A Method for Evaluating the Navigability of Recommendation Algorithms

Daniel Lamprecht, Markus Strohmaier, and Denis Helic

Abstract Recommendations are increasingly used to support and enable discovery, browsing and exploration of large item collections, especially when no clear classification of items exists. Yet, the suitability of a recommendation algorithm to support these use cases cannot be comprehensively evaluated by any evaluation measures proposed so far. In this paper, we propose a method to expand the repertoire of existing recommendation evaluation techniques with a method to evaluate the navigability of recommendation algorithms. The proposed method combines approaches from network science and information retrieval and evaluates navigability by simulating three different models of information seeking scenarios and measuring the success rates. We show the feasibility of our method by applying it to four non-personalized recommendation algorithms on three datasets and also illustrate its applicability to personalized algorithms. Our work expands the arsenal of evaluation techniques for recommendation algorithms, extends from a one-click-based evaluation towards multi-click analysis and presents a general, comprehensive method to evaluating navigability of arbitrary recommendation algorithms.

1 Introduction

Websites with large collections of items need to support three ways of information retrieval: (i) retrieval of familiar items (ii) retrieval of items that cannot be explicitly described but will be recognized once retrieved and (iii) serendipitous discovery [32]. For a website with a large collection of items, such as an e-commerce website, (i) can be enabled with a full-text search function. For (ii) and (iii), however, a search

Daniel Lamprecht and Denis Helic
KTI, Graz University of Technology, e-mail: daniel.lamprecht@tugraz.at,dehlic@tugraz.at

Markus Strohmaier
GESIS and University of Koblenz-Landau, e-mail: strohmaier@uni-koblenz.de

function is generally not sufficient. These types of information retrieval are therefore often supported by recommendations that connect items and enable discovery and navigation.

Users have been found to enjoy perusing item collections such as e-commerce sites or recommender systems without the immediate intention of making a purchase [15]. More generally, some users prefer navigation to direct search even when they know the target [31]. For platforms where users immediately consume content, such as YouTube or Quora, recommendations serve the use case of *unarticulated want*, and are therefore a crucial part of the user experience [10]. In item collections that do not associate descriptions or metadata with content (such as videos) frequently no clear structuring of items exists, and recommendations play a vital role in the user interfaces. It is therefore critical for these systems to support discovery via links.

When a website provides recommendations along with each item, the items and the associated recommendations form a *recommendation network*—an implicit view of a recommender system where items are nodes and recommendations are edges. This type of recommendations are frequent on e-commerce websites, such as Amazon ("customers who bought this also bought"). Many websites associate a fixed number of recommendations with each item, which leads to a constant outdegree and a varying indegree for each node in the network

Knowing more about recommendation networks would give web-site operators the possibility to assess the effects of recommendations and help to produce recommendations that make it easier for users to discover and explore items. While a few studies have already looked at recommendation networks and provided first important insights into the nature and structure of these networks [6, 8, 20, 29], there is no systematic approach to evaluating the navigability of recommendation algorithms.

This paper presents a general method to evaluate the practical navigability of arbitrary recommendation networks by using simulations based on three navigation models established in the literature, namely *point-to-point navigation* [16], *navigation via berrypicking* [2] and *navigation via information foraging* [28]. The combination of established techniques from the fields of network science and information retrieval allows us to present a novel method that extends common evaluation measures towards a path-based evaluation and expands the arsenal of existing recommendation evaluation techniques.

We show the feasibility of this method by applying it to four non-personalized recommendation algorithms on three datasets and investigate their properties. We also illustrate the general suitability of our method to personalized recommendations and report initial results for a sample configuration.

