Bibliographic Details | BibTeX, EndNote… | Authors' Details | Related Work

Towards Reproducibility in Recommender-Systems Research

Joeran Beel[1,5], Corinna Breitinger[1,2], Stefan Langer[1,3],
Andreas Lommatzsch[4], Bela Gipp[1,5]

[1] Docear, Konstanz, Germany
[2] Linnaeus University, Sweden
[3] Otto-von-Guericke University, Magdeburg, Germany
[4] Technische Universität Berlin, DAI-Lab, Berlin, Germany
[5] University of Konstanz, Germany

{beel | breitinger | langer | gipp}@docear.org, andreas@dai-lab.de

Abstract. Numerous recommendation approaches are in use today. However, comparing their effectiveness is a challenging task because evaluation results are rarely reproducible. In this article, we examine the challenge of reproducibility in recommender-system research. We conduct experiments using Plista's news recommender system, and Docear's research-paper recommender system. The experiments show that there are large discrepancies in the effectiveness of identical recommendation approaches in only slightly different scenarios, as well as large discrepancies for slightly different approaches in identical scenarios. For example, in one news-recommendation scenario, the performance of a content-based filtering approach was twice as high as the second-best approach, while in another scenario the same content-based filtering approach was the worst performing approach. We found several determinants that may contribute to the large discrepancies observed in recommendation effectiveness. Determinants we examined include user characteristics (gender and age), datasets, weighting schemes, the time at which recommendations were shown, and user-model size. Some of the determinants have interdependencies. For instance, the optimal size of an algorithms' user model depended on users' age. Since minor variations in approaches and scenarios can lead to significant changes in a recommendation approach's performance, ensuring reproducibility of experimental results is difficult. We discuss these findings and conclude that to ensure reproducibility, the recommender-system community needs to (1) survey other research fields and learn from them, (2) find a common understanding of reproducibility, (3) identify and understand the determinants that affect reproducibility, (4) conduct more comprehensive experiments (5) modernize publication practices, (6) foster the development and use of recommendation frameworks, and (7) establish best-practice guidelines for recommender-systems research.

Keywords: recommender systems, evaluation, experimentation, reproducibility

1 Introduction

Reproducibility of experimental results has been called the “fundamental assumption” in science [Casadevall and Fang 2010], and the “cornerstone” for drawing generalizable conclusions [Rehman 2013]. If an experiment cannot be reproduced then it is a single occurrence and is of “no significance to science” [Popper 1959].

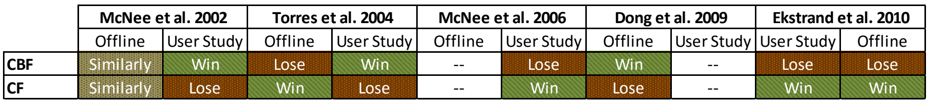
In the recommender-systems community, we found that reproducibility is rarely given, particularly in the *research-paper* recommender-system community [Beel, Gipp, et al. 2015; Beel, Langer, Genzmehr, Gipp, et al. 2013a]. In a review of 89 evaluations of research-paper recommender-systems, we found several cases in which very slight variations in the experimental set-up led to surprisingly different outcomes.

In one case, the developers of the recommender system *bX* reported that the effectiveness of their recommender system varied by a factor of three at different institutions although the same recommendation approach was used [Thomas et al. 2011]. In another case, Lu et al. [2011] reported that the *translation* model had an accuracy twice as high as that of the *language* model, but in another evaluation, accuracy was only 18% higher [He et al. 2012]. Huang et al. [2012] report that the *Context-aware Relevance Model* (CRM) and *cite-LDA* performed similarly, but in another evaluation by the same authors, CRM underperformed cite-LDA.

Lu et al. [2011] found that sometimes terms from an article’s abstract performed better than terms from the article’s body, but in other cases they observed the opposite. Zarrinkalam and Kahani [2013] found that terms from the title *and* abstract were most effective for content-based filtering approaches in some cases, while in other cases terms from the title, abstract, *and* citation context were most effective. Bethard and Jurafsky [2010] reported that using citation counts in the recommendation process *strongly* increased the effectiveness of their recommendation approach, while He et al. [2010] reported that citation counts *slightly* increased the effectiveness of their approach.

Several evaluations performed by the TechLens team and Dong et al. [2009] provide another example where currently unknown determinants skew the evaluation outcomes in unforeseen ways (Table 1). The researchers proposed and evaluated several content-based filtering (CBF) and collaborative filtering (CF) approaches for research-paper recommendations. In one experiment, CF and CBF performed similarly well [McNee et al. 2002]. In other experiments, CBF outperformed CF [Dong et al. 2009; McNee et al. 2002; Torres et al. 2004], and in some more experiments CF outperformed CBF [Ekstrand et al. 2010; McNee et al. 2006; Torres et al. 2004].

Table 1: Results of different CBF and CF evaluations [Beel 2015; Beel, Gipp, et al. 2015]



	McNee et al. 2002		Torres et al. 2004		McNee et al. 2006		Dong et al. 2009		Ekstrand et al. 2010	
	Offline	User Study	Offline	User Study	Offline	User Study	Offline	User Study	User Study	Offline
CBF	Similarly	Win	Lose	Win	--	Lose	Win	--	Lose	Lose
CF	Similarly	Lose	Win	Lose	--	Win	Lose	--	Win	Win

Recommendation effectiveness in our own recommender system Docear also varies significantly. In one experiment, removing stop words increased recommendation effectiveness by 50% (CTR = 4.16% vs. CTR = 6.54%) [Beel, Langer, Genzmehr and Nürnberger 2013a]. In another experiment, effectiveness was almost the same (CTR = 5.94% vs. CTR = 6.31%) [Beel and Langer 2015]. Similarly, in one experiment, the ‘stereotype’ recommendation approach was around 60% more effective than in another experiment (CTR = 3.08% vs. CTR = 4.99%) [Beel, Langer, et al. 2015; Beel, Langer, Genzmehr, et al. 2014].

Why these contradictions occurred, even among very similar literature recommendation scenarios, remains widely unknown. The authors of the studies only mention a few potential reasons for the variations, such as different datasets or variations in the recommendation approaches. Yet, these reasons can only explain *some* of the discrepancies in the results. The majority of determinants are not being discussed by the academic community, as we will later explore in detail.

The current difficulties in reproducing research results lead to a problematic situation for recommender-system developers and researchers. Developers who need an effective recommendation approach, and researchers who need a baseline against which to compare a novel approach, find little guidance in existing publications.

For instance, the current publications would not help us decide if, for instance, CF or CBF is more suitable in implementing an effective research-paper recommender system for a given scenario. Neither could it tell us if the *Context-aware Relevance Model* or *cite-LDA* is more promising; or how strongly the utilization of citation counts will increase effectiveness of research-paper recommender systems.

The recommender-system community, and in particular the research-paper recommender-system community, widely seems to accept that research results are difficult to reproduce, and that publications provide little guidance for recommender-system developers and researchers. When we submitted a paper to the ACM RecSys conference pointing out the inconsistent results of the above-mentioned evaluations, which are not contributing to identifying effective recommendation approaches, one reviewer commented:

> *"I think that it is widely agreed in the community that this [difficulty to reproduce results] is just the way things are – if you want a recsys for a specific application, there is no better way than just test and optimize a number of alternatives."*

The standpoint of the reviewer appears to be shared among many researchers. Looking at publications about recommender-systems evaluation, many authors do not cover the issue of reproducibility. For instance, Al-Maskari et al. [2007] analyzed how well classic IR evaluation metrics correlated with user satisfaction in recommender systems. Gunawardana and Shani [2009] published a survey about accuracy metrics to measure the effectiveness of recommender systems. Herlocker et al. [2004] wrote an article on how to evaluate collaborative-filtering approaches. Various authors showed that offline and online evaluations often provide contradictory results [Cremonesi et al. 2012; McNee et al. 2002], and several more papers about various aspects of recommender-system evaluation have been published [Amatriain et al. 2009; Beel and Langer 2015; Bollen and Rocha 2000; Bogers and Bosch 2007; Cremonesi et al. 2011; Domingues Garcia et al. 2012; Ge et al. 2010; Hayes et al. 2002; Hofmann et al. 2014; Jannach et al. 2012; Knijnenburg et al. 2012; Knijnenburg et al. 2011; Konstan and Riedl 2012; Manouselis and Verbert 2013; Pu et al. 2011; Pu et al. 2012; Said 2013; Shani and Gunawardana 2011]. However, while many of the findings in these papers are important with respect to reproducibility, the authors did not mention or discuss their findings in the context of reproducibility. Similarly, most, if not all, recommender-systems text books or literature surveys address the topic of evaluations, but lack a section about reproducibility [Bobadilla et al. 2013; Felfernig et al. 2013; Jannach 2014; Konstan and Ekstrand 2015; Ricci et al. 2015; Ricci et al. 2011; Sharma and Gera 2013].

Although many researchers do not address the topic of reproducibility, or accept that the difficulty to reproduce experimental results is "just the way things are", some believe that the community should place more emphasis on reproducibility. For instance, Ekstrand, Ludwig, Konstan, et al. [2011] criticize that "it is currently difficult to reproduce and extend recommender systems research results," and that evaluations are "not handled consistently". Konstan and Adomavicius [2013] add that many research papers "contribute little to collective knowledge", primarily due to the difficulty of reproducing the results. They conclude:

> *"[T]he Recommender Systems research community is facing a crisis where a significant number of papers present results that contribute little to collective knowledge [...] often because the research lacks the [...] evaluation to be properly judged and, hence, to provide meaningful contributions."*

We agree with Ekstrand, Konstan, et al. and believe that ensuring reproducibility in recommender-system research is crucial, and should receive more attention in the community. Therefore, we explore the reasons for the volatility of results in recommender-system research.

We review the literature on reproducibility and recommender-systems evaluation, conduct experiments with the news recommender system *Plista* and the research-paper recommender system *Docear*, discuss current challenges, and suggest several actions to increase reproducibility in recommender-systems research. We do not claim to provide definitive guidelines for ensuring reproducibility, but we hope to provide initial ideas and empirical evidence to stimulate a discussion that will contribute to making research in the recommender-systems field more reproducible.

Our focus lies on exploring how differences in recommendation (1) *scenarios* and (2) *approaches* affect recommendation effectiveness. Another important question is "How do differences in recommendation *evaluations* affect recommendation effectiveness?" Although there is no agreement on the answer, there has at least been a lively discussion about it [Beel and Langer 2015; Ge et al. 2010; Knijnenburg et al. 2012; Knijnenburg et al. 2011; Konstan and Riedl 2012; Pu et al. 2011; Pu et al. 2012; Said 2013]. Therefore, we concentrate on examining differences in scenarios and approaches, which have not received much attention in the recommender-system community.

2 Definitions

To understand the discussion in this article, a few definitions are required[1]. Figure 1 summarizes the relationships among the definitions described in this section.

We use "idea" to refer to a hypothesis about how recommendations could be generated. To differentiate how specific the idea is, we distinguish between recommendation classes, approaches, algorithms, and implementations.

We define a "recommendation class" as the least specific idea, namely a broad concept that vaguely describes how recommendations might be given. Recommendation classes are, for instance, collaborative filtering (CF) and content-based filtering (CBF). These classes fundamentally differ in their underlying ideas. For instance, the underlying idea of CBF is that users are interested in items that are similar to items the users previously liked, where similarity is measured in terms of content similarity. In contrast, the idea of CF is that users like items that the users' peers also liked. These ideas are rather vague and leave room for speculation about how the idea should be realized.

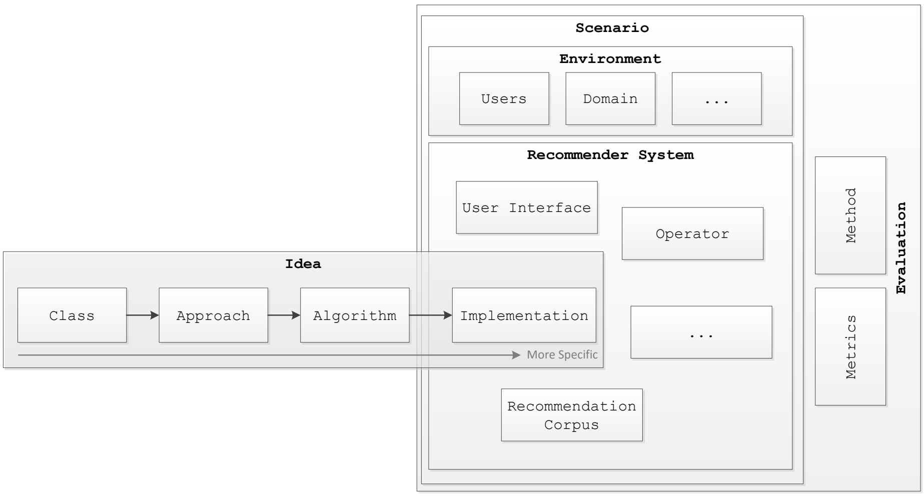


Figure 1: Illustration of recommendation idea, recommender system, environment, scenario, and evaluation [Beel 2015]

A "recommendation approach" is a model of how to bring a recommendation class into practice. For instance, the concept of CF can be realized with user-based CF [Resnick et al. 1994], content-boosted CF [Melville et al. 2002], and various other approaches [Shi et al. 2014]. These approaches are quite different but are each consistent with the central idea of CF. Nevertheless, approaches are rather vague, leaving some room for speculation on how recommendations are precisely generated.

