Towards Multi-Stakeholder Utility Evaluation of Recommender Systems

Robin Burke
DePaul University
Chicago, USA
rburke@cdm.depaul.edu

Himan Abdollahpouri
DePaul University
Chicago, USA
habdolla@cdm.depaul.edu

Bamshad Mobasher
DePaul University
Chicago, USA
mobasher@cdm.depaul.edu

Trinadh Gupta
DePaul University
Chicago, USA
kumar.ivin@gmail.com

ABSTRACT

A core value in recommender systems is personalization, the idea that the recommendations produced are those that match the user's preferences. However, in many real-world recommendation contexts, the concerns of additional stakeholders may come into play, such as the producers of items or those of the system owner. Some researchers have examined special cases of such concerns, for example, in reciprocal recommendation. However, there has not been a comprehensive treatment of the integration of multiple stakeholders into recommendation calculations. The paper suggests a utility-based framework for representing stakeholder values in recommendation actions and calculating a multi-dimensional utility. We demonstrate how a standard algorithm performs in a simulation of a multi-stakeholder recommendation task requiring on-line optimization, and show that a simple greedy approach can lead to enhanced overall utility with minimal loss of accuracy for users.

CCS Concepts

•Information systems → Recommender systems; Retrieval effectiveness; •Computing methodologies → *Online learning settings;*

Keywords

Recommender systems; Utility-theoretic approaches; Recommendation Evaluation; Coverage

1. INTRODUCTION

One of the key characteristics of recommender systems research is an emphasis on personalization. Recommender systems are typically evaluated on their ability to provide items that satisfy the needs and interests of the end user. Researchers have also examined additional metrics (such as diversity and novelty) that can be used to measure other aspects of the suitability of recommendation results to the target user, but it remains true that the end user as the receiver of recommenders is, for the most part, the only consideration.

:

Such focus on the end user is entirely appropriate. Users would not flock to recommender systems if they believed such systems were not providing items that matched their interests. Still, it is also clear that, in many recommendation domains, the end user for whom recommendations are generated is not the only stakeholder in the recommendation outcome. Several pertinent examples can be given. Reciprocal recommendation is the term applied to a situation in which a recommendation must be acceptable to both parties in a transaction. For example, in on-line dating, the value of a recommendation of partner Y for user X may be a function of both the acceptability of Y to X, and the acceptability of X to Y. Both parties must be interested in order for a match to be successful [15].

The multi-stakeholder dynamic is also at work in on-line advertising. The retrieval of a display ad in a real-time display advertising context depends not just on whether the ad is of interest to the user but, because advertisers pay for each impression, it also matters if the user is of interest to the advertiser [22]. An ad campaign may have a specific target audience in mind and an acceptable ad retrieval is one where the ad matches user interest and browsing context and where the user is a member of the desired demographic. For example, teenage boys may be very interested in exotic Italian sports cars, but they are typically not purchasers of them. An advertiser seeking potential buyers would probably not want to spend scarce dollars on such users.

The on-line advertising setting has the additional constraint in that a budget is involved. If the ad budget for a given timeframe (typically a day or a month) is exhausted, no additional ads can be delivered even if a highly desirable customer arrives. There is therefore an opportunity cost associated with each ad placement, which is also a factor in its desirability. [8]

We believe that, far from being special "edge cases" of recommendation, such examples illustrate a more general point about recommendation, namely, that recommender systems serve multiple goals and that a purely user-centered approach does not allow all such goals to enter into the design and evaluation of recommendation algorithms where appropriate. We believe that our view of recommender system evaluation should be broadened to include the perspectives and utilities of multiple stakeholders.

A utility-oriented approach also has the advantage of allowing the evaluation of recommender system performance to be based on a range of considerations, of which similarity to prior ratings may be only one factor. Concerns about the "filter bubble" that can be generated by a strict focus on users' prior ratings [13] can be directly addressed by incorporating additional types of utility into the recommendation algorithm. A news site, for example, might formulate utility in terms of citizens' need to see information about important issues, in addition to their previously-expressed preferences. A learning system may incorporate pedagogical aims into its recommendations of educational resources, aiming, for example, to confront the student with a variety of viewpoints.