2 Related Work

Initially, recommender systems were mostly evaluated in terms of prediction accuracy [12]. However, the focus on accuracy has been found to neglect other import applications of recommender systems such as support for the discovery of novel items, browsing, or diversified recommendations, and may lead to a bias towards popular items [8, 30] or a filter bubble effect [25]. For these reasons, a series of evaluation metrics for additional properties of recommender systems has been developed. These metrics include diversity [4, 7], novelty [7, 12], serendipity and coverage [11, 12, 15] and are considered orthogonal to prediction accuracy.

The evaluation method presented in this paper is rooted in Stanley Milgram's small world experiments [24], which laid the foundation for *decentralized search*. Kleinberg [17] and Watts [34] later formalized the property that a navigable network requires short paths between all (or almost all) nodes. Kleinberg also found that an *efficiently navigable* network possesses certain structural properties that make it possible to design efficient decentralized search algorithms that only have local knowledge of the network [16]. The delivery time of such algorithms is then sublinear in the number of network nodes. In this paper, we investigate the efficient navigability of recommendation networks through the simulation of navigation models based on decentralized search.

The static topology of recommendation networks has been extensively studied for the case of music recommenders [8, 29]. Their corresponding recommendation networks have been found to exhibit heavy-tail degree distributions and small-world properties [6], implying that they are efficiently navigable with local search algorithms. A first study [20] has already explored the reachability and navigability of the recommender systems of IMDb. The corresponding recommendation networks were shown to lack support for navigation scenarios. However, the use of diversified recommendations was able to substantially improve this and lead to more navigable recommendation networks. A similar methodology has been applied to suggest links to improve navigability on Wikipedia [19].

3 Evaluation Method

Navigation is at the core of exploration and browsing, which are important use cases of a recommender system, as many users find browsing pleasant [15], use it to discover novel content [22] or consume the content along the browsing path (e.g., on YouTube). A defining property of online navigation is that the knowledge about a website is mostly local: users only perceive the links emanating from the current page and generally only have intuitions about where those links might lead, but lack global knowledge about the system. In the case of a top-N recommender system, users are generally only aware of the recommendations provided with the current item.

The evaluation method we propose makes use of greedy decentralized search to simulate navigation in recommender systems and measures the success rate. This model has been used in previous work to analyze navigation dynamics in networks [13, 14] and has been found to produce comparable results to human navigation patterns [21, 33]. At each step, this algorithm evaluates a heuristic for every present link and greedily selects the one maximizing that heuristic. We take the heuristic to

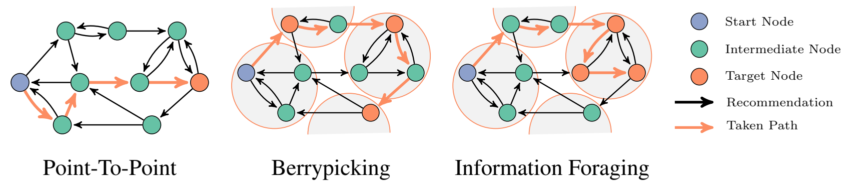

Fig. 1: **Information Seeking Scenarios.** We use three information seeking scenarios to study navigability of recommendation networks. The objective in point-to-point navigation is to find a single goal item. For berrypicking, we cluster the networks and set the goal of finding any one item in four clusters (shown in gray). For information foraging, the goal is to find multiple items in a single cluster.

represent vague intuitions about navigation that users might gain from looking at the descriptions of recommendation targets. For example, if a user was looking for a new science-fiction movie, they might be tempted to follow recommendations to other science fiction movies based on the title, a brief textual description or the displayed image. We use an implementation that does not revisit previously explored nodes. In case no unvisited item is present, the simulation backtracks.

A number of information seeking models have been established in the literature. To investigate the general suitability of recommendation algorithms to navigation based on different approaches, we evaluate navigation scenarios based on three of these models: point-to-point navigation [16], berrypicking [2], and information foraging [28]. For all scenarios, the start and target nodes in the network are determined independently of the network structure, i.e., regardless of whether the recommendation algorithm actually enabled a path between them. This allows us to fairly compare all recommendation algorithms and shows how well they support navigability. In what follows, we describe the three navigation scenarios in more detail (cf. Figure 1).