A "recommendation algorithm" outlines the idea behind a recommendation approach in more detail. For instance, an algorithm of a CBF approach specifies whether terms are extracted from the title or from the body of a document, and how these terms are processed (e.g. stop word removal or stemming) and weighted (e.g. TF-IDF). Algorithms are not necessarily complete. For instance, pseudo code might contain only the most important components of an approach and ignore details, such as weighting schemes. This means that for a particular recommendation approach there might be several (slightly) different algorithms.

[1] Some of the definitions have been previously introduced by Beel [2015].

An "implementation" is the (compiled) source code of an algorithm, which can be applied in a recommender system. An implementation leaves no room for speculation on how recommendations are generated. It is hence the most specific idea describing how recommendations are generated.

A "recommender system" is a fully functional software system that applies at least one implementation for generating recommendations. Recommender systems feature several other components, such as a user interface and a corpus of recommendation candidates. Some recommender systems also apply two or more recommendation approaches. For instance, CiteULike lets their users choose which approach to use [Bogers and Bosch 2008; CiteULike 2011], and Docear randomly selects one of three approaches each time users request recommendations [Beel, Langer, Gipp, et al. 2014].

The "recommendation scenario" describes the entire setting of a recommender system, including the recommender system and the recommendation environment, i.e. the domain, user characteristics, and the provider that maintains the recommender system. Usually, we say that the implementation of a recommendation approach is part of the scenario, since the implementation is part of the recommender system and the recommender system is part of the scenario. However, in this article, we distinguish between the implementation *and* the scenario, since we focus on the question of how the characteristics of a scenario affect the effectiveness of an implementation.

By "effectiveness" we mean the degree to which a recommender system achieves its objective. The recommendation objective is typically to provide “good” [Gunawardana and Shani 2009] and “useful” [Herlocker et al. 2004] recommendations that make users “happy” [Ge et al. 2010] by satisfying their needs [McNee et al. 2006]. The needs of users vary and, consequently, different items might make users happy. For instance, while all users expect recommendations that are specific to their fields of research, some users are interested in *novel* research-paper recommendations, while others are interested in *authoritative* research-paper recommendations [Torres et al. 2004]. Objectives beyond user satisfaction are also present, particularly for the provider of a recommender system [Gunawardana and Shani 2009]. For instance, the provider's goal might be to achieve high revenues or to keep users on the provider's website for as long as possible.

"Evaluation" describes any kind of assessment that measures the effectiveness of recommendation algorithms. We use the terms “performance” and “effectiveness” interchangeably. There are different classifications of recommender-system evaluation methods. Some researchers distinguish between *offline* and *online* evaluations [Zheng et al. 2010], between *data-centric* and *user-centric* evaluations [Said 2013], between *live user experiments* and *offline analyses* [Herlocker et al. 2004], and between *offline evaluations*, *online evaluations*, and *user studies* [Ricci et al. 2011].

We adopt the classification by Ricci et al., i.e. we distinguish between offline evaluations, online evaluations, and user studies. User studies and online evaluations assess the effectiveness of the entire recommendation scenario. This means, for instance, that user-interfaces may affect users’ satisfaction with recommendations. Offline evaluations focus on the effectiveness of the recommendation approaches but are also affected by factors including datasets and the recommendation corpus. Researchers commonly assume that if their specific implementation of a recommendation approach was successfully evaluated, the corresponding recommendation approach is effective in general. Therefore, we speak of evaluating recommendation approaches, although it is the individual implementation that is being evaluated.

"Reproducibility" describes the case in which similar ideas lead to similar experimental results given similar evaluations and scenarios, where "similar results" are results that allow the same conclusions to be drawn [Casadevall and Fang 2010].

Conversely, if changes in the ideas, scenarios, or evaluations cause different outcomes, i.e. the same conclusions cannot be drawn, we speak of results not being reproducible. When significant changes are made to the ideas, scenarios, or evaluations then we expect results to differ. However, if only minor changes are made, yet the results are surprisingly dissimilar, we know there must be unknown factors that alone or in combination influence the observed dissimilarity. We can refer to these factors as unknown contextual determinants.

"Replicability" describes an exact copy of an experiment that uses the same tools, follows the same steps, and produces the same results [Drummond 2009]. Therefore, replicability is typically used to describe if an experiment can be repeated under *identical* circumstances, i.e. if identical algorithms achieve identical results in identical scenarios. In contrast, reproducibility is important for generalizing from an evaluation's result and drawing valid conclusions, for example, that algorithm A is – under a set of conditions – faster than algorithm B, even if the algorithms and scenarios differ to some extent.

3 Related Work

3.1 Impact of Differences in the Ideas

Ekstrand, Ludwig, Konstan, et al. [2011] believe that variations in algorithms and implementations are a major reason for results being difficult to reproduce. Hence, they criticize researchers' superficial descriptions of algorithms that force other researchers to make guesses about the details of an algorithm. The information sparsity leads to variations in the implementation, which might lead to results different to those from the original experiment. To address this problem, Ekstrand et al. suggest that researchers should publish reusable code for existing recommendation frameworks. As a result, researchers would not have to guess the specifics of an algorithm implementation but could use the same implementation. To bring their suggestion to fruition, Ekstrand et al. [2011] introduced the recommendation toolkit *LensKit* that helps run and evaluate recommendation algorithms.

Different implementations might explain *some* of the discrepancies observed in the evaluations presented in Section 1. In some evaluations, the implementations, and even approaches, differed slightly. For instance, the CF approaches of the TechLens team were not always the same. When different authors evaluated the same approaches, we suspect that they used different implementations.

3.2 Impact of Differences in the Scenarios

Different users tend to have different preferences and needs, and these needs are each uniquely satisfied by a particular product [Burns and Bush 2013]. Hence, results from user studies should not be generalized to scenarios where user populations differ strongly from the original population – this is common knowledge in many research disciplines [Burns and Bush 2013]. We would assume that the same holds true for recommender-system research.

To some extent, this issue has been addressed by the TechLens team, although not in the context of reproducibility [Torres et al. 2004]. The TechLens team found that varying approaches performed differently depending on the task pursued by the user. For instance, when users were beginning their research, approaches that recommended more authoritative papers and survey articles performed better. When users were already familiar with a research field, approaches that favored novel articles performed better.

We would assume that the task a user is pursuing correlates with the user's background, e.g. their experience level. If this assumption is true, differences in user populations might explain the differences in some of the results of the examples in Section 1. For instance, in one study by TechLens that compared CF and CBF, more than 80% of the participants were professors [McNee et al. 2006]. In another study, around one-fifth were professors [Torres et al. 2004]. In a third study, there were no professors [Ekstrand et al. 2010]. Given this user information, and assuming that professors have different experience levels than students, variations in the effectiveness of CBF and CF based on the user studies seem plausible.

Another aspect of recommendation scenarios are the datasets. It is well known that recommendation approaches perform differently on different datasets [Breese et al. 1998; Gunawardana and Shani 2009]. Such differences could explain the contradictions of the CRM and cite-LDA evaluations [Huang et al. 2012]. The evaluation in which cite-LDA and CRM performed alike used data from CiteSeer. The evaluation in which cite-LDA performed better than CRM used data from CiteULike.

How intensely users interact with the recommender system might also be responsible for different outcomes. For instance, collaborative filtering needs a certain number of user ratings before it becomes effective (cold start problem) [Schein et al. 2002]. When CF is evaluated using a well-established recommender system, results can differ from an evaluation in a new system with few users.

3.3 Impact of Differences in the Evaluations

Konstan and Adomavicius [2013] voice the concern that a significant number of research papers "contribute little to collective knowledge", primarily because the evaluations quality cannot be properly judged. When it comes to the quality of evaluations, Konstan and Adomavicius identify several problems, such as the evaluations and research results not always being thoroughly documented, the use of private datasets, technical mistakes when choosing metrics, and not comparing approaches against appropriate baselines. Konstan and Adomavicius conclude that such problems can cause recommender-system research to lack the required technical rigor. They suggest that the recommender-system community needs best-practice guidelines on how to perform recommender-system evaluations. Such guidelines would provide novice researchers with instructions on how to design evaluations correctly, i.e. make results reproducible and empower reviewers to judge the soundness of evaluations more objectively.

Currently, the recommender-system community is divided on the question if current evaluations are stringent enough, and whether best-practice guidelines are needed. Konstan and Adomavicius [2013] sent a survey to the program committee of the ACM Recommender Systems Conference 2013. They asked 43 committee members how often problems, such as private datasets and mistakes in metrics occurred in recommender-system articles based on the committee members' experience. 19 participants (44%) responded that such problems occurred too frequently. 24 participants (56%) believed such problems occur occasionally or very rarely. 12 out of 45 participants (27%) thought that best-practice guidelines might be helpful for novice researchers, although they assumed that "most researchers already know these things". 71% of participants believed that such guidelines could be useful for many researchers.

We can only speculate about why the study participants had different perceptions on the current state of recommender-systems evaluation. Perhaps evaluation practices differ among the recommender-system domains (movies, news, music, etc.). Reviewers from a domain with more stringent evaluation standards might have responded differently to the survey than reviewers from domains with less standards. Another explanation could be differences in the perceptions of the importance of rigorous evaluations and reproducibility. As mentioned in Section 1, some researchers accept the difficulty to reproduce evaluations as "*just the way things are*". These researchers probably judge the current state of evaluation practices as sufficient.

In the domain of research-paper recommender systems, we identified a clear need for best-practice guidelines. We reviewed 89 research-paper recommender system approaches and found some shared shortcomings in the evaluations [Beel, Gipp, et al. 2015; Beel, Langer, Genzmehr, Gipp, et al. 2013a]. 21% of the approaches introduced in the research papers were not evaluated. Of the evaluated approaches, 69% were evaluated using offline evaluations, a method with serious shortcomings [Beel, Langer, Genzmehr, Gipp, et al. 2013b; Cremonesi et al. 2011; Cremonesi et al. 2012; W. Hersh et al. 2000; W. R. Hersh et al. 2000; Jannach et al. 2013; Knijnenburg et al. 2012; McNee et al. 2002;

Said 2013; Turpin and Hersh 2001]. Furthermore, many approaches were evaluated against trivial (if any) baselines. Datasets were heavily pruned, sometimes in questionable ways, and user studies often had insufficient participants for statistically significant results.

In addition to these technical problems, two more challenges in evaluations can affect reproducibility.

First, different *evaluation methods* assess different objectives. Offline evaluations measure accuracy, while user studies and online evaluations measure (implicit) user satisfaction [Beel, Langer, Genzmehr, Gipp, et al. 2013b; Beel and Langer 2015]. Hence, comparing the results of different evaluation methods is like comparing apples and oranges.

Second, the chosen evaluation methods and metrics might be *unsuitable*. Some researchers criticize offline evaluations as being unsuitable for evaluating recommender systems because they ignore human factors and are often based on biased ground-truths [Beel, Langer, Genzmehr, Gipp, et al. 2013b; Cremonesi et al. 2011; Cremonesi et al. 2012; W. Hersh et al. 2000; W. R. Hersh et al. 2000; Jannach et al. 2013; Knijnenburg et al. 2012; McNee et al. 2002; Said 2013; Turpin and Hersh 2001]. If the critics are right, contradictions between offline experiments and user studies are to be expected because offline evaluations would provide meaningless results. Similarly, if offline evaluations were generally unreliable, contradictions between different offline evaluations are to be expected.

Problems in the evaluation design, using different evaluation methods, or unsuitable metrics might explain some of the discrepancies observed in Section 1. For instance, in the TechLens evaluations, one user study had only 19 participants [Ekstrand et al. 2010]. Other studies had more than 100 participants [McNee et al. 2006; Torres et al. 2004], but evaluated different approaches, which meant that each approach was evaluated using 20 - 30 participants, possibly leading to statistically insignificant results. In other evaluations, researchers pruned their datasets so strongly that less than 1% of the originally included documents remained [Pennock et al. 2000]. In addition, terms from the title and abstract were most effective when evaluated with *co-citation probability*. Terms from the abstract, title, and citation context were most effective when evaluated with *recall*. Since co-citation measures something different than recall, differences in the results and conclusions seem plausible.

3.4 Impact of Variable Interactions

Another reason for results being difficult to reproduce can be the unpredictable interactions among variables. At first glance, we would assume that if one variable proves effective with one recommendation approach, it should also be effective in other recommendation approaches. For instance, if using terms from the abstract is effective for one recommendation approach, we assume that it should be effective for other sufficiently similar approaches. However, the studies examined indicate that this assumption is false. For instance, the *translation model* performed best with terms from the abstract [Lu et al. 2011] while the *language model* performed best with terms from the text body [Lu et al. 2011]. Apparently, one or more unidentified contextual determinants are interacting in a way that leads to different levels of effectiveness for the selected document fields (e.g. abstract vs. text body).

