information Systems Research and Exploring Social Artifacts*. IGI Global, Chapter 14, 266–293. https://doi.org/10.4018/978-1-4666-2491-7.ch014

[172] Pearl Pu, Li Chen, and Rong Hu. 2011. A user-centric evaluation framework for recommender systems. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys'11)*. ACM, 157–164. https://doi.org/10.1145/2043932.2043962

[173] Pearl Pu, Li Chen, and Rong Hu. 2012. Evaluating recommender systems from the user's perspective: Survey of the state of the art. *User Model. User-Adapt. Interact.* 22, 4–5 (2012), 317–355. https://doi.org/10.1007/s11257-011-9115-7

[174] Massimo Quadrana, Paolo Cremonesi, and Dietmar Jannach. 2018. Sequence-aware recommender systems. *Comput. Surveys* 51, 4, Article 66 (2018). https://doi.org/10.1145/3190616

[175] Massimo Quadrana, Alexandros Karatzoglou, Balázs Hidasi, and Paolo Cremonesi. 2017. Personalizing session-based recommendations with hierarchical recurrent neural networks. In *Proceedings of the 11th ACM Conference on Recommender Systems (RecSys'17)*. ACM, 130–137. https://doi.org/10.1145/3109859.3109896

[176] Steffen Rendle, Christoph Freudenthaler, Zeno Gantner, and Lars Schmidt-Thieme. 2009. BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence (UAI'09)*. AUAI Press, 452–461.

[177] Steffen Rendle, Li Zhang, and Yehuda Koren. 2019. On the difficulty of evaluating baselines: A study on recommender systems. Retrieved from arXiv:1905.01395.

[178] Paul Resnick and Hal R. Varian. 1997. Recommender systems. *Commun. ACM* 40, 3 (1997), 56–58. https://doi.org/10.1145/245108.245121

[179] Michael Reusens, Wilfried Lemahieu, Bart Baesens, and Luc Sels. 2018. Evaluating recommendation and search in the labor market. *Knowl.-Based Syst.* 152 (2018), 62–69. https://doi.org/10.1016/j.knosys.2018.04.007

[180] Francesco Ricci and Quang Nhat Nguyen. 2007. Acquiring and revising preferences in a critique-based mobile recommender system. *IEEE Intell. Syst.* 22, 3 (2007), 22–29. https://doi.org/10.1109/mis.2007.43

[181] Francesco Ricci, Lior Rokach, and Bracha Shapira. 2022. Recommender systems: Techniques, applications, and challenges. In *Recommender Systems Handbook* (3rd ed.), Francesco Ricci, Lior Rokach, and Bracha Shapira (Eds.). Springer US, New York, NY, 1–35. https://doi.org/10.1007/978-1-0716-2197-4_1

[182] Christian Richthammer and Günther Pernul. 2018. Situation awareness for recommender systems. *Electr. Comm. Res.* 20, 4 (2018), 783–806. https://doi.org/10.1007/s10660-018-9321-z

[183] Marco Rossetti, Fabio Stella, and Markus Zanker. 2016. Contrasting offline and online results when evaluating recommendation algorithms. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys'16)*. ACM, 31–34. https://doi.org/10.1145/2959100.2959176

[184] Daniel Russo, Paolo Ciancarini, Tommaso Falasconi, and Massimo Tomasi. 2018. A meta-model for information systems quality: A mixed study of the financial sector. *ACM Trans. Manage. Info. Syst.* 9, 3, Article 11 (2018). https://doi.org/10.1145/3230713

[185] Alan Said and Alejandro Bellogín. 2014. Comparative recommender system evaluation: Benchmarking recommendation frameworks. In *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys'14)*. ACM, 129–136. https://doi.org/10.1145/2645710.2645746

[186] Alan Said and Alejandro Bellogín. 2014. Rival: A toolkit to foster reproducibility in recommender system evaluation. In *Proceedings of the 8th ACM Conference on Recommender Systems (RecSys'14)*. ACM, 371–372. https://doi.org/10.1145/2645710.2645712

[187] Alan Said, Ernesto W. De Luca, and Sahin Albayrak. 2011. Inferring contextual user profiles-improving recommender performance. In *Proceedings of the 3rd RecSys Workshop on Context-Aware Recommender Systems (CARS'11)*, Vol. 791. Ceur-WS.org. Retrieved from http://ceur-ws.org/Vol-791/paper7.pdf.