Point-To-Point Navigation. Point-to-point navigation [16] represents the task of finding a single target item in a recommendation network and models the navigational behavior of users with a specific item in mind that they cannot explicitly describe. For example, a user could try to find a science-fiction movie with a specific motif or to rediscover something on tip of their tongue. As such, this scenario covers point (ii) ("retrieval of items that cannot be explicitly described") of Toms's ways of information retrieval [32]. We then simulate navigation starting at the start node of a pair and with the objective of reaching the target node. As start-target pairs we sample pairs of nodes proportionally to how often they were corated by users in the corresponding rating dataset.

Navigation via Berrypicking. Berrypicking is an information seeking model which regards information seeking as a dynamic process where the information need is evolving and can be satisfied by multiple pieces of information in a *bit-at-a-time retrieval*—an analogy to picking berries on bushes [2]. Berrypicking can be thought of as covering points (ii) ("retrieval of items that cannot be explicitly described") and (iii) ("serendipitous discovery") of Toms's ways of information retrieval [32].

We model this scenario based on clusters, which we obtain with k-means based on the rating vectors. We randomly pick a first cluster and then draw one of the top four closest clusters based on Euclidian distance randomly. We then repeat this to find two more clusters. Starting from a randomly chosen node in the first cluster, the objective of the scenario is then to reach any node from the second cluster, followed by any node from the third and then the fourth cluster. In this way, the scenario models the evolving stages of berrypicking, where users inspect an item and adapt their information needs based on it.

Navigation via Information Foraging. Information foraging [28] is an information seeking theory inspired by optimal foraging theory in nature, where organisms have adopted strategies maximizing energy intake. For instance, when foraging on a patch of food, an animal must decide when to move on to the next patch (e.g., when finding apples on a tree is becoming too tedious). Some of the same mechanisms have identified for human information seeking behavior, where humans try to maximize information gain. Information can be modeled as occurring in patches, and information seekers as guided by *information scent* [9]. In a scenario based on information foraging, we model the scenario of depleting a patch of information. We assume that the objective is to retrieve nodes in a patch—guided by information scent in terms of the search heuristic. We take information foraging to model points (ii) and (iii) ("retrieval of items that cannot be explicitly described") and "serendipitous discovery") of Toms's ways of information retrieval [32].

Baselines. We also evaluate two baseline solutions: An *optimal solution* makes use of the shortest possible paths for a scenario (that users with perfect knowledge of the network could take). A *random solution* performs a random walk with no background knowledge at all.

4 Experimental Setup

We use three datasets for this paper:

- **MovieLens** is a film recommender systems maintained by GroupLens Research at the University of Minnesota. For this work, we use their dataset consisting of one million ratings from $6,000$ users on $4,000$ movies.
- **BookCrossing** is a book exchange platform. For this work, we use a 2005 crawl of the website [35]. We use only the explicit ratings, combine ratings for duplicate books and use ratings from users with ≥ 20 ratings on ≥ 5 books. This leaves us with roughly $50,000$ ratings by $1,088$ users on $3,637$ books.
- **IMDb** is a database of movies and TV shows. We use a 2015 crawl of the website [20], from which we use ratings for items published in 2013 and 2014 and condense them in the same way as for the BookCrossing dataset, resulting in 2.3M ratings for $6,690$ titles by $37,216$ users.

We calculate recommendations in the following way: For a given set of items I and a recommendation algorithm R, we use R to compute the pairwise similarities for all pairs of items $(i, j) \in I$. For each item $i \in I$, we then define the set of the top-N most similar items to i as $L_{i,N}$. We then create a directed top-N recommendation network $G(V,N,E)$, where $V = I$, N is the number of recommendations available for each item and $E = \{(i, j) \,|\, i \in I, j \in L_{i,N}\}$. This method leads to recommendation networks with constant outdegree and varying indegree—representing a typical setting.