4 Experimental Setup

From the available research, one can conclude that when differences in ideas, scenarios, and evaluations become too large, this will likely lead to results being difficult to reproduce. However, it is unknown how large differences can be before being "too large" and which determinants are responsible for different results. For instance, we might know that different datasets affect reproducibility, but not which features of the datasets are responsible (e.g. the number of items in the datasets, the features of the items, or the characteristics of the users from which the items are from). Additionally, in the majority of research, more than one variable was changed at a time. For instance, in some of the TechLens evaluations, both user populations *and* recommendation approaches were changed. Hence, it is unknown, which of these two factors caused the change in outcome and to what extent.

To explore the issue of reproducibility we conducted the following experiments:

1. We varied the recommendation scenarios (and kept using the same implementations and evaluation methods).
2. We varied the implementations (while using the same scenario and evaluation methods).

We conducted the experiments with the news article recommender system *Plista*[2] and the research-paper recommender system of *Docear* [Beel, Langer, Genzmehr and Nürnberger 2013a; Beel, Langer, Gipp, et al. 2014].

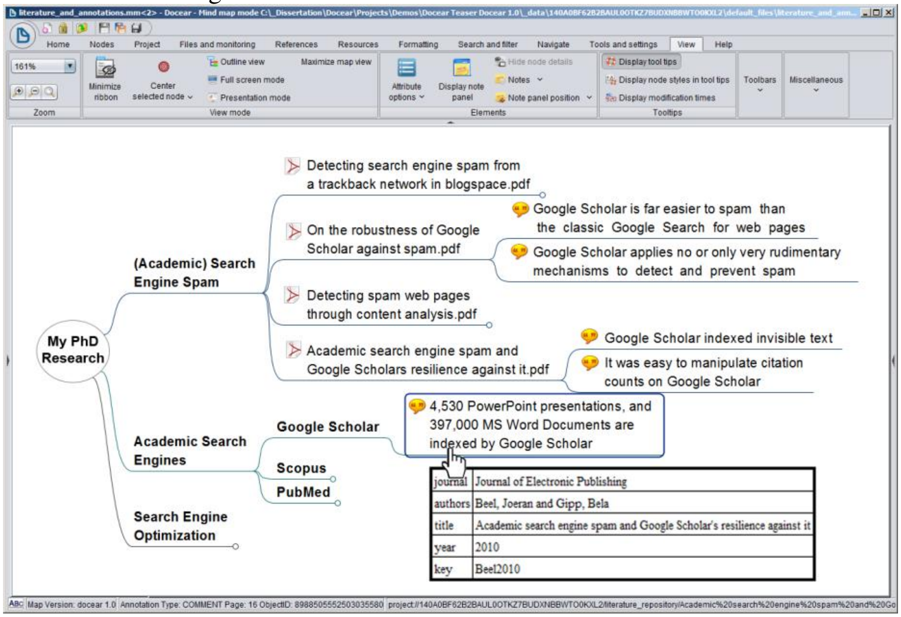
Plista is an advertisement platform that delivers news recommendations to its partners. In 2013, Plista hosted the RecSys News Challenge where participating teams received recommendation requests from Plista, including information about the current users and potential recommendation candidates. The teams generated recommendations in real-time (within 100ms) and sent them back to Plista. Plista then forwarded them to its partners, i.e. news websites. We participated in the challenge in which a number of recommendation approaches were evaluated. The algorithms ranged from recommending the most popular news to content-based filtering and collaborative filtering. We expected that well-performing approaches for one news website would also perform well on other news sites[3]. Effectiveness was measured using *precision* in a "near-to-online" evaluation. When Plista requested recommendations, Plista provided information, such as a list of recommendations a user had seen before, and which recommendations the user had previously clicked. We analyzed the stream of click events provided by Plista. When a click event was received, we sent a request to one (or more) recommendation approaches and asked for predictions of what articles the user would have clicked. The predicted article lists were checked against the clicks observed in the online evaluation. We computed precision by counting an answer as correct if the list of suggestions contained the correct article. Based on this definition, the expected precision grows with the number of suggestions in the predicted list. Compared to an online evaluation, this near-to-online evaluation allowed us to benchmark several different approaches simultaneously, enabling us to efficiently measure the effectiveness of different approaches and

[2] http://plista.com

[3] With “well-performing” we mean if one algorithm was the most effective on a particular news site, it should be the most effective algorithm on other news sites, or at least should be among the most effective.

parameters. For more details on the algorithms and the evaluation, refer to Lommatzsch [2014a; 2014b].

Docear is an open-source desktop software for organizing references and PDFs [Beel et al. 2011]. It is comparable to other reference managers, such as *Zotero* or *Endnote*. While most academic reference managers use tables or social tags to manage PDFs and references, Docear primarily uses mind-maps.Mind-mapping is a visual technique to record and organize information and to develop new ideas [Holland et al. 2004]. Mind-maps consist of three elements: *nodes*, *connections*, and *visual clues*. To create a new mind-map, users gather their ideas around a central concept that is stored in the *root node* [Davies 2011]. Users then create sub-nodes that branch from the root node in the form of child-nodes and sibling-nodes.


literature_and_annotations.mm<2> - Docear - Mind map mode C:_Dissertation\Docear\Projects\Demos\Docear Teaser Docear 1.0_data\140A0BF62B2BAUL0OTKZ7BUDXNBBWTO0KXL2\default_files\literature_and_ann...
Home
Nodes
Project
Files and monitoring
References
Resources
Formatting
Search and filter
Navigate
Tools and settings
View
Help
161%
Zoom
Minimize ribbon
Center selected node
Outline view
Full screen mode
Presentation mode
Maximize map view
View mode
Attribute options
Display note panel
Hide node details
Notes
Note panel position
Elements
Display tool tips
Display node styles in tool tips
Display modification times
Tooltips
Toolbars
Miscellaneous
My PhD Research
(Academic) Search Engine Spam
Detecting search engine spam from a trackback network in blogspace.pdf
On the robustness of Google Scholar against spam.pdf
Detecting spam web pages through content analysis.pdf
Academic search engine spam and Google Scholars resilience against it.pdf
Academic Search Engines
Google Scholar
Scopus
PubMed
Search Engine Optimization

Google Scholar is far easier to spam than the classic Google Search for web pages

Google Scholar applies no or only very rudimentary mechanisms to detect and prevent spam

Google Scholar indexed invisible text

It was easy to manipulate citation counts on Google Scholar

4,530 PowerPoint presentations, and 397,000 MS Word Documents are indexed by Google Scholar

journal	Journal of Electronic Publishing
authors	Beel, Joeran and Gipp, Bela
title	Academic search engine spam and Google Scholar's resilience against it
year	2010
key	Beel2010

Map Version: docear 1.0 Annotation Type: COMMENT Page: 16 ObjectID: 8988505552503035580 project://140A0BF62B2BAUL0OTKZ7BUDXNBBWTO0KXL2/literature_repository/Academic%20search%20engine%20spam%20and%20Go

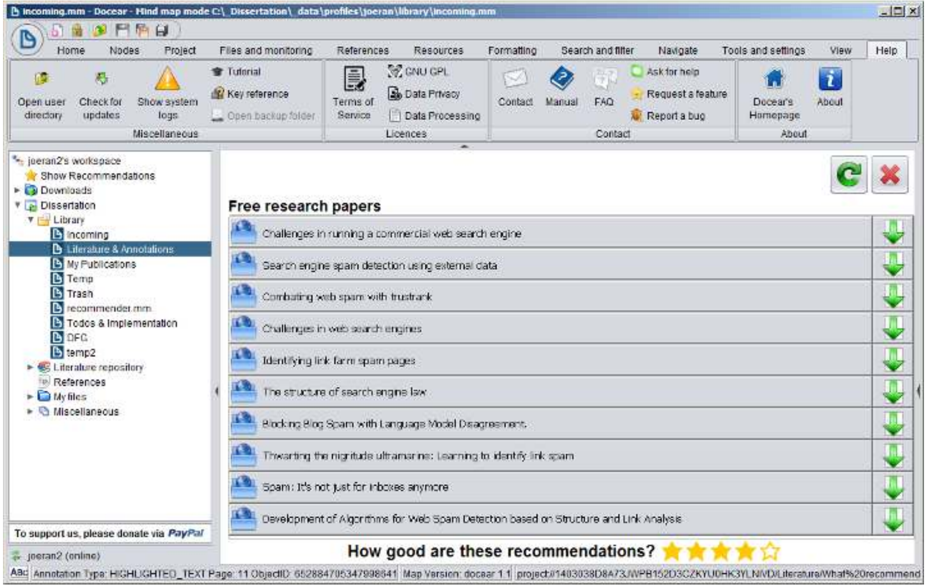
Figure 2: Screenshot of a mind-map in Docear[4]

Figure 2 shows a mind map used to manage academic PDFs and references in Docear. Any annotations made in the PDFs (e.g. comments, highlighted text, and bookmarks) are imported into the mind-map. Clicking the PDF icon opens the linked file. Docear extracts metadata from PDF files (e.g. title and journal name), and displays metadata when the mouse hovers over a PDF icon [Beel et al. 2010; Beel, Langer, Genzmehr and Müller 2013]. A circle at the end of a node indicates that the node has child nodes, which have been collapsed – clicking the circle would unfold the node.

Since 2012, Docear has offered a recommender system for 1.8 million publicly available research papers on the Web [Beel, Langer, Genzmehr and Nürnberger 2013a; Beel, Langer, Gipp, et al. 2014]. When setting up Docear, users can choose if they want to activate the recommender system, and if they want to provide information on their age and gender. Recommendations are displayed as a list of ten research papers showing the titles of the recommended papers (Figure 3). Clicking a recommendation opens the paper in the user's web browser. Recommendations are primarily created via content-based filtering. This means that the recommender system analyses the user's mind-maps, extracts the most frequently occurring terms (or citations), and recommends research

[4] http://www.docear.org

papers that contain the same terms (or citations) as the user's mind-maps. The recommender system randomly selects a number of variables each time recommendations are created. For instance, the user-model size is selected randomly, so user models sometimes contain as few as 10 terms and other times over 100 terms. By introducing these variations, we could evaluate how differences in the algorithms affected the effectiveness of content-based filtering in the same recommendation scenario.



Free research papers

- Challenges in running a commercial web search engine
- Search engine spam detection using external data
- Combating web spam with trustrank
- Challenges in web search engines
- Identifying link farm spam pages
- The structure of search engine law
- Blocking Blog Spam with Language Model Disagreement.
- Thwarting the nigritude ultramarine: Learning to identify link spam
- Spam: It's not just for inboxes anymore
- Development of Algorithms for Web Spam Detection based on Structure and Link Analysis

How good are these recommendations? ★★★★☆

Figure 3: Screenshot of Docear's recommendations[5]

In addition to the two CBF approaches (term-based and citation-based), we apply a stereotype recommendation approach [Rich 1979]. Based on this approach, we generalize that Docear users are researchers. Some users only use Docear for its mind-mapping functionality and not its reference management capability, so this is not strictly true. However, the nature of stereotyping is to generalize, and the majority of our users are researchers. To give recommendations using the stereotype approach, we manually compiled a list of research articles that we assumed to be relevant for researchers in general, namely articles and books about academic writing. If the stereotype recommendation approach is randomly chosen, the pre-compiled list of articles is recommended. We chose the stereotype approach mainly as a baseline and to have an approach that was fundamentally different from content-based filtering. For a detailed overview of the recommender system's architecture and algorithms please refer to [Beel 2015; Beel, Langer, Gipp, et al. 2014; Beel, Langer, et al. 2015].

There are three types of Docear users: registered users, local users, and anonymous users [Beel, Langer, Nürnberger, et al. 2013]. *Local users* chose not to register when they install Docear. Consequently, they cannot use Docear's online services, such as recommendations or online backup. We have no further information about these users, nor do we know how many local users there are. *Registered users* sign up with a username, password, and email address and can use Docear's online services. During the registration process, these users are encouraged to provide information on their age and gender. Between March 2012 and June 2014, around 1,000 users registered every month, resulting in 24,689 registered users. *Anonymous users* decline to register but can still use

[5] http://www.docear.org

some of Docear's online services. In this case, Docear automatically creates a user account with a randomly selected user name that is tied to a user's computer. For anonymous users, we have no information about gender or age. Anonymous users cannot login on Docear's website, but they can receive recommendations since their mind-maps are transferred to Docear's servers if they opt-in. Due to spam issues, anonymous user accounts were deactivated in late 2013. Until then, 9,511 anonymous user accounts had been created by non-spammers.

Our evaluation is based on 3,002 users who received 289,657 recommendations from March 2013 to January 2014. Of these users, 38.62% were anonymous and 61.38% were registered. Of the registered users, 33.17% were males, 6.42% were females, and 21.79% declined to specify (see Figure 4, left chart). Based only on the users who specified their gender, 16.22% were female and 83.78% were male (Figure 4, right). Of the registered users who specified their age, most were between 25 and 34 years old (48.23%) (Figure 5).