[188] Alan Said, Ben Fields, Brijnesh J. Jain, and Sahin Albayrak. 2013. User-centric evaluation of a K-furthest neighbor collaborative filtering recommender algorithm. In *Proceedings of the Conference on Computer Supported Cooperative Work (CSCW'13)*. ACM, 1399–1408. https://doi.org/10.1145/2441776.2441933

[189] Alan Said, Domonkos Tikk, Klara Stumpf, Yue Shi, Martha Larson, and Paolo Cremonesi. 2012. Recommender systems evaluation: A 3D benchmark. In *Proceedings of the Workshop on Recommendation Utility Evaluation: Beyond RMSE (RUE'12)*, Vol. 910. CEUR-WS.org, 21–23. Retrieved from http://ceur-ws.org/Vol-910/.

[190] Steven L. Salzberg. 1997. On comparing classifiers: Pitfalls to avoid and a recommended approach. *Data Min. Knowl. Discov.* 1, 3 (1997), 317–328. https://doi.org/10.1023/a:1009752403260

[191] Piotr Sapiezynski, Wesley Zeng, Ronald E. Robertson, Alan Mislove, and Christo Wilson. 2019. Quantifying the impact of user attentionon fair group representation in ranked lists. In *Companion Proceedings of the World Wide Web Conference (WWW'19)*. IW3C2, 553–562. https://doi.org/10.1145/3308560.3317595

[192] Badrul Sarwar, George Karypis, Joseph Konstan, and John Reidl. 2001. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web (WWW'01)*. IW3C2, 285–295. https://doi.org/10.1145/371920.372071

[193] J. Ben Schafer, Dan Frankowski, Jon Herlocker, and Shilad Sen. 2007. Collaborative filtering recommender systems. In *The Adaptive Web*. Springer, Berlin, 291–324. https://doi.org/10.1007/978-3-540-72079-9_9

[194] Markus Schedl, Stefan Brandl, Oleg Lesota, Emilia Parada-Cabaleiro, David Penz, and Navid Rekabsaz. 2022. LFM-2b: A dataset of enriched music listening events for recommender systems research and fairness analysis. In *Proceedings of the 7th ACM SIGIR Conference on Human Information Interaction and Retrieval (CHIIR'22)*. ACM, New York, NY, 337–341. https://doi.org/10.1145/3498366.3505791

[195] Gunnar Schröder, Maik Thiele, and Wolfgang Lehner. 2011. Setting goals and choosing metrics for recommender system evaluations. In *Proceedings of the Workshop on Human Decision Making in Recommender Systems (Decisions@RecSys'11) and User-Centric Evaluation of Recommender Systems and Their Interfaces-2 (UCERSTI'11)*, Vol. 811. CEUR-WS.org, Article 12, 78–85. Retrieved from http://ceur-ws.org/Vol-811/paper12.pdf.

[196] Andrew D. Selbst, Danah Boyd, Sorelle A. Friedler, Suresh Venkatasubramanian, and Janet Vertesi. 2019. Fairness and abstraction in sociotechnical systems. In *Proceedings of the Conference on Fairness, Accountability, and Transparency (FAT*'19)*. ACM, 59–68. https://doi.org/10.1145/3287560.3287598

[197] Upendra Shardanand and Pattie Maes. 1995. Social information filtering: Algorithms for automating "word of mouth." In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI'95)*. ACM, 210–217. https://doi.org/10.1145/223904.223931

[198] Ashudeep Singh and Thorsten Joachims. 2018. Fairness of exposure in rankings. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'18)*. ACM, 2219–2228. https://doi.org/10.1145/3219819.3220088

[199] Jan Smedslund. 2015. Why psychology cannot be an empirical science. *Integr. Psychol. Behav. Sci.* 50, 2 (2015), 185–195. https://doi.org/10.1007/s12124-015-9339-x

[200] Barry Smyth and Paul McClave. 2001. Similarity vs. diversity. In *Proceedings of the International Conference on Case-Based Reasoning (ICCBR'01)*. Springer, 347–361. https://doi.org/10.1007/3-540-44593-5_25