For simplicity's sake, we investigate recommendation algorithms based on non-personalized recommendations. The similarities these recommendations are based on, however, are directly taken from the similarities used in the personalized recommendation algorithms. They therefore represent the recommendation networks as an unregistered or newly registered user would see them. For most websites, the vast majority of visitors does not contribute or register—this is known as the *90-9-1 Rule* (90% lurkers, 9% intermittent contributers and 1% heavy contributers) [26, 27]. However, our method is general and also applicable to personalized recommendation algorithms, which we exemplarily demonstrate in Section 6.

We use the following four recommendation algorithms in this work:

Association Rules (AR). Association rules are based on the market-basket model, where, in this case, we put all items rated by the same user into a basket and regard ratings as binary (i.e., rated/not rated). For every ordered pair of items (i, j), we then rank all items by how much more likely an item is to be consumed after a given item was consumed (similiar to the Apriori algorithm [1]). Specifically, we compute the fraction of co-ratings of i and j over the total ratings of i (i.e., the fraction users who rated both i and j, out of those who rated i). Let U_i be the set of users who rated item i. We can then compute this as as $(|U_i \cap U_j|)/(|U_i|)$. To compensate for the popularity of j, we then divide by the fraction of users who did not rate i but still rated j. Let $\overline{U}_i$ be the set of users who did not rate item i. We can then divide by $(|\overline{U}_i \cap U_j|)/(|\overline{U}_i|)$ to counter the effect of highly popular items that are likely to be co-rated with every item, but would not be very useful as a recommendation. We then take the top-N items most likely to be co-rated with it.

Collaborative Filtering (CF). For a given user u and an unrated item i, item-based collaborative filtering predicts the rating of u for i from a small number of other items that u previously rated. These other items are commonly selected as the ones maximizing the centered cosine similarity to i. The rating prediction is then computed as the weighted sum of their ratings, weighted by their similarity. To obtain unpersonalized recommendations, we compute the centered cosine similarity of an item i to all other items j in the dataset and use the top-N.

Interpolation Weights (IW). Interpolation weights are computed in a similar way to item-based collaborative filtering. However, instead of using a predefined similarity measure (such as the centered cosine similarity) to weight the contributions of other ratings, *interpolation weights* representing the relations between pairs of items are learned from the data. We use gradient descent to learn item-based interpolation weights by minimizing the root-mean square error for predictions on a test set [3] and then use the resulting weights as the similarity measure to obtain the top-N most similar items to an item.

Matrix Factorization (MF). Matrix factorization describes both items and users of a recommender system by affinities to a number of latent factors [18]. To find these factors, this algorithm factorizes the rating matrix U into two matrices as $U = Q^T P$ that represent the associations of users and items with the latent factors. We learn these matrices by minimizing the root-mean-square prediction error on a test set with gradient descent. After this minimization, we represent each item by the vector of its association with the latent factors and compute the centered cosine similarity between the latent factors for all pairs of items to obtain the top-N most similar items.

As the heuristic for decentralized search, we use the TF-IDF cosine similarity of brief textual descriptions of titles (namely title and plot summary of IMDb for the movies and the summary provided by GoodReads for the books). At each step, the simulation uses this heuristic to select the link leading to the item that has the highest TF-IDF cosine similarity to the navigation goal. We use a heuristic independent of ratings to decouple it from the recommendations used to generate the networks. For sake of brevity, we only report the results for a deterministic greedy search with 50 steps. However, we also evaluated all simulations for 10 and 25 steps as well as with an ε-greedy approach [13] and found that, while the total success rates decreased, the relative differences between the approaches did not change.