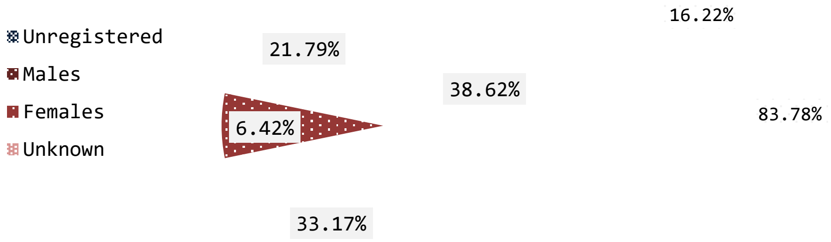


Figure 4: Gender distribution of Docear users

Evaluations in Docear are primarily based on click-through rate (CTR), i.e. the ratio of displayed and clicked recommendations. For instance, if users A, B, and C saw 100, 200, and 1,000 recommendations respectively, all created with the same algorithm, and User A clicked seven recommendations, User B clicked 16, and User C clicked 300, then the overall CTR of the algorithm would be $\frac{7+16+300}{100+200+1000} = 24.85\%$. In the past, researchers have criticized CTR as an evaluation metric, and have questioned the accuracy of displaying only the title of recommended research articles. However, we found CTR based on displaying the title was the most sensible metric for our evaluations [Beel and Langer 2015].

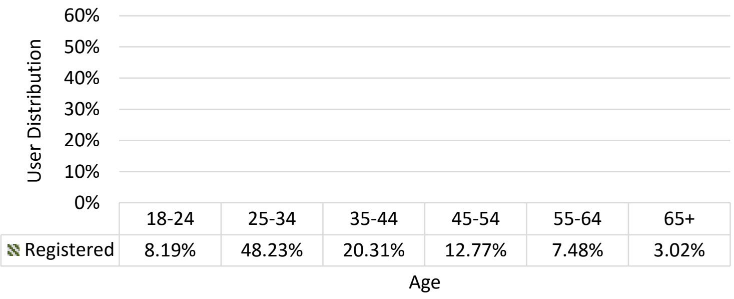


Figure 5: Age distribution of registered users who activated recommendations

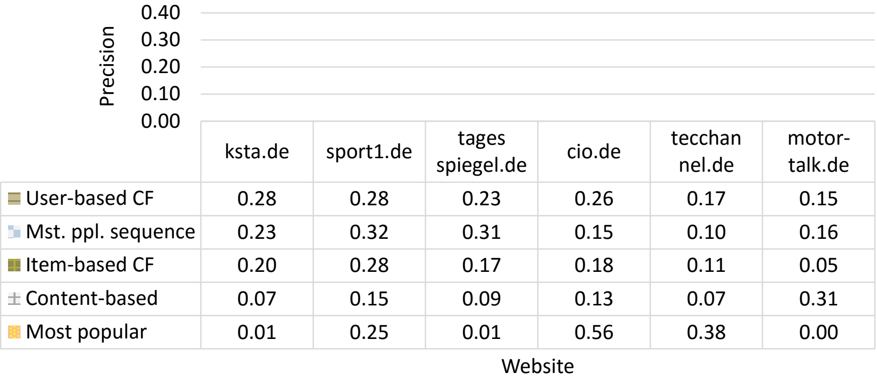
We calculate CTRs for the overall average and different user segments, i.e. for different gender and age groups. Note that the overall average includes all recommender-system users, including those who did not specify their gender or age. This means, CTR of the overall average might differ from the average of males and females or the average of different age groups. For instance, when we report a CTR for males of 5% and for females of 4%, then the overall average will not necessarily lie between 5% and 4%. Instead, the overall average might be higher or lower because users who provided no age or gender information might show a different behavior towards recommendations.

5 Results

The following subsections present the results of our experiments. Unless stated otherwise, the results are statistically significant at the p-0.05 level using a two-tailed t-test. We are publishing the dataset on which the current analysis is based so other researchers can validate our analyses [Beel, Langer, Gipp, et al. 2014].

5.1 News recommendations in difference scenarios

We applied five news recommendation approaches in six scenarios, i.e. on six different news websites[6]. The websites differed in many variables, such as the layout, the news content, the devices on which the news was accessed, and the users themselves. Since these differences are not marginal, we expected some variations in the effectiveness of the five recommendation approaches when applied on the different news-websites. However, since all websites where within the news domain, we expected that an approach performing well on one news website would perform at least reasonably well on other news websites. The results of our evaluation did not confirm our expectation (Figure 6).



	ksta.de	sport1.de	tages spiegel.de	cio.de	tecchan nel.de	motor-talk.de
▤ User-based CF	0.28	0.28	0.23	0.26	0.17	0.15
▨ Mst. ppl. sequence	0.23	0.32	0.31	0.15	0.10	0.16
▦ Item-based CF	0.20	0.28	0.17	0.18	0.11	0.05
± Content-based	0.07	0.15	0.09	0.13	0.07	0.31
▩ Most popular	0.01	0.25	0.01	0.56	0.38	0.00

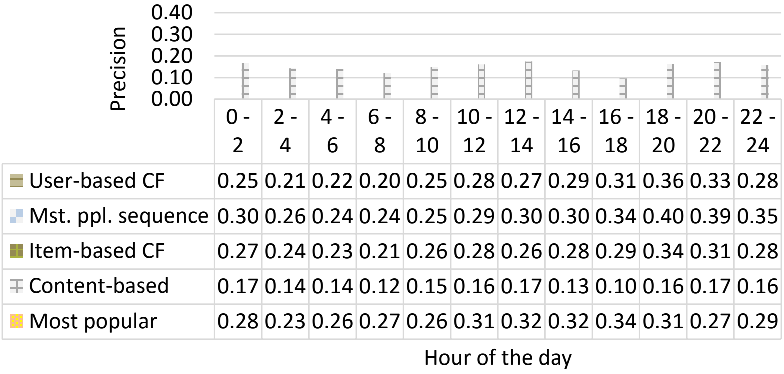
Figure 6: Effect of different scenarios (news websites)

The "most popular" recommendation approach performed best on *Cio.de* (twice as well as the second best approach) and *Techchannel.de*. However, the same approach performed worst on *Ksta.de* (a precision of 0.01 compared to 0.28 for the best approach) and *Tagesspiegel.de*. Content-based filtering performed best on *Motor-Talk.de* (twice as effective as the second best approach), but second worst on *Ksta.de*.

5.2 Time of day

The effectiveness of the news recommendation approaches depended, among other factors, on the time of day. For instance, on *sport1.de*, the "most popular sequence" algorithm performed best between 4pm and 4am (Figure 7). At other times, recommending the "most popular" news performed best. In addition, user-based CF performed better than item-based CF between 12pm and 8pm. During the remaining hours, item-based CF performed better than user-based CF. We observed similar differences on the other news websites.

[6] For more details on the algorithms and the evaluation, refer to Lommatzsch [2014a; 2014b].

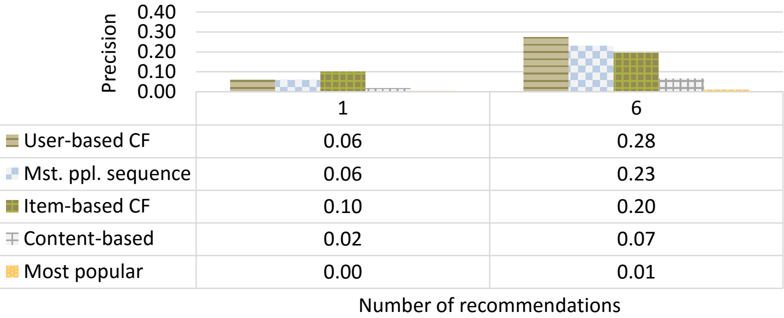


	0 - 2	2 - 4	4 - 6	6 - 8	8 - 10	10 - 12	12 - 14	14 - 16	16 - 18	18 - 20	20 - 22	22 - 24
▤ User-based CF	0.25	0.21	0.22	0.20	0.25	0.28	0.27	0.29	0.31	0.36	0.33	0.28
░ Mst. ppl. sequence	0.30	0.26	0.24	0.24	0.25	0.29	0.30	0.30	0.34	0.40	0.39	0.35
▦ Item-based CF	0.27	0.24	0.23	0.21	0.26	0.28	0.26	0.28	0.29	0.34	0.31	0.28
╪ Content-based	0.17	0.14	0.14	0.12	0.15	0.16	0.17	0.13	0.10	0.16	0.17	0.16
▩ Most popular	0.28	0.23	0.26	0.27	0.26	0.31	0.32	0.32	0.34	0.31	0.27	0.29

Figure 7: Impact of 'time of day' on effectiveness of recommendation approaches

5.3 Number of requested recommendations

Another factor that affected the news recommendation effectiveness was the number of requested recommendations. On *Ksta.de*, user-based CF performed best when six recommendations were requested (see Figure 8). However, if only one recommendation was requested, item-based CF performed nearly twice as well as user-based CF. We observed similar effects on most other news websites.



	1	6
▤ User-based CF	0.06	0.28
░ Mst. ppl. sequence	0.06	0.23
▦ Item-based CF	0.10	0.20
╪ Content-based	0.02	0.07
▩ Most popular	0.00	0.01

Figure 8: Impact of number of recommendations on recommendation effectiveness

Unfortunately, we had had no access to further information on users or devices used. To further explore the factors that affect recommender effectiveness, we next used Docear's recommender system, which gave us access to user information.

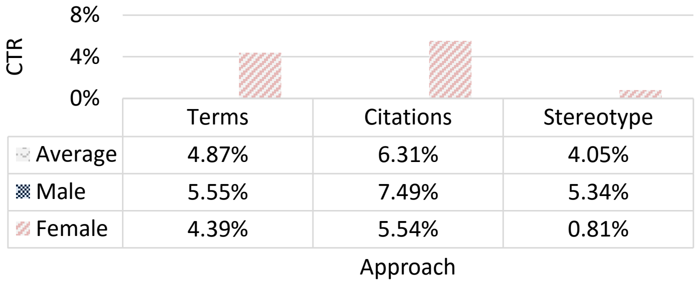
5.4 Gender and age

In Docear, we applied three recommendation approaches, i.e. two content-based filtering approaches (CBF) – one based on terms and one based on citations – and a stereotype approach (cf. Section 4). On average, citation-based CBF performed best (CTR = 6.31%), term-based CBF performed second best (CTR = 4.87%) and stereotype recommendations came in last, although they were still reasonably effective (CTR = 4.05%).

In regard to gender, there was no significant difference among the users for the two CBF approaches: citation-based CBF was more effective than term-based CBF for both genders (Figure 9). However, there were significant differences in the effectiveness of the stereotype approach. While males had an average CTR of 5.34% for stereotype recommendations, females had an average CTR of 0.81%.

We performed the same analysis for different age groups but could not find any significant differences between the approaches' effectiveness.

These results show that recommender effectiveness differs depending on the user population. This finding is perhaps unsurprising, but its implication is that if the three approaches were evaluated in any scenario with a different gender ratio than in the Docear scenario, the average effectiveness of the stereotype recommendations is no longer reproducible.

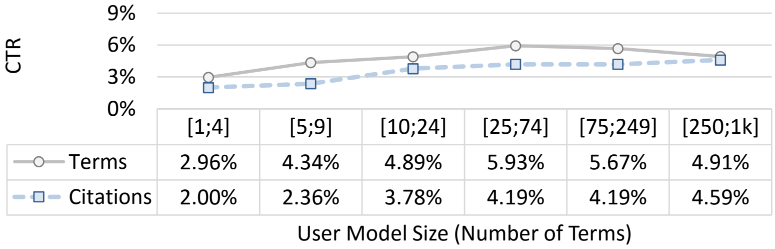


	Terms	Citations	Stereotype
Average	4.87%	6.31%	4.05%
Male	5.55%	7.49%	5.34%
Female	4.39%	5.54%	0.81%

Figure 9: Effect of recommendation approaches and gender

5.5 User-model size & feature-type

The effectiveness of the CBF approaches also depended on user-model size. User-model size denotes the number of terms (or citations) extracted from the user's mind maps to express the user's interests. For instance, a user model with a user-model size of ten would contain the ten most frequent terms that occur in a user's mind-maps. Any additional terms in the user's mind-maps would be ignored for identifying recommendations. Limiting a user model to a certain size is a common approach, because too many terms in a user model can introduce noise into the recommendation process.



	[1;4]	[5;9]	[10;24]	[25;74]	[75;249]	[250;1k]
Terms	2.96%	4.34%	4.89%	5.93%	5.67%	4.91%
Citations	2.00%	2.36%	3.78%	4.19%	4.19%	4.59%

Figure 10: Effect of user-model size (terms and citations)

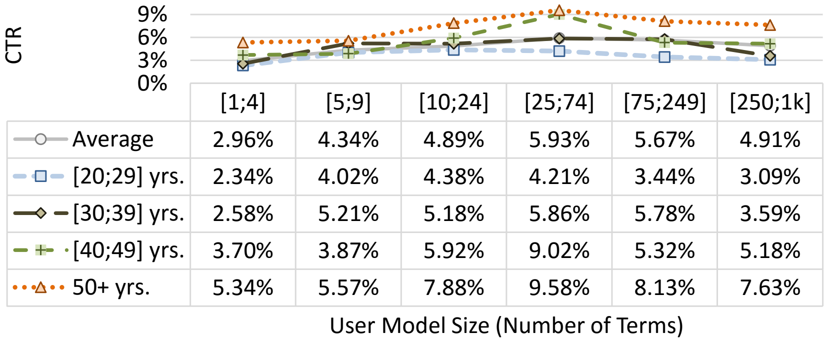
For Docear's recommendation approaches, the optimal user-model size varied depending on the feature type, i.e. for user models consisting of terms a different user model size was optimal than for user models consisting of citations (Figure 10). For term-based CBF the optimal user-model size was between 25 and 74 terms (CTR = 5.93%). For citation-based CBF, the optimal user-mode size was 250 and more citations (CTR = 4.19%).