[201] Eric R. Spangenberg, Ioannis Kareklas, Berna Devezer, and David E. Sprott. 2016. A meta-analytic synthesis of the question–behavior effect. *J. Consum. Psychol.* 26, 3 (2016), 441–458. https://doi.org/10.1016/j.jcps.2015.12.004

[202] Brian St. Thomas, Praveen Chandar, Christine Hosey, and Fernando Diaz. 2021. Mixed method development of evaluation metrics. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (KDD'21)*. ACM, 4070–4071. https://doi.org/10.1145/3447548.3470802

[203] Harald Steck. 2011. Item popularity and recommendation accuracy. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys'11)*. ACM, 125–132. https://doi.org/10.1145/2043932.2043957

[204] Harald Steck. 2013. Evaluation of recommendations: Rating-prediction and ranking. In *Proceedings of the 7th ACM Conference on Recommender Systems (RecSys'13)*. ACM, 213–220. https://doi.org/10.1145/2507157.2507160

[205] Elliot Stern. 2005. *Eval. Res. Methods*. SAGE. https://doi.org/10.4135/9781446261606

[206] Xiaoyuan Su and Taghi M. Khoshgoftaar. 2009. A survey of collaborative filtering techniques. *Advances in Artificial Intelligence* 2009, Article 4 (2009). https://doi.org/10.1155/2009/421425

[207] Adith Swaminathan and Thorsten Joachims. 2015. Batch learning from logged bandit feedback through counterfactual risk minimization. *J. Mach. Learn. Res.* 16, 52 (2015), 1731–1755. Retrieved from http://jmlr.org/papers/v16/swaminathan15a.html.

[208] John A. Swets. 1963. Information retrieval systems: Statistical decision theory may provide a measure of effectiveness better than measures proposed to date. *Science* 141, 3577 (1963), 245–250. https://doi.org/10.1126/science.141.3577.245

[209] Yan-Martin Tamm, Rinchin Damdinov, and Alexey Vasilev. 2021. Quality metrics in recommender systems: Do we calculate metrics consistently? In *Proceedings of the 15th ACM Conference on Recommender Systems (RecSys'21)*. ACM, 708–713. https://doi.org/10.1145/3460231.3478848

[210] Jiliang Tang, Charu Aggarwal, and Huan Liu. 2016. Recommendations in signed social networks. In *Proceedings of the 25th International Conference on World Wide Web (WWW'16)*. IW3C2, 31–40. https://doi.org/10.1145/2872427.2882971

[211] Nava Tintarev and Judith Masthoff. 2010. Designing and evaluating explanations for recommender systems. In *Recommender Systems Handbook* (2nd ed.). Springer US, 479–510. https://doi.org/10.1007/978-0-387-85820-3_15

[212] Aäron Van Den Oord, Sander Dieleman, and Benjamin Schrauwen. 2013. Deep content-based music recommendation. In *Proceedings of the 26th International Conference on Neural Information Processing Systems (NIPS'13)*. Curran Associates, 2643–2651. https://doi.org/10.5555/2999792.2999907

[213] C. J. van Rijsbergen. 1979. *Information Retrieval*. Butterworths. Retrieved from http://www.dcs.gla.ac.uk/Keith/Preface.html.

[214] Saúl Vargas and Pablo Castells. 2011. Rank and relevance in novelty and diversity metrics for recommender systems. In *Proceedings of the 5th ACM Conference on Recommender Systems (RecSys'11)*. ACM, 109–116. https://doi.org/10.1145/2043932.2043955

[215] Ellen M. Voorhees. 2000. The TREC-8 Question Answering Track Report, 77–82. Retrieved from https://tsapps.nist.gov/publication/get_pdf.cfm?pub_id=151495.

[216] Ellen M. Voorhees. 2002. The philosophy of information retrieval evaluation. In *Proceedings of the Workshop of the Cross-Language Evaluation Forum for European Languages*. Springer, 355–370. https://doi.org/10.1007/3-540-45691-0_34

[217] Eric-Jan Wagenmakers, Ruud Wetzels, Denny Borsboom, Han L. J. van der Maas, and Rogier A. Kievit. 2012. An agenda for purely confirmatory research. *Perspect. Psychol. Sci.* 7, 6 (2012), 632–638. https://doi.org/10.1177/1745691612463078 PMID: 26168122.