We evaluate a total of $1,200$ navigation simulations per scenario. For the clusters, we only use those consisting of 4–30 nodes to balance the difficulty. As a difference to the point-to-point navigation scenario, the target of the navigation for the berrypicking and information foraging scenario is not represented by a single node but by the centroid of the target cluster. The TF-IDF cosine similarity of a potential link target l is therefore represented by the average of the similarity between l and all items in the target cluster.

5 Results

Point-To-Point Navigation. The first row of Figure 2 displays the success rate (i.e., the fraction of successful simulations) for point-to-point navigation. Since the number of steps per simulation (50) is larger than the distances between all start-target pairs in the recommendation networks, the optimal solutions (shown in gray bars) correspond to all start-target pairs between which a path of any length existed. The optimal solution is therefore a measure of how well a recommendation algorithm theoretically supports this navigation scenario. The second baseline approach is a random walk, which shows the success rates achievable by an uninformed random process and serves to demonstrate that the simulations based on greedy search are able to exploit the link selection heuristic to reach navigation goals. The simulation for point-to-point navigation with greedy search for $N = 5$ recommendations leads to an average success rate of 6.86%. This indicates that users would be able to retrieve only a very small share of items in the recommender systems by focused point-to-point navigation. For $N = 20$ recommendations, the success rates increase substantially (average of 24.4%). Recommendations generated by interpolation weights lead to the best success rates (42–48%).

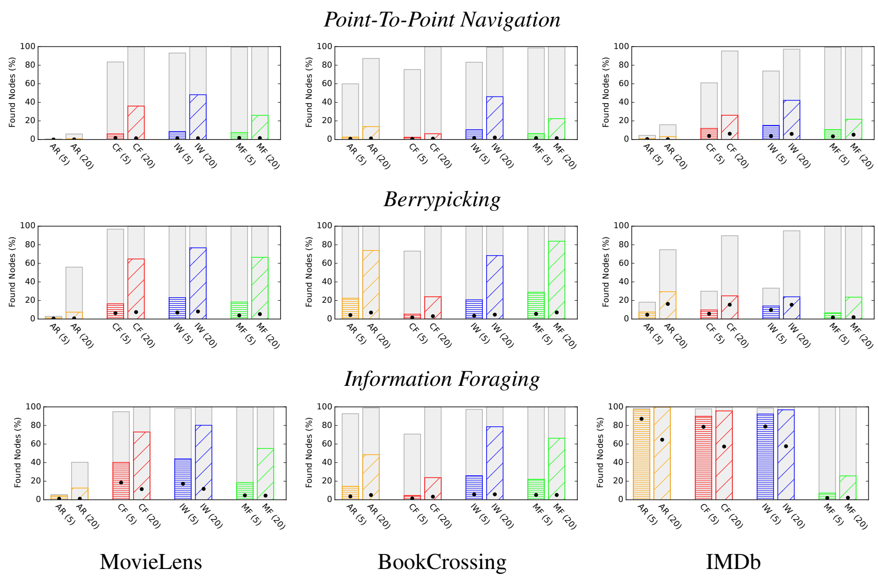

Fig. 2: **Success Ratios for the navigation simulations.** The bars depict the average percentage of found targets. Baseline success rates are depicted as gray bars (optimal solutions) and black dots (random walk solutions). Success rates are computed as the average number of found targets. Recommendation networks generated by interpolation weights (IW) generally performed best.

Navigation via Berrypicking. For five recommendations, the success rates for the case of genre-based clusters are 14.5% on average. With 20 recommendations, this increases to to 47%. Since the targets consists of three clusters, a success rate of 33% indicates that an average of one cluster was found.

The success rates for the IMDb dataset are substantially lower than for the other two datasets. A more detailed analysis shows that the networks for IMDb are clustered more strongly than those of the other two datasets. For a dynamic information seeking scenario such as berrypicking, this means that the simulation of adapting information needs was not very well supported for IMDb. Overall, reommendations generated by matrix factorization and interpolation weights fared best.