5.6 User-model size and user characteristics

The optimal user-model size did not only depend on the feature type, but also on the age of the users (Figure 11).

For users aged 20 to 29, the optimal user-model size was smaller (between 10 and 24 terms) than for the overall average (between 25 and 74 terms). For all other age groups, the same user-model size was optimal as for the overall average (25 – 74 terms). CTRs were generally higher for older users compared to the overall average CTR. Additionally, for younger users, a user-model size between five and nine terms achieved a CTR that was nearly as good as the optimum CTR of the user model size of 10 – 24 terms. We can only speculate about the reasons, but younger users might use a more restricted terminology, so fewer terms describe their information needs sufficiently. Or, older users might be doing more interdisciplinary work, so more terms are needed to describe their information needs comprehensively.

We performed the same analysis for males and females and found no significant difference in the CTR.

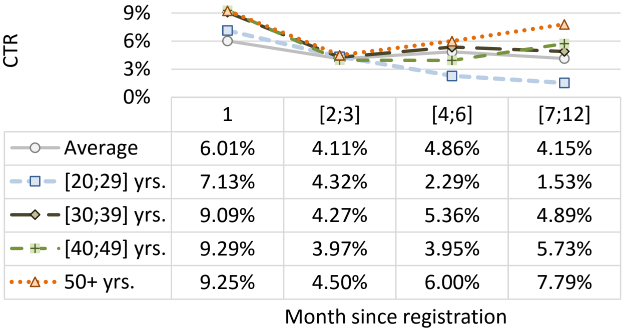


	[1;4]	[5;9]	[10;24]	[25;74]	[75;249]	[250;1k]
○ Average	2.96%	4.34%	4.89%	5.93%	5.67%	4.91%
□ [20;29] yrs.	2.34%	4.02%	4.38%	4.21%	3.44%	3.09%
◇ [30;39] yrs.	2.58%	5.21%	5.18%	5.86%	5.78%	3.59%
+ [40;49] yrs.	3.70%	3.87%	5.92%	9.02%	5.32%	5.18%
△ 50+ yrs.	5.34%	5.57%	7.88%	9.58%	8.13%	7.63%

Figure 11: Effect of user-model size (age)

5.7 Duration of use

Recommendation effectiveness also depended on the duration of use of Docear (and its recommender system). New Docear users, who received recommendations in the first month after registration, had an average CTR of 6.01% (Figure 12). In the second and third month after registration, CTR decreased to 4.11% and on average remained between 4% and 5% in the following months.



	1	[2;3]	[4;6]	[7;12]
○ Average	6.01%	4.11%	4.86%	4.15%
□ [20;29] yrs.	7.13%	4.32%	2.29%	1.53%
◇ [30;39] yrs.	9.09%	4.27%	5.36%	4.89%
+ [40;49] yrs.	9.29%	3.97%	3.95%	5.73%
△ 50+ yrs.	9.25%	4.50%	6.00%	7.79%

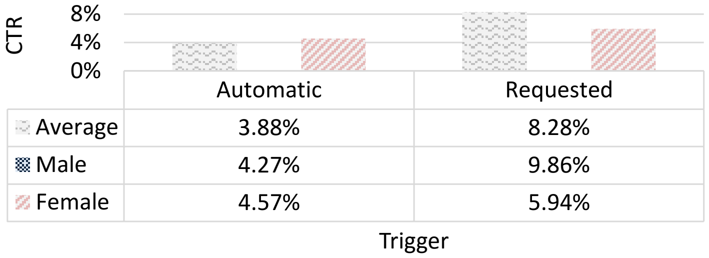
Figure 12: Effect of usage duration and user age on CTR

Looking at the age groups revealed more information. For younger users, aged 20 to 29, CTR continuously decreases over time, down to 1.53% when Docear was used for 7 months or more. For older users (50+), CTR initially decreases, but then increases again to 7.79%. This means, in a scenario with primarily older users, Docear's term-based CBF approach could be considered effective, since CTR tends to increase over time. In a scenario with primarily younger users, using Docear's approach would not be effective because CTR decreases over time. These observations show that the effectiveness of recommendation algorithms are – as to be expected – situation-dependent and that *user's age* and *usage duration* are just two more determinants that should be clearly specified by researchers when evaluating recommendation systems.

5.8 Recommendation trigger

Docear has two methods by which recommendations are triggered, i.e. generated to be shown to users [Beel, Langer, Genzmehr and Nürnberger 2013b]. First, Docear displays recommendations automatically every five days when Docear starts. Second, users can explicitly request recommendations at any time.

On average, CTR for requested recommendations was around twice that of automatically shown recommendations (8.28% vs. 3.88%). For automatically displayed recommendations, there was no significant difference between male and female users – CTR for each group was slightly over 4% (Figure 13). However, there was a significant difference for requested recommendations. CTR for recommendations requested by females was only slightly higher than for automatically displayed recommendations (5.94% vs 4.57%). For males, CTR was more than twice as high (9.50% vs. 4.27%).

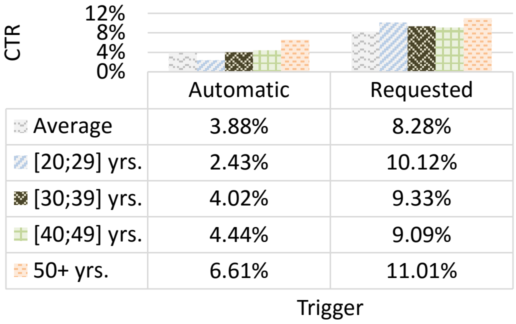


	Automatic	Requested
Average	3.88%	8.28%
Male	4.27%	9.86%
Female	4.57%	5.94%

Figure 13: Effect of recommendation trigger method (automatic vs. requested) and user gender on CTR

These results have at least two implications. First, the trigger method by which recommendations are delivered makes a significant difference – in this case, especially for males. Second, in spite of what has been indicated by previous results, males do not generally have higher click-through rates than females. Instead, males only have higher CTRs for requested recommendations.

CTR for the two trigger methods differed also by age (Figure 14). For automatically displayed recommendations, there was a positive correlation between user age and CTR – the older the users, the higher the CTR tended to be. Young users, aged 20 to 29, had an average CTR of 2.42%, while older users, aged 50+, had an average CTR of 6.61%. However, for requested recommendations there was no clear trend for the different user age groups. Apparently, users of all ages have similar interest in recommendations when they explicitly request them, but not when recommendations are shown automatically.



	Automatic	Requested
Average	3.88%	8.28%
[20;29] yrs.	2.43%	10.12%
[30;39] yrs.	4.02%	9.33%
[40;49] yrs.	4.44%	9.09%
50+ yrs.	6.61%	11.01%

Figure 14: Effect of recommendation trigger and user age

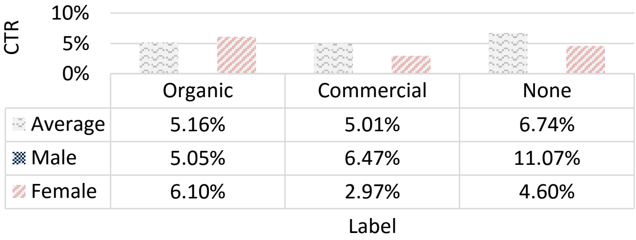
5.9 Labeling[7]

By default, Docear labels its recommendations with the text “Free research papers” (Figure 3). Recently, we researched whether different labels affect click-through rates even if the recommendation approaches remained identical [Beel, Langer and Genzmehr 2013]. For each user, Docear randomly selected a label implying that the recommendations were organic (“*Free research papers*”), commercial (“*Full-text research papers (Advertisement)*”), or no label was assigned. Click-through rates were highest for no label and, unsurprisingly, CTR for organic labels was higher than for commercial labels.

When we repeated the analysis but distinguished between males and females the results were different (Figure 15). For males, recommendations with no label achieved the highest CTR. In contrast, for females organic recommendations achieved the highest CTR. It was also interesting that males had higher CTRs for commercial than for organic recommendations.

This analysis shows that the results of our previous paper would not be reproducible in scenarios with a different user population. While it might be sensible to use no label in scenarios with primarily male users, scenarios with primarily female users should use a label that indicates the organic nature of the recommendations.

We performed the same analysis for different age groups and found no significant differences in the CTR.



	Organic	Commercial	None
Average	5.16%	5.01%	6.74%
Male	5.05%	6.47%	11.07%
Female	6.10%	2.97%	4.60%

Figure 15: Effect of the labeling of recommendations on CTR

[7] Please note that the results of this section are not statically significant, and a further analysis based on more data is required.

6 Discussion & Conclusions

Our analysis shows that minor changes in the experimental setup – i.e. in recommendation scenarios, approaches, or evaluations – may lead to unpredictable variations in the effectiveness of recommendation approaches. Consequently, the examples of apparently not reproducible experiments that we found in the literature (cf. Section 1) seem plausible to a large extent. Nevertheless, we do not advocate for accepting the current state as "just the way things are". On the contrary, we are confident that today's degree of reproducibility in recommender-systems research can be increased significantly.

To make recommender-systems research more reproducible, we suggest seven actions that we outline in the following sections[8].

6.1 Survey and learn from other research fields

In the medical sciences, social sciences, and natural sciences the issue of reproducibility has already been discussed and analyzed in depth [Collaboration and others 2015; Deyo et al. 1991; Downing 2004; Flyvbjerg 2001; Hoeymans et al. 1997; McNutt 2014; Rothwell and Martyn 2000; Schmidt 2009]. Even in information retrieval and machine learning, significant efforts have been made to standardize evaluations and ensure reproducibility [Castilho and Gurevych 2011; Guyon and Elisseeff 2003; Hawking et al. 1999; Sonnenburg et al. 2007; Voorhees 2005]. This does not mean that these disciplines have provided the ultimate answers to ensuring reproducibility. However, the work that has been done might be helpful for the recommender-system community. Therefore, we suggest to conduct a thorough literature review on reproducibility in other research disciplines to learn how their findings can help in accomplishing the following actions.

6.2 Find a common understanding of reproducibility

The recommender-system community needs to agree on a common understanding of how important reproducibility is, and what degree of reproducibility we want (and can) achieve. This agreement should be based on a discussion involving large parts of the recommender-system community. This, in turn, requires raising the awareness of reproducibility. To raise the awareness, there are numerous options. Workshops about reproducibility, such as RepSys [Bellogin et al. 2014], should be continued and extended. Conferences and journals should place more emphasis on reproducibility when reviewing articles. Leading figures in our community could advocate the importance of reproducibility, and text-books and lectures about recommender systems should cover the topic of reproducibility in more detail.

We are confident that the community will agree on the fundamental importance of reproducibility, and that the current state of ensuring reproducibility can be improved. However, even if difficulties in reproducing experimental results were to be accepted as "just the way things are", many questions need to be answered. Assuming that "there is no better way than to test and optimize a number of alternatives", then how should a number of promising alternatives be determined? Judging by the research papers we surveyed, researchers must not only consider a *few* promising alternatives for generating research-paper recommendations; but instead they must consider *all* alternatives, because

[8] Several suggestions are inspired from Ekstrand, Ludwig, Konstan, et al. [2011] and Konstan and Adomavicius [2013].

each approach was most effective in at least one evaluation [Beel 2015; Beel, Gipp, et al. 2015]. Can it be that researchers in need of a baseline should just pick a few approaches randomly?

Assuming a few promising approaches were identified, how should they be optimized? There is no list of variables that might be worth optimizing, and even if there was, there would be dozens of variables, each with dozens or even hundreds of possible values that would require testing. How can a researcher or a provider of a recommender system perform these optimizations in a reasonable amount of time? In addition, recommender systems change over time. What does this mean for the provider of a recommender-system who wants to use an effective recommendation approach? Would the provider have to reevaluate the alternative recommendation approaches every time items are added to the recommendation corpus, when new users register, or when minor features of the recommender system are changed? Finally, why would we continue performing evaluations at all, if they do *not* help in predicting recommendation effectiveness in other scenarios?

6.3 Identify and understand the determinants affecting reproducibility

It is known *that* differences in recommendation scenarios, approaches, and evaluations affect recommendation effectiveness and hence reproducibility. However, it remains mostly unknown *what* differences affect recommendation effectiveness *how strong*, and *why*. Therefore, the community needs to identify and understand the determinants that affect reproducibility.