In this paper, we derive a general formulation of the multi-stakeholder utility question and examine the implications of this formulation in a specific recommendation problem. We examine an on-line recommendation setting and demonstrate that a simple greedy optimization approach can yield considerable utility gains. This is a simple demonstration of the multi-stakeholder concept and its applicability. We intend to explore its potential further in future work.

2. PROBLEM FORMULATION

We account for multiple stakeholders by performing explicit, if rough, estimation of the utilities of each party involved in a recommendation transaction. For the purposes of this paper, we assume an environment in which a user will periodically seek lists of recommendations from a system. There is utility $u_v(i)$ for the user v associated with the inclusion of item i in such a list. Note that there may, of course, be differential utilities based on the rank of an item in the list. We leave the consideration of the interaction of rank and utility for future work. The utility of a given recommendation list to the user is therefore the sum of these utilities $u_v(L) = \sum_{i \in L} u_v(i)$, where L is the set of items retrieved. If the user visits the system multiple times, additional utility is gained from each list. This is, of course, also a simplification since there may or may not be utility in receiving the same recommendation more than once, depending on the application and context.

We also are assuming a setting where each item has a supplier or owner d, and there are many such suppliers. Each supplier owns a set of items ω and gains utility from the presence of an item $i \in \omega$ in the recommendation list. The utility may be a function of how well matched the item is for the receiver of the list, and is therefore a function of v: $u_d(i, v)$. Therefore, the utility for an owner d of a given recommendation list is $u_d(L, v) = \sum_{i \in L} u_d(i, v)\chi_d(i)$, where $\chi_d(i)$ is the indicator function that returns 1 if item i is owned by owner d and 0 otherwise. Each owner is a separate stakeholder just like each user.

In addition, there is another party whose interests must be represented and that is the system owner s. The system owner may have a complex utility calculation related to recommendation presentation. For example, the system owner may make more profit on some items than others. In general, the utility will be a function of the item, the user and the supplier: $u_s(i, v, d)$.

The system delivers recommendation lists to users over time and in doing so produces utility for all the stakeholders. The full multi-stakeholder utility over any time period can be represented as a tuple of the following form

$$U_{full} = \langle u_{v_1}, u_{v_2}, ..., u_{d_1}, u_{d_2}, ..., u_s \rangle \qquad (1)$$

which has dimension $|V| + |D| + 1$ where V is the set of users and D is the set of suppliers. With such a representation, it is possible to compare the utility profiles of different recommendation algorithms, for example, looking at their Pareto efficiency.

In this paper, we simplify this full utility calculation by summarizing over each class of stakeholder, considering the total utility delivered to all users, to all suppliers, and to the system. We leave the consideration of the full high dimensional utility space for future work. Here we apply a more tractable three-dimensional model of recommendation utility:

$$U = \langle \sum_{v \in V} u_v, \sum_{dinD} u_d, u_s \rangle \qquad (2)$$

What should be clear from this exercise is that there are dimensions of recommender system performance that cannot be captured by a strict focus on users, and that expanding our view beyond users as stakeholders will have practical benefits. In practice, many businesses apply "business rules" to the output for their recommender systems for a variety of reasons: for example, to encourage users to try new products. A multi-stakeholder approach to recommendation invites such considerations into the overall system design and evaluation, rather than relegating them to separate processing tasks.

Optimizing this utility directly requires very detailed knowledge of the user and supplier base, down to the shape of each of these individual utility functions, knowing, for example, the value of each item to each user. Any practical application of this idea requires making assumptions that enable us to approximate these utilities. Below we discuss a particular recommendation scenario and the utility functions implied by it.