Navigation via Information Foraging. A priori, it is not clear if retrieving multiple items from the same cluster represents an easier task than retrieving them from different clusters, as a cluster of items does not necessarily mean that items are located in proximity in the recommendation network. However, the resulting success rates show that items from the same clusters in the network are easier to retrieve: five recommendations lead to a success rate of 38.3%, and twenty recommendations to 63.1%. This indicates that the recommendation algorithms are able to use the characteristics in the ratings to support both genre-based and rating-based clustering.

The success rates again measures the number of found items in a cluster. The results for this scenario show that the success rates for the baselines, namely the random walks and the optimal solutions are consistently very high. This also indicates that the network structures reflect the clustering very well. Whereas for berrypicking, the simulations on the IMDb dataset perform poorly, the contrary is the case for information foraging, where the success rates range up to 99%. This again confirms the strong clustering in these networks, that lead to densely interconnected regions among similar items and facilitate retrieval of items in the same cluster. Recommendations generated by the interpolation weights algorithm generally fare best.

6 Personalized Recommendations

We now demonstrate the general suitability of our method to personalized recommendation approaches and report initial results for a sample configuration of parameters. The key difference for personalized recommendations is that a separate recommendation network emerges for every user based on their rated items. For this illustration, we follow the approach of Amazon.com, as detailed by Linden, Smith and York in 2003 [23], which consists of two steps: First, a set of similar items is determined for each item. Second, the items with the highest predicted rating among this set are recommended. We study two variants of this:

- **Pure.** We first compute a candidate set of similar items for an item—these are simply the non-personalized recommendations. Then we select the N items from this set that have the highest predicted rating for the specific user.
- **Mixed.** We again compute the set of similar items, but only use the $N/2$ recommendations with the highest predictions and add the $N/2$ top non-personalized recommendations (without introducing duplicates).

For both algorithms, we allow the recommendation of items that the user had already rated (which is yet another parameter to tune). We note that for this setting, the differences between the personalized networks for users decrease. When not allowing this, the resulting recommendation networks show a decrease in navigability the more items a user has already rated. For sake of space, we only report results for a restricted set of parameters. The results for the other combinations of parameters were similar, but we leave it to future work to examine them in more details.

Figure 3 shows the evaluation for recommendations generated by interpolation weights and matrix factorization for the user with the median number of ratings in the BookCrossing dataset. The outcome is generally similar to non-personalized networks. The pure algorithm leads to notably higher success rates for the optimal solution, but not for the simulation results themselves. This indicates that while the mixed algorithm leads to a better connectivity in the networks, this was not necessarily the case for navigability. This in turn suggests that the recommendations generated by this algorithm did not capture the intuitions used in the navigation

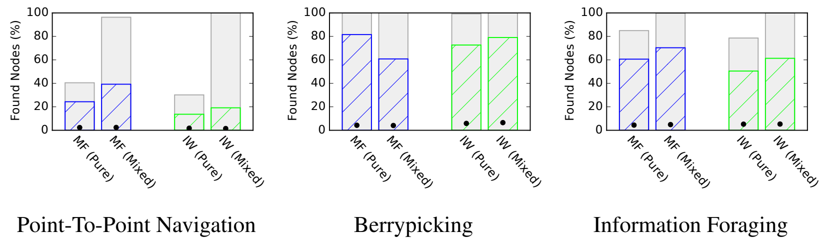

Fig. 3: **Navigational Success Rates for Personalized Recommendations.** All simulations were evaluated for BookCrossing, 20 recommendations and personalized for the user with the median number of ratings in the dataset. The results show that while the mixed recommendations enable a better optimal solution, the recommendations did not reflect the intuitions of the navigation simulations very well.

simulations very well. In future work, the evaluation method proposed in this paper could be used to develop a more effective personalized recommendation selections.