With respect to recommendation *approaches*, the number of variables affecting reproducibility is rather limited. For instance, standard content-based filtering approaches could differ by the feature type (terms, n-grams, citations, …), the document field from which the features are extracted (title, abstract, body, …), the weighting scheme (TF, TF-IDF, …), user-model size (1, 2, 3, …), and a few additional variables. Identifying all variables is probably a manageable task. Evaluating the impact of the variables, and which of the variables' parameters (or range of parameters) is most effective would be more challenging but probably still manageable (as long as interactions between variables are ignored).

With respect to recommendation *evaluations*, the number of variables seems manageable as well. There are only three major evaluation methods (user studies, online evaluations, and offline evaluations), each with some variations (e.g. real-world and lab user studies) and some evaluation metrics. We are confident that with some more research and discussions, the community will be capable of understanding the differences of the evaluation methods and metrics and defining evaluation standards that contribute to greater reproducibility.

With respect to recommendation *scenarios*, the number of variables seems larger and more difficult to measure than for recommendation approaches and evaluations. For instance, a *user population* alone could vary by age, gender, profession, religion, nationality, education, income, language, country of residence, and many factors more (we began investigating this issue for Docear's users, cf. Langer and Beel [2014]). The *recommendation corpus* could vary by many different factors as well, for instance to take the example of research papers: the number of papers in the corpus, the papers' page count, citation counts, language, discipline, and so on. The *devices* used to access the recommender system can vary (e.g. desktop, tablet, or mobile) along with the *operating systems* and *software programs* in which the recommender systems are embedded. In addition, *user interfaces* would be very difficult to formally describe. We consider this variety of variables and parameters in the recommendation scenarios as a huge research challenge that will likely only be contained in some dimensions; however, any progress made will contribute to higher reproducibility of future research.

Another major challenge lies in the *interactions* between variables. With no interactions, several hundred or thousand variables would have to be analyzed, each with maybe a few dozens or hundreds of potential values. However, considering interactions among the hundreds or thousands of variables leads to millions of potential combinations that would require analysis. Hence, aiming at understanding these interactions is a mammoth task with uncertain outcome.

6.4 Conduct more comprehensive experiments

We suggest that researchers conduct experiments that are more comprehensive. This implies that researchers vary the parameters of their recommendation approach and use, for instance, different user-model sizes and weighting schemes. Ideally, they would also use different evaluation methods and metrics, and vary the scenario in which they apply the recommendation approach. “Varying the scenario” does not necessarily mean to explicitly modify a scenario or applying a recommendation approach in truly different scenarios (though this certainly would be preferable). It might be sufficient to do the analyses for different user groups (e.g. age groups or gender), which, in turn, would require large user groups to arrive at statistically significant results.

Conducting experiments that are more comprehensive would be beneficial in two ways. First, it would help identifying the optimal configuration of a recommendation approach. As long as researchers randomly choose parameter values, for example the user-model size, the effectiveness of their approach and meaningfulness of their evaluation will probably be suboptimal. Currently, selecting a random user-model size is common practice, at least in the domain of research-paper recommender systems [Beel, Gipp, et al. 2015]). Second, it would help to understand the underlying determinants that are affecting reproducibility (cf. Section 6.3). If, for instance, all recommender-system researchers in the past had published details on the gender of their users, we would probably have a much better understanding today on whether and how gender affects recommendation effectiveness.

6.5 Modernize publication practices

We propose to modernize the current publication practices in two ways.

First, more details about recommendation approaches, scenarios, and evaluations have to be reported in publications. Our results showed that variables, such as user-model size and user population, may affect recommendation effectiveness by a factor of two and higher. Consequently, we consider it essential that all variables that authors are aware of are disclosed in the publications. Researchers trying to reproduce the experiments should not be left to guess the variables. This could lead to significant differences in recommendation effectiveness, and hence the experimental results would not be reproducible. It might seem like an obvious demand that researchers should publish all relevant details about their recommendation approaches, scenarios, and evaluations. However, at least in the domain of research-paper recommender systems, most authors provide only superficial information and, for instance, do not report details on user-model size or the user population [Beel, Gipp, et al. 2015]. Publishing more details might require publishers to loosen page limits, particularly for conference submissions.

Second, data described in the publication needs to be made available. A scientific publication is only the summary of the research performed, or as Buckheit and Donoho [1995] put it: an "advertisement" for the actual research. To fully understand and reproduce the research, more information than just the publication is needed. When it comes to recommender systems, this supplementary data can include the source code of the recommendation approach, or a ready-to-run software library for a recommendation framework. It can also include the datasets used in the evaluation, calculations, and raw data from the data analysis tools (SPSS, R, Excel, etc.). This also means that new methods for reviewing and publishing additional data are required. Innovations such as Elsevier's "Executable Paper" might be one option to combine a normal publication with additional data and functionality [Koers et al. 2013].

6.6 Foster the development and use of recommender-system frameworks

Publishing ready-to-run implementations for recommender-systems frameworks would certainly be a great advancement for the recommender-system community. However, the current state of recommender-system frameworks faces two problems.

First, a comprehensive overview of the existing frameworks is missing. Currently, there is more than a dozen recommender-system (evaluation) frameworks: LensKit [Ekstrand, Ludwig, Kolb, et al. 2011], recommenderlab [Hahsler 2011], EasyRec, Mahout, MyMediaLite [Gantner et al. 2011], LibRec [Guo et al. 2015], Duine, RiVaL [Said and Bellogin 2014], and TagRec [Kowald et al. 2014] are just some of them. Finding the best framework for one's purpose is a time consuming task due to a missing overview of the frameworks' strengths and weaknesses. Consequently, we suggest that the community create a comprehensive overview of the existing recommendation frameworks.

Second, most frameworks are developed by rather small groups. Consequently, many frameworks might not provide all features that researchers require or the frameworks may be no longer maintained. For instance, Duine and EasyRec are not maintained any more, and when we implemented a recommender system for our reference manager Docear [Beel, Langer, Genzmehr and Nürnberger 2013a; Beel, Langer, Gipp, et al. 2014], we chose to develop our own proprietary solution because none of the frameworks fulfilled our needs. Admittedly, we had very special needs, as we wanted to create user models based on mind-maps, which is a unique recommendation scenario. Nevertheless, given the importance of recommender-system frameworks, we advocate that measures are taken to ensure the long-term sustainability and make recommender frameworks more powerful. How these measures should look like could be discussed by the community.

6.7 Create and establish best-practice guidelines

Once, the previous steps have been completed, best-practice guidelines should be created. These guidelines should cover how to conduct recommender-systems evaluations, including instructions on which evaluation methods and metrics to use, which parameters to vary, and how to vary them, how to formally describe recommendation scenarios, how to report the results in publications, which data to publish in addition to a "normal" publication, and which recommender-system frameworks should be used. Equally important will be to ensure that reviewers judge how stringent the authors of submitted papers followed the best-practice guidelines.

7 Summary

In the field of recommender-system research, experimental results are often difficult to reproduce: similar recommendation scenarios, approaches, or evaluations lead to large and unexpected differences in experimental results. This unpredictability is attributable to the large number of contextual determinants that heavily influence the outcome of recommender-systems evaluations.

Our research goal was to obtain ideas what determinants might affect reproducibility of experimental results. To achieve the goal, we experimented with the news recommender system Plista and our research-paper recommender system of Docear. More precisely, we a) varied the recommendation scenarios and kept using the same implementations and evaluation methods, and b) varied the recommendation approaches and kept the same scenarios and evaluation methods.

Recommendation *scenarios* varied in our experiments, among others, by the time of the day, recommendation labels, users' gender and age, number of requested recommendations, and the duration and intensity with which the recommender system was used. All these variables had an impact on recommendation effectiveness and hence the reproducibility of experimental results. In some cases, the variations in the scenarios led to very strong variations in the effectiveness. For instance, stereotype recommendations performed much better for males (CTR = 5.34%) than for females (CTR = 0.81%). Recommendations *explicitly* requested by users had CTRs twice as high as *automatically* displayed recommendations – but only for males. For females, CTR remained about the same. For young users, the effectiveness of content-based filtering constantly decreased the longer the users were using the system. For older users, the effectiveness remained stable or even increased over time. When multiple variables among the scenarios differed, the effectiveness of a recommendation approach became unpredictable. For instance, the "most popular" recommendation approach performed best on *Cio.de* but worst on *Ksta.de*. Content-based filtering performed best, by far, on *Motor-Talk.de*, but second worst on *Ksta.de*. Depending on the time of the day, the effectiveness of recommendation approaches sometimes even completely changed. Between 12pm and 8pm, user-based CF performed better than item-based CF. During the remaining hours, it was the opposite (item-based CF performed better than user-based CF).

Recommendation *approaches* varied in our experiments by the feature type (terms or citations) and user model size. A CBF approach based on citations was more effective than a CBF approach based on terms. In addition, the user model size had a strong impact. For term-based CBF, a user model size of 25–74 terms was around twice as effective as 1–4 terms (CTR of 5.93% vs. 2.96%). Most researchers would probably expect intuitively that recommendation effectiveness depends on user-model size and feature type. However, to our knowledge, it has not been shown empirically how strong the differences are, at least not in the field of research-paper recommender systems. Again, these examples show that minor variations may lead to large discrepancies in the recommendation effectiveness. Consequently, as long as not all details of a recommendation approach are known, estimates about the effectiveness will be subject to high uncertainty.

Recommendation *evaluations* were not the subject to our current research. However, based on our previous research [Beel and Langer 2015] and current literature review, we conclude that differences in evaluation methods and metrics may lead to different assessments in recommender effectiveness, and hence to difficulties in reproducing experimental results.

We further found that the variables in the recommendation scenarios (e.g. age) and the approaches (e.g. user-model size) *interrelated* with each other and hence affected recommendation effectiveness and reproducibility. For instance, the optimal user model size depended on the users' age. For users age 20 to 29, the optimal user-model size was smaller (between 10 and 24 terms) than for the other age groups (between 25 and 74 terms). Whether age directly affects the optimal user model size remains speculation. It could be that there are some other determinants that are responsible for the effect and that correlate with age.

Our results demonstrate the challenge of achieving reproducibility in recommender-systems research. When minor variations in recommendation approaches, scenarios and evaluations lead to major changes in the recommendation effectiveness, it becomes difficult to reproduce research results. Some researchers accept these difficulties as "just the way things are". Others, including ourselves, believe that the current state of reproducibility in recommender-system research leaves room for significant improvement. Improving the status quo will be a gradual journey given the large task at hand. Nevertheless, we want to encourage the community to tackle this challenge by partaking in the following actions:

1. Surveying and learning from other research fields
2. Finding a common understanding of reproducibility
3. Identifying and understanding the factors that affect reproducibility
4. Conducting more comprehensive experiments
5. Adjusting publication practices
6. Fostering the development and use of recommender-system framework
7. Creating and establishing best-practice guidelines

We are confident that by pursuing these actions, reproducibility of recommender-systems research can be enhanced. This, in turn, would ease future research, increase the value of individual research contributions, and support the operators of recommender systems who seek the most effective recommendation approaches for their use case.

8 References

Xavier Amatriain, Jjosep Pujol, and Nurias Oliver. 2009. I like it... i like it not: Evaluating user ratings noise in recommender systems. *User Modeling, Adaptation, and Personalization* (2009), 247–258.

Joeran Beel. 2015. Towards Effective Research-Paper Recommender Systems and User Modeling based on Mind Maps. *PhD Thesis. Otto-von-Guericke Universität Magdeburg* (2015).

Joeran Beel, Bela Gipp, Stefan Langer, and Corinna Breitinger. 2015. Research Paper Recommender Systems: A Literature Survey. *International Journal on Digital Libraries* (2015), 1–34. DOI:http://dx.doi.org/10.1007/s00799-015-0156-0

Joeran Beel, Bela Gipp, Stefan Langer, and Marcel Genzmehr. 2011. Docear: An Academic Literature Suite for Searching, Organizing and Creating Academic Literature. In *Proceedings of the 11th Annual International ACM/IEEE Joint Conference on Digital Libraries (JCDL).* JCDL '11. ACM, 465–466. DOI:http://dx.doi.org/10.1145/1998076.1998188

Joeran Beel, Bela Gipp, Ammar Shaker, and Nick Friedrich. 2010. SciPlore Xtract: Extracting Titles from Scientific PDF Documents by Analyzing Style Information (Font Size). In M. Lalmas, J. Jose, A. Rauber, F. Sebastiani, & I. Frommholz, eds. *Research and Advanced Technology for Digital Libraries, Proceedings of the 14th European Conference on Digital Libraries (ECDL'10).* Lecture Notes of Computer Science (LNCS). Glasgow (UK): Springer, 413–416.

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 Sarantos Kapidakis, Cezary Mazurek, & Marcin Werla, eds. *Proceedings of the 19th International Conference on Theory and Practice of Digital Libraries (TPDL).* Lecture Notes in Computer Science. 153–168. DOI:http://dx.doi.org/10.1007/978-3-319-24592-8_12

Joeran Beel, Stefan Langer, and Marcel Genzmehr. 2013. Sponsored vs. Organic (Research Paper) Recommendations and the Impact of Labeling. In Trond Aalberg, Milena Dobreva, Christos Papatheodorou, Giannis Tsakonas, & Charles Farrugia, eds. *Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries (TPDL 2013).* Valletta, Malta, 395–399.