[218] Qingyun Wu, Hongning Wang, Liangjie Hong, and Yue Shi. 2017. Returning is believing: Optimizing long-term user engagement in recommender systems. In *Proceedings of the ACM on Conference on Information and Knowledge Management (CIKM'17)*. ACM, 1927–1936. https://doi.org/10.1145/3132847.3133025

[219] Bo Xiao and Izak Benbasat. 2007. E-commerce product recommendation agents: Use, characteristics, and impact. *Manage. Info. Syst. Quart.* 31, 1 (2007), 137. https://doi.org/10.2307/25148784

[220] Sirui Yao and Bert Huang. 2017. Beyond parity: Fairness objectives for collaborative filtering. In *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17)*. Curran Associates, 2925–2934. https://doi.org/10.5555/3294996.3295052

[221] Robert K. Yin. 1989. *Case Study Research: Design and Methods* (5th revised ed.). SAGE.

[222] Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L. Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD'18)*. ACM, 974–983. https://doi.org/10.1145/3219819.3219890

[223] Hsiang-Fu Yu, Cho-Jui Hsieh, Si Si, and Inderjit Dhillon. 2012. Scalable coordinate descent approaches to parallel matrix factorization for recommender systems. In *Proceedings of the IEEE 12th International Conference on Data Mining (ICDM'12)*. IEEE, 765–774. https://doi.org/10.1109/icdm.2012.168

[224] Eva Zangerle, Wolfgang Gassler, Martin Pichl, Stefan Steinhauser, and Günther Specht. 2016. An empirical evaluation of property recommender systems for wikidata and collaborative knowledge bases. In *Proceedings of the 12th International Symposium on Open Collaboration (OpenSym'16)*. ACM, Article 18. https://doi.org/10.1145/2957792.2957804

[225] Markus Zanker, Laurens Rook, and Dietmar Jannach. 2019. Measuring the impact of online personalisation: Past, present and future. *Int. J. Hum.-Comput. Studies* 131 (2019), 160–168. https://doi.org/10.1016/j.ijhcs.2019.06.006

[226] Shuai Zhang, Lina Yao, Aixin Sun, and Yi Tay. 2019. Deep learning based recommender system: A survey and new perspectives. *Comput. Surveys* 52, 1, Article 5 (2019). https://doi.org/10.1145/3285029

[227] Qian Zhao, Gediminas Adomavicius, F. Maxwell Harper, Martijn Willemsen, and Joseph A. Konstan. 2017. Toward better interactions in recommender systems: Cycling and serpentining approaches for Top-N item lists. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing (CSCW'17)*. ACM, New York, NY, 1444–1453. https://doi.org/10.1145/2998181.2998211

[228] Guanjie Zheng, Fuzheng Zhang, Zihan Zheng, Yang Xiang, Nicholas Jing Yuan, Xing Xie, and Zhenhui Li. 2018. DRN: A deep reinforcement learning framework for news recommendation. In *Proceedings of the World Wide Web Conference (WWW'18)*. IW3C2, 167–176. https://doi.org/10.1145/3178876.3185994

[229] Lei Zheng, Vahid Noroozi, and Philip S. Yu. 2017. Joint deep modeling of users and items using reviews for recommendation. In *Proceedings of the 10th ACM International Conference on Web Search and Data Mining (WSDM'17)*. ACM, 425–434. https://doi.org/10.1145/3018661.3018665

[230] Tao Zhou, Zoltán Kuscsik, Jian-Guo Liu, Matúš Medo, Joseph Rushton Wakening, and Yi-Cheng Zhang. 2010. Solving the apparent diversity-accuracy dilemma of recommender systems. *Proc. Natl. Acad. Sci. U.S.A.* 107, 10 (2010), 4511–4515. https://doi.org/10.1073/pnas.1000488107

[231] Cai-Nicolas Ziegler, Sean M. McNee, Joseph A. Konstan, and Georg Lausen. 2005. Improving recommendation lists through topic diversification. In *Proceedings of the 14th International Conference on World Wide Web (WWW'05)*. IW3C2, 22–32. https://doi.org/10.1145/1060745.1060754

Received 31 March 2021; revised 22 December 2021; accepted 18 July 2022