7 Discussion

We have presented a novel evaluation method that expands the repertoire of recommendation evaluation measures with a technique to assess navigability. The proposed method evaluates the navigation dynamics of recommendation networks by simulating three different navigation models, namely point-to-point navigation, navigation via berrypicking and navigation via information foraging. We believe that applying this method can broaden our understanding of recommendation algorithms and lead to a more complete characterization of their properties.

To demonstrate the feasibility of our method, we applied it to three exemplary datasets and highlighted differences in navigability for four different, non-personalized, recommendation algorithms. For five recommendations per item, we find that the recommendation algorithms we investigate considerably limit the navigability. However, we find that it can be improved by raising the number of recommendations. For the three navigation scenarios we investigate we find that the explorative scenarios inspired by berrypicking and information foraging lead to the best retrieval performance, while the scenario based on point-to-point navigation was less well supported. While increasing the number of recommendations represents a simple solution, a large number of recommendations could potentially clutter the interface and overwhelm users [5]. This shows that there is still a substantial potential to improve recommendation algorithms to better support navigation dynamics. As for the recommendation algorithms, we find that the recommendations generated by interpolation weights and matrix factorization performed best overall. However, more work is necessary to confirm these findings.

The selection of algorithms and datasets was naturally arbitrary, but they serve the purpose of illustrating the evaluation and therefore do not limit our main contribution of presenting a novel evaluation method. We have shown the suitability of our method for non-personalized recommendation algorithms and thereby effectively inspected recommendation networks for users who are either new to the system or simply browsing without being registered, and have also illustrated the applicability of our method to personalized recommendations.

The navigation models applied in this method are well-established in the research community and cover a wide range of typical user interaction scenarios with information systems in general, and recommender systems in particular. Greedy decentralized search, the basis for our navigation scenarios based on these models, has been used in previous work to analyze navigation dynamics in networks [13, 14] and has been found to produce comparable results to human navigation patterns [21, 33]. The navigation models we used do, however, have limitations and were deliberately kept simple, as the focus of our work was not on the information seeking models and their validity but on the properties of the recommendation algorithms. However, this does not limit our work, as our evaluation method does not depend on this particular model, which can easily be adapted or exchanged in future work. Possible enhancements to the navigation models could include a teleportation element (as in PageRank) modeling jumps between items without recommendations.

In summary, our work extends common evaluation measures of recommendation algorithms towards a path-based evaluation. Just as the evaluation of recommender systems has been shifting from accuracy-based measures towards diversification, coverage and time-dependent evaluations, we believe that our method helps to push the frontier of recommendation algorithms towards producing recommendations that make it easier for users to discover and explore items.

References

1. R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In *VLDB'94*, 1994.
2. M. J. Bates. The design of browsing and berrypicking techniques for the online search interface. *Online Information Review*, 13(5):407–424, 1989.
3. R. M. Bell and Y. Koren. Improved neighborhood-based collaborative filtering. In *KDD Cup and Workshop at the KDD'07*, 2007.
4. R. Boim, T. Milo, and S. Novgorodov. Diversification and refinement in collaborative filtering recommender. In *CIKM'11*, 2011.
5. D. Bollen, B. P. Knijnenburg, M. C. Willemsen, and M. Graus. Understanding choice overload in recommender systems. In *RecSys'04*. ACM, 2010.
6. P. Cano, O. Celma, M. Koppenberger, and J. M. Buldú. Topology of music recommendation networks. *Chaos*, 16(1), 2006.
7. P. Castells, N. J. Hurley, and S. Vargas. Novelty and diversity in recommender systems. In *Recommender Systems Handbook*, pages 881–918. Springer, 2015.
8. Ò. Celma and P. Herrera. A new approach to evaluating novel recommendations. In *RecSys'08*, 2008.
9. E. H. Chi, P. Pirolli, K. Chen, and J. Pitkow. Using information scent to model user information needs and actions on the web. In *CHI'01*, 2001.