Joeran Beel, Stefan Langer, Marcel Genzmehr, and Bela Gipp. 2014. Utilizing Mind-Maps for Information Retrieval and User Modelling. In Vania Dimitrova, Tsvi Kuflik, David Chin, Francesco Ricci, Peter Dolog, & Geert-Jan Houben, eds. *Proceedings of the 22nd Conference on User Modelling, Adaption, and Personalization (UMAP).* Lecture Notes in Computer Science. Springer, 301–313. DOI:http://dx.doi.org/10.1007/978-3-319-08786-3_26

Joeran Beel, Stefan Langer, Marcel Genzmehr, Bela Gipp, Corinna Breitinger, and Andreas Nürnberger. 2013a. Research Paper Recommender System Evaluation: A Quantitative Literature Survey. In *Proceedings of the Workshop on Reproducibility and Replication in Recommender Systems Evaluation (RepSys) at the ACM Recommender System Conference (RecSys).* ACM International Conference Proceedings Series (ICPS). ACM, 15–22. DOI:http://dx.doi.org/10.1145/2532508.2532512

Joeran Beel, Stefan Langer, Marcel Genzmehr, Bela Gipp, and Andreas Nürnberger. 2013b. A Comparative Analysis of Offline and Online Evaluations and Discussion of Research Paper Recommender System Evaluation. In *Proceedings* *of the Workshop on Reproducibility and Replication in Recommender Systems Evaluation (RepSys) at the ACM Recommender System Conference (RecSys)*. ACM International Conference Proceedings Series (ICPS). 7–14. DOI:http://dx.doi.org/10.1145/2532508.2532511

Joeran Beel, Stefan Langer, Marcel Genzmehr, and Christoph Müller. 2013. Docears PDF Inspector: Title Extraction from PDF files. In *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'13)*. ACM, 443–444. DOI:http://dx.doi.org/10.1145/2467696.2467789

Joeran Beel, Stefan Langer, Marcel Genzmehr, and Andreas Nürnberger. 2013a. Introducing Docear's Research Paper Recommender System. In *Proceedings of the 13th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'13)*. ACM, 459–460. DOI:http://dx.doi.org/10.1145/2467696.2467786

Joeran Beel, Stefan Langer, Marcel Genzmehr, and Andreas Nürnberger. 2013b. Persistence in Recommender Systems: Giving the Same Recommendations to the Same Users Multiple Times. In Trond Aalberg, Milena Dobreva, Christos Papatheodorou, Giannis Tsakonas, & Charles Farrugia, eds. *Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries (TPDL 2013)*. Lecture Notes of Computer Science (LNCS). Valletta, Malta: Springer, 390–394.

Joeran Beel, Stefan Langer, Bela Gipp, and Andreas Nürnberger. 2014. The Architecture and Datasets of Docear's Research Paper Recommender System. *D-Lib Magazine* 20, 11/12 (2014). DOI:http://dx.doi.org/10.1045/november14-beel

Joeran Beel, Stefan Langer, Georgia M. Kapitsaki, Corinna Breitinger, and Bela Gipp. 2015. Exploring the Potential of User Modeling based on Mind Maps. In Francesco Ricci, Kalina Bontcheva, Owen Conlan, & Séamus Lawless, eds. *Proceedings of the 23rd Conference on User Modelling, Adaptation and Personalization (UMAP)*. Lecture Notes of Computer Science. Springer, 3–17. DOI:http://dx.doi.org/10.1007/978-3-319-20267-9_1

Joeran Beel, Stefan Langer, Andreas Nürnberger, and Marcel Genzmehr. 2013. The Impact of Demographics (Age and Gender) and Other User Characteristics on Evaluating Recommender Systems. In Trond Aalberg, Milena Dobreva, Christos Papatheodorou, Giannis Tsakonas, & Charles Farrugia, eds. *Proceedings of the 17th International Conference on Theory and Practice of Digital Libraries (TPDL 2013)*. Valletta, Malta: Springer, 400–404.

Alejandro Bellogin, Pablo Castells, Alan Said, and Domonkos Tikk. 2014. Report on the workshop on reproducibility and replication in recommender systems evaluation (RepSys). In *ACM SIGIR Forum*. ACM, 29–35.

Steven Bethard and Dan Jurafsky. 2010. Who should I cite: learning literature search models from citation behavior. In *Proceedings of the 19th ACM international conference on Information and knowledge management*. ACM, 609–618.

Jesús Bobadilla, Fernando Ortega, Antonio Hernando, and Abraham Gutiérrez. 2013. Recommender systems survey. *Knowledge-Based Systems* 46 (2013), 109–132.

Toine Bogers and Antal van den Bosch. 2007. Comparing and Evaluating Information Retrieval Algorithms for News Recommendation. In *RecSys'07*. Minneapolis, Minnesota, USA: ACM, 141–144.

Toine Bogers and Antal van den Bosch. 2008. Recommending scientific articles using citeulike. In *Proceedings of the 2008 ACM conference on Recommender systems*. ACM New York, NY, USA, 287–290.

Johan Bollen and Luis M. Rocha. 2000. An adaptive systems approach to the implementation and evaluation of digital library recommendation systems. In

Proceedings of the 4th European Conference on Digital Libraries. Springer, 356–359.

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*. Microsoft Research, 43–52.

Jonathan B. Buckheit and David L. Donoho. 1995. Wavelab and reproducible research. In *Wavelets and Statistics*. Lecture Notes in Statistics. Springer, 55–81.

Alvin C. Burns and Ronald F. Bush. 2013. *Marketing Research* 7th edition., Prentice Hall.

Arturo Casadevall and Ferric C. Fang. 2010. Reproducible science. *Infection and immunity* 78, 12 (2010), 4972–4975.

Richard Eckart de Castilho and Iryna Gurevych. 2011. A lightweight framework for reproducible parameter sweeping in information retrieval. In *Proceedings of the 2011 workshop on Data infrastructurEs for supporting information retrieval evaluation*. ACM, 7–10.

CiteULike. 2011. My Top Recommendations. *Webpage (http://www.citeulike.org/profile/joeran/recommendations)* (2011).

Open Science Collaboration and others. 2015. Estimating the reproducibility of psychological science. *Science* 349, 6251 (2015). DOI:http://dx.doi.org/10.1126/science.aac4716

Paolo Cremonesi, Franca Garzotto, Sara Negro, Alessandro Vittorio Papadopoulos, and Roberto Turrin. 2011. Looking for "good" recommendations: A comparative evaluation of recommender systems. In *Human-Computer Interaction–INTERACT 2011*. Springer, 152–168.

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

Martin Davies. 2011. Concept mapping, mind mapping and argument mapping: what are the differences and do they matter? *Higher education* 62, 3 (2011), 279–301.

Richard A. Deyo, Paula Diehr, and Donald L. Patrick. 1991. Reproducibility and responsiveness of health status measures statistics and strategies for evaluation. *Controlled clinical trials* 12, 4 (1991), S142–S158.

Renato Domingues Garcia, Matthias Bender, Mojisola Anjorin, Christoph Rensing, and Ralf Steinmetz. 2012. FReSET: an evaluation framework for folksonomy-based recommender systems. In *Proceedings of the 4th ACM RecSys workshop on Recommender systems and the social web*. ACM, 25–28.

R. Dong, L. Tokarchuk, and A. Ma. 2009. Digging Friendship: Paper Recommendation in Social Network. In *Proceedings of Networking & Electronic Commerce Research Conference (NAEC 2009)*. 21–28.

Steven M. Downing. 2004. Reliability: on the reproducibility of assessment data. *Medical education* 38, 9 (2004), 1006–1012.

Chris Drummond. 2009. Replicability is not reproducibility: nor is it good science. In *Proc. of the Evaluation Methods for MachineLearning Workshop at the 26th ICML.*

Michael D. Ekstrand, Praveen Kannan, James A. Stemper, John T. Butler, Joseph A. Konstan, and John T. Riedl. 2010. Automatically building research reading lists. In *Proceedings of the fourth ACM conference on Recommender systems*. ACM, 159–166.

Michael D. Ekstrand, Michael Ludwig, Jack Kolb, and John T. Riedl. 2011. LensKit: a modular recommender framework. In *Proceedings of the fifth ACM conference on Recommender systems*. ACM, 349–350.

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 fifth ACM conference on Recommender systems*. ACM, 133–140.

Alexander Felfernig, Michael Jeran, Gerald Ninaus, Florian Reinfrank, and Stefan Reiterer. 2013. Toward the next generation of recommender systems: applications and research challenges. In *Multimedia services in intelligent environments*. Springer, 81–98.

Bent Flyvbjerg. 2001. *Making social science matter: Why social inquiry fails and how it can succeed again*, Cambridge university press.

Zeno Gantner, Steffen Rendle, Christoph Freudenthaler, and Lars Schmidt-Thieme. 2011. MyMediaLite: A free recommender system library. In *Proceedings of the fifth ACM conference on Recommender systems*. ACM, 305–308.

Mouzhi Ge, Carla Delgado-Battenfeld, and Dietmar Jannach. 2010. Beyond accuracy: evaluating recommender systems by coverage and serendipity. In *Proceedings of the fourth ACM conference on Recommender systems*. ACM, 257–260.

Asela Gunawardana and Guy Shani. 2009. A survey of accuracy evaluation metrics of recommendation tasks. *The Journal of Machine Learning Research* 10 (2009), 2935–2962.

Guibing Guo, Jie Zhang, Zhu Sun, and Neil Yorke-Smith. 2015. Librec: A java library for recommender systems. In *Posters, Demos, Late-breaking Results and Workshop Proceedings of the 23rd International Conference on User Modeling, Adaptation and Personalization*.

Isabelle Guyon and André Elisseeff. 2003. An introduction to variable and feature selection. *The Journal of Machine Learning Research* 3 (2003), 1157–1182.

Michael Hahsler. 2011. recommenderlab: A Framework for Developing and Testing Recommendation Algorithms. *https://cran.r-project.org/web/packages/recommenderlab/vignettes/recommenderlab.pdf* (2011).

David Hawking, Nick Craswell, Paul Thistlewaite, and Donna Harman. 1999. Results and challenges in web search evaluation. *Computer Networks* 31, 11 (1999), 1321–1330.

Conor Hayes, P. Massa, P. Avensani, and Pádraig Cunningham. 2002. An on-line evaluation framework for recommender systems. In *Proceedings of the AH'2002 Workshop on Recommendation and Personalization in eCommerce*. Trinity College Dublin, Department of Computer Science.

Jing He, Jian-Yun Nie, Yang Lu, and Wayne Xin Zhao. 2012. Position-Aligned translation model for citation recommendation. In *Proceedings of the 19th international conference on String Processing and Information Retrieval*. Springer, 251–263.

Qi He, Jian Pei, Daniel Kifer, Prasenjit Mitra, and Lee Giles. 2010. Context-aware citation recommendation. In *Proceedings of the 19th international conference on World wide web*. ACM, 421–430.

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

William Hersh et al. 2000. Do batch and user evaluations give the same results? In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 17–24.

William R. Hersh et al. 2000. Further Analysis of Whether Batch and User Evaluations Give the Same Results with a Question-Answering Task. In *In Proceedings of the Ninth Text REtrieval Conference (TREC 9)*. 16–25.

Nancy Hoeymans, Emmy RCM Wouters, Edith JM Feskens, Geertrudis AM van den Bos, and Daan Kromhout. 1997. Reproducibility of performance-based and self-reported measures of functional status. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences* 52, 6 (1997), M363–M368.

Katja Hofmann, Anne Schuth, Alejandro Bellogin, and Maarten de Rijke. 2014. Effects of Position Bias on Click-Based Recommender Evaluation. In *Advances in Information Retrieval*. Springer, 624–630.

Brian Holland, Lynda Holland, and Jenny Davies. 2004. An investigation into the concept of mind mapping and the use of mind mapping software to support and improve student academic performance. *Centre for Learning and Teaching – Learning and Teaching Project Report. University of Wolverhampton* (2004), 89–94.

Wenyi Huang, Saurabh Kataria, Cornelia Caragea, Prasenjit Mitra, C. Lee Giles, and Lior Rokach. 2012. Recommending citations: translating papers into references. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 1910–1914.

Dietmar Jannach. 2014. Recommender systems: an introduction. *Lecture Slides (PhD School 2014)* (2014).

Dietmar Jannach, Lukas Lerche, Fatih Gedikli, and Geoffray Bonnin. 2013. What Recommenders Recommend–An Analysis of Accuracy, Popularity, and Sales Diversity Effects. In *User Modeling, Adaptation, and Personalization*. Springer, 25–37.

Dietmar Jannach, Markus Zanker, Mouzhi Ge, and Marian Gröning. 2012. Recommender Systems in Computer Science and Information Systems – A Landscape of Research. In *Proceedings of the 13th International Conference, EC-Web*. Springer, 76–87.

Bart P. Knijnenburg, Martijn C. Willemsen, Zeno Gantner, Hakan Soncu, and Chris Newell. 2012. Explaining the user experience of recommender systems. *User Modeling and User-Adapted Interaction* 22, 4-5 (2012), 441–504.