10. J. Davidson, B. Liebald, J. Liu, P. Nandy, T. V. Vleet, U. Gargi, S. Gupta, Y. He, M. Lambert, B. Livingston, and D. Sampath. The youtube video recommendation system. In *RecSys'10*, 2010.
11. M. Ge, C. Delgado-Battenfeld, and D. Jannach. Beyond accuracy: Evaluating recommender systems by coverage and serendipity. In *Proc. of the 4th ACM Conference on Recommender Systems*, 2010.
12. A. Gunawardana and G. Shani. Evaluating recommender systems. In *Recommender Systems Handbook*, pages 265–308. Springer, 2015.
13. D. Helic, M. Strohmaier, M. Granitzer, and R. Scherer. Models of human navigation in information networks based on decentralized search. In *HT'13*, 2013.
14. D. Helic, M. Strohmaier, C. Trattner, M. Muhr, and K. Lerman. Pragmatic evaluation of folksonomies. In *WWW'11*, 2011.
15. J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, 22(1):5–53, 2004.
16. J. M. Kleinberg. Navigation in a small world. *Nature*, 406(6798):845, August 2000.
17. J. M. Kleinberg. The small-world phenomenon: An algorithmic perspective. In *TOC'00*, 2000.
18. Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.
19. D. Lamprecht, D. Dimitrov, D. Helic, and M. Strohmaier. Evaluating and improving navigability of wikipedia: A comparative study of eight language editions. In *OpenSym'16*, 2016.
20. D. Lamprecht, F. Geigl, T. Karas, S. Walk, D. Helic, and M. Strohmaier. Improving recommender system navigability through diversification: A case study of IMDb. In *IKNOW'15*, 2015.
21. D. Lamprecht, M. Strohmaier, D. Helic, C. Nyulas, T. Tudorache, N. F. Noy, and M. A. Musen. Using ontologies to model human navigation behavior in information networks: A study based on wikipedia. *Semantic Web*, 6(4):403–422, 2015.
22. K. Lerman and L. Jones. Social Browsing on Flickr. In *ICWSM'07*, 2007.
23. G. Linden, B. Smith, and J. York. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Computing*, 7(1):76–80, 2003.
24. S. Milgram. The small world problem. *Psychology Today*, 1(2):60–67, 1967.
25. T. T. Nguyen, P.-M. Hui, M. F. Harper, L. Terveen, and J. A. Konstan. Exploring the filter bubble: the effect of using recommender systems on content diversity. In *WWW'14*, 2014.
26. J. Nielsen. The 90-9-1 rule for participation inequality in social media and online communities, 2006. `www.nngroup.com/articles/participation-inequality`.
27. B. Nonnecke and J. Preece. Lurker demographics: Counting the silent. In *CHI'00*, 2000.
28. P. Pirolli. *Information Foraging Theory: Adaptive Interaction with Information*. Oxford University Press, 2007.
29. K. Seyerlehner, P. Knees, D. Schnitzer, and G. Widmer. Browsing music recommendation networks. In *ISMIR'09*, 2009.
30. J. Su, A. Sharma, and S. Goel. The effect of recommendations on network structure. In *WWW'16*, 2016.
31. J. Teevan, C. Alvarado, M. S. Ackerman, and D. R. Karger. The perfect search engine is not enough: A study of orienteering behavior in directed search. In *CHI'04*, 2004.
32. E. G. Toms. Serendipitous information retrieval. In *DELOS Workshop: Information Seeking, Searching and Querying in Digital Libraries*, 2000.
33. C. Trattner, P. Singer, D. Helic, and M. Strohmaier. Exploring the differences and similarities between hierarchical decentralized search and human navigation in information networks. In *IKNOW'12*, 2012.
34. D. J. Watts, P. S. Dodds, and M. Newman. Identity and search in social networks. *Science*, 296:1302–1305, 2002.
35. C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. Improving recommendation lists through topic diversification. In *WWW'05*, 2005.