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 fifth ACM conference on Recommender systems*. ACM, 321–324.

Hylke Koers, Ann Gabriel, and Rebecca Capone. 2013. Executable papers in computer science go live on ScienceDirect. *https://www.elsevier.com/connect/executable-papers-in-computer-science-go-live-on-sciencedirect* (2013).

Joseph A. Konstan and Gediminas Adomavicius. 2013. Toward identification and adoption of best practices in algorithmic recommender systems research. In *Proceedings of the International Workshop on Reproducibility and Replication in Recommender Systems Evaluation*. ACM, 23–28.

Joseph A. Konstan and John Riedl. 2012. Recommender systems: from algorithms to user experience. *User Modeling and User-Adapted Interaction* (2012), 1–23.

Joseph Konstan and Michael D. Ekstrand. 2015. Introduction to Recommender Systems. *Coursera Lecture Slides* (2015).

Dominik Kowald, Emanuel Lacic, and Christoph Trattner. 2014. Tagrec: Towards a standardized tag recommender benchmarking framework. In *Proceedings of the 25th ACM conference on Hypertext and social media*. ACM, 305–307.

Stefan Langer and Joeran Beel. 2014. The Comparability of Recommender System Evaluations and Characteristics of Docear's Users. In *Proceedings of the Workshop on Recommender Systems Evaluation: Dimensions and Design (REDD) at the 2014 ACM Conference Series on Recommender Systems (RecSys)*. CEUR-WS, 1–6.

Andreas Lommatzsch. 2014a. Real-Time News Recommendation Using Context-Aware Ensembles. In *Proceedings of the 36th European Conference on Information Retrieval (ECIR)*. Springer, 51–62.

Andreas Lommatzsch. 2014b. Real-Time News Recommendation Using Context-Aware Ensembles. *PowerPoint Presentation, http://euklid.aot.tu-berlin.de/ andreas/20140414__ECIR/20140414__Lommatzsch-ECIR2014.pdf* (2014).

Y. Lu, J. He, D. Shan, and H. Yan. 2011. Recommending citations with translation model. In *Proceedings of the 20th ACM international conference on Information and knowledge management*. ACM, 2017–2020.

Nikos Manouselis and Katrien Verbert. 2013. Layered Evaluation of Multi-Criteria Collaborative Filtering for Scientific Paper Recommendation. In *Procedia Computer Science*. Elsevier, 1189–1197.

Azzah Al-Maskari, Mark Sanderson, and Paul Clough. 2007. The relationship between IR effectiveness measures and user satisfaction. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 773–774.

Sean M. McNee et al. 2002. On the Recommending of Citations for Research Papers. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work*. New Orleans, Louisiana, USA: ACM, 116–125. DOI:http://dx.doi.org/http://doi.acm.org/10.1145/587078.587096

Sean M. McNee, Nishikant Kapoor, and Joseph A. Konstan. 2006. Don't look stupid: avoiding pitfalls when recommending research papers. In *Proceedings of the 2006 20th anniversary conference on Computer supported cooperative work*. ACM, 171–180.

Marcia McNutt. 2014. Reproducibility. *Science* 343, 6168 (2014), 229–229.

Prem Melville, Raymond J. Mooney, and Ramadass Nagarajan. 2002. Content-boosted collaborative filtering for improved recommendations. In *Proceedings of the National Conference on Artificial Intelligence*. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 187–192.

David M. Pennock, Eric Horvitz, Steve Lawrence, and C. Lee Giles. 2000. Collaborative filtering by personality diagnosis: A hybrid memory-and model-based approach. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*. Morgan Kaufmann Publishers Inc., 473–480.

Karl Popper. 1959. *The Logic of Scientific Discovery*, London, United Kingdom: Hutchinson.

Pearl Pu, Li Chen, and Rong Hu. 2011. A user-centric evaluation framework for recommender systems. In *Proceedings of the fifth ACM conference on Recommender systems*. ACM, 157–164.

Pearl Pu, Li Chen, and Rong Hu. 2012. Evaluating recommender systems from the user's perspective: survey of the state of the art. *User Modeling and User-Adapted Interaction* (2012), 1–39.

Jalees Rehman. 2013. Cancer research in crisis: Are the drugs we count on based on bad science? *http://www.salon.com/2013/09/01/is_cancer_research_facing_a_crisis/* (2013).

Paul Resnick, Neophytos Iacovou, Mitesh Suchak, Peter Bergstrom, and John Riedl. 1994. GroupLens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*. ACM, 175–186.

Francesco Ricci, Lior Rokach, and Bracha Shapira. 2015. *Recommender Systems Handbook (2nd edt.)*, Springer.

Francesco Ricci, Lior Rokach, Bracha Shapira, and Kantor B. Paul. 2011. *Recommender systems handbook*, Springer.

Elaine Rich. 1979. User modeling via stereotypes. *Cognitive science* 3, 4 (1979), 329–354.

Peter M. Rothwell and Christopher N. Martyn. 2000. Reproducibility of peer review in clinical neuroscience. *Brain* 123, 9 (2000), 1964–1969.

Alan Said. 2013. Evaluating the Accuracy and Utility of Recommender Systems. *PhD Thesis. Technische Universität Berlin* (2013).

Alan Said and Alejandro Bellogin. 2014. Rival: a toolkit to foster reproducibility in recommender system evaluation. In *Proceedings of the 8th ACM Conference on Recommender systems*. ACM, 371–372.

Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar, and David M. Pennock. 2002. Methods and metrics for cold-start recommendations. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 253–260.

Stefan Schmidt. 2009. Shall we really do it again? The powerful concept of replication is neglected in the social sciences. *Review of General Psychology* 13, 2 (2009), 90.

Guy Shani and Asela Gunawardana. 2011. Evaluating recommendation systems. In *Recommender systems handbook*. Springer, 257–297.

Lalita Sharma and Anju Gera. 2013. A survey of recommendation system: Research challenges. *International Journal of Engineering Trends and Technology (IJETT)* 4, 5 (2013), 1989–1992.

Yue Shi, Martha Larson, and Alan Hanjalic. 2014. Collaborative Filtering Beyond the User-Item Matrix: A Survey of the State of the Art and Future Challenges. *ACM Comput. Surv.* 47, 1 (2014), 3:1–3:45. DOI:http://dx.doi.org/10.1145/2556270

Soren Sonnenburg et al. 2007. The need for open source software in machine learning. *Journal of Machine Learning Research* 8 (2007).

Dana Thomas, Amy Greenberg, and Pascal Calarco. 2011. Scholarly Usage Based Recommendations: Evaluating bX for a Consortium. *Presentation, http://igelu.org/wp-content/uploads/2011/09/bx_igelu_presentation_updated_september-13.pdf* (2011).

Roberto Torres, Sean M. McNee, Mara Abel, Joseph A. Konstan, and John Riedl. 2004. Enhancing digital libraries with TechLens+. In *Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*. ACM New York, NY, USA, 228–236.

Andrew H. Turpin and William Hersh. 2001. Why batch and user evaluations do not give the same results. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 225–231.

Ellen M. Voorhees. 2005. TREC: Improving information access through evaluation. *Bulletin of the American Society for Information Science and Technology* 32, 1 (2005), 16–21.

Fattane Zarrinkalam and Mohsen Kahani. 2013. SemCiR - A citation recommendation system based on a novel semantic distance measure. *Program: electronic library and information systems* 47, 1 (2013), 92–112.

Hua Zheng, Dong Wang, Qi Zhang, Hang Li, and Tinghao Yang. 2010. Do clicks measure recommendation relevancy?: an empirical user study. In *Proceedings of the fourth ACM conference on Recommender systems*. ACM, 249–252.

9 Author Biographies

(1) Dr. Joeran Beel:
Konstanz University, Department of Information Science, Universitätsstraße 10, D-78464 Konstanz, Germany

J. Beel is founder of the open-source reference-management software Docear, which features a research-paper recommender system. He received a Ph.D. in Computer Science from the Otto-von-Guericke University in Magdeburg, and M.S. in Project Management from the Lancaster University Management School. His current research interest lies in recommender systems and information retrieval with a focus on research-paper recommender systems and recommender-system evaluation. As visiting scholar at the University of California, Berkeley, he conducted research in the field of information extraction, web services, and academic search engines.

(2) Corinna Breitinger:
Linnaeus University, School of Computer Science, Physics and Mathematics, 351 95 Växjö, Sweden

C. Breitinger is a graduate student at Linnaeus University. She received her B.S. from the University of California Berkeley in 2011. Her interests lie in recommender systems, semantic similarity analysis and emerging web technologies. During a research stay at the National Institute of Informatics in Tokyo, she worked on advancing methods of semantic similarity identification for improving document recommendations. She participated in the research and development of the Docear open source project, where she collaborated with J. Beel and S. Langer on performing evaluations of the recommendation system.

(3) Stefan Langer:
Otto-von-Guericke University, Dept. of Computer Science, D-39106 Magdeburg, Germany

S. Langer is a researcher at Otto-von-Guericke University. His interests lie in the fields of recommender systems, search engines and the semantic analysis of written documents. In 2011 he co-founded the open source literature management software Docear together with J. Beel. During research stays at the University of California Berkeley and the University of Cyprus in Nikosia, he designed and implemented large parts of Docear's literature recommender system and search engine. Since February 2015 Langer is co-founder and CEO of the in4s IT company in Magdeburg, Germany.

(4) Dr. Andreas Lommatzsch:
Technische Universität Berlin, DAI-Lab, Ernst-Reuter-Platz 7, D-10587 Berlin, Germany

A. Lommatzsch received his PhD degree in Computer Science from the Berlin Institute of Technology. His research focuses on distributed knowledge management and information retrieval and systems. Since 2009 he has been a senior researcher and project coordinator in the domain of Recommender Systems and Machine Learning. His primary interests lie in the areas of recommendations based on data-streams and context-aware meta-recommender algorithms. He is one of the organizers of the CLEF NewsREEL challenge focusing on recommender algorithms for online news portals.

(5) Prof. Dr. Bela Gipp:
Konstanz University, Department of Information Science, Universitätsstraße 10, D-78464 Konstanz, Germany

B. Gipp is head of the Information Science Group (www.isg.uni.kn) at the University of Konstanz, Germany. Prior to this, he was a post-doctoral researcher at the University of California Berkeley and the National Institute of Informatics in Tokyo working in the area of information retrieval and visualization, knowledge management systems and web technologies. He co-founded Docear together with Joeran Beel while completing his doctoral research.

Additional Information

Bibliographic Data | J. Beel, C. Breitinger, S. Langer, A. Lommatzsch, and B. Gipp. “Towards Reproducibility in *Recommender-Systems Research.” User Modeling and User-Adapted Interaction (UMAI) 26*, no. 1 (2016): 69–101. DOI: http://dx.doi.org/10.1007/s11257-016-9174-x

Original Article | http://dx.doi.org/10.1007/s11257-016-9174-x

Preprint | https://www.gipp.com/wp-content/papercite-data/pdf/joeranbeeletal2016a.pdf

Related Work | http://docear.org/docear/research-activities/

Authors | Joeran Beel

Corinna Breitinger

Stefan Langer

Andreas Lommatzsch

Bela Gipp

BibTeX

```
@Article{JoeranBeelEtAl2016a,
Title                  = {{T}owards reproducibility in recommender-systems research},
Author                 = {{B}eel, {J}oeran
and {B}reitinger, {C}orinna
and {L}anger, {S}tefan
and {L}ommatzsch, {A}ndreas
and {G}ipp, {B}ela},
Journal                = {{U}ser {M}odeling and {U}ser-{A}dapted {I}nteraction ({UMUAI})},
Year                   = {2016},
Pages                  = {69-101},
Volume                 = {26},
Doi                    = {10.1007/s11257-016-9174-x},
ISSN                   = {1573-1391},
Url                    = {http://dx.doi.org/10.1007/s11257-016-9174-x}
}
```

RefMan (RIS)

```
TY  - JOUR
AU  - Beel, Joeran
AU  - Breitinger, Corinna
AU  - Langer, Stefan
AU  - Lommatzsch, Andreas
AU  - Gipp, Bela
M3  - http://doi.org/10.1007/s11257-016-9174-x
N1  - 1573-1391
PY  - 2016
SP  - 69-101
ST  - Towards reproducibility in recommender-systems research
T2  - User Modeling and User-Adapted Interaction (UMUAI)
TI  - Towards reproducibility in recommender-systems research
UR  - http://dx.doi.org/10.1007/s11257-016-9174-x
VL  - 26
ID  - 1
ER  -
```

EndNote

```
%0 Journal Article
%A Beel, Joeran
%A Breitinger, Corinna
%A Langer, Stefan
%A Lommatzsch, Andreas
%A Gipp, Bela
%D 2016
%T Towards reproducibility in recommender-systems research
%B User Modeling and User-Adapted Interaction (UMUAI)
%V 26
%P 69-101
%9 http://doi.org/10.1007/s11257-016-9174-x
%Z 1573-1391
%U http://dx.doi.org/10.1007/s11257-016-9174-x
```