3. RECOMMENDATION TASK

Consider a movie recommendation system through which users choose pay-per-view streaming videos to watch. We will assume that there is a fixed utility for the presentation of movies that the user likes and no utility for options the user does not like. (One could imagine more sophisticated utility functions for users, of course.) For studios, each recommended movie is an opportunity to capture the user's streaming purchase. We will assume a fixed utility when one of the studio's movies is recommended to the user and none otherwise. (Again, more sophisticated utility modeling is possible.)

We make another assumption about the owner of the system. We assume that the streaming arrangements are exclusive: a movie is available only on one platform at a time, and the platform inventory may change as different deals are cut between the platform owner and studios. The owner wishes to please all of the studios with movies in the catalog so that they are not tempted to take their movies to a different platform. At the same time, the owner wishes to keep the customers happy as well: a user getting poor recommendations may also shift their business elsewhere. One could formulate this utility in different ways: for example, calculating lifetime value for customers and studios and their probability of flipping [6]. For our purposes, we will model this as simply counting the total number of satisfied stakeholders: happy users are those that are shown high quality movies (based on their interest), and happy suppliers are those whose movies get recommended.

Let T be a set of tuples of the form $\langle v, L\rangle$, where v is a target user and L is the list delivered to that user during a particular visit. T therefore represents the activity of the recommender system over a time period.

The utility for a particular user v over the time period is expressed by:

$$U_v(T) = \sum_{(v,L)\in T} \sum_{i\in L} G_v(i) \tag{3}$$

where $G_v(i)$ is the gain for user v in getting i as a recommendation.

The utility for each supplier over the time period is expressed by:

$$U_d(T) = \sum_{(v,L)\in T} \sum_{i\in L} G_d(i,v)\chi_d(i) \tag{4}$$

where $G_d(i,v)$ is the gain value of item i for supplier d when it is shown to user v and $\chi_d(i)$ is the indicator function for ownership of i by supplier d.

The utility for the system owner depends on the two sets of satisfied stakeholders for this time period: V^+ is the set of users that have positive utility, $V^+ = \{v \text{ where } U_v(T) > 0\}$; And D^+ is the set of suppliers with positive utility, $D^+ = \{d \text{ where } U_d(T) > 0\}$.

So, the total expected utility is represented by the tuple:

$$U = \langle \sum_{v\in V} U_v(T), \sum_{d\in D} U_d(T), |V^+| + |D^+|\rangle \tag{5}$$

What remains is to estimate the various utility functions involved. In the MovieLens dataset, user ratings are represented as an integer scale from 1 to 5, with 5 being the best. For our purposes here, we will make the simplifying assumption that a movie with rating 3 represents a "break-even" point with respect to utility. That is, a movie that has rating less then or equal to 3 has zero utility for user. Therefore, let $\rho(v,i)$ be the rating of the user v for the item i, the user gain G_a can be represented as follows:

$$G_v(i) = \begin{cases} 1 & \text{if } \rho(v,i) > 3 \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

Simply, we set the user utility for movies that his predicted rating is above a certain threshold (in this case it is 3) to 1 and otherwise we set the value to 0.

In this experiment we also suppose every supplier gains a fixed value from showing their movies. G_v is therefore constant. So, the supplier's utility reduces to the indicator function χ.

$$\chi_d(i) = \begin{cases} 1 & \text{if } i \text{ is produced by } d \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

4. COVERAGE-ORIENTED ALGORITHM

Collaborative recommendation algorithms are designed to optimize solely for users as stakeholders. Therefore, we would expect that they would have high values in the first dimension of our three-dimensional utility vector, but it is not clear that the other dimensions will be optimized. For example, the popularity bias in collaborative recommendation is well-known [2, 14]. A recommender with a strong popularity bias might concentrate recommendations on just a few suppliers resulting in low utilities for many suppliers and for the system owner.

As a simple example of the kind of algorithm that a utility-theoretic analysis suggests, we propose here a coverage-oriented algorithm that attempts to counter popularity bias, but does so at the supplier level. For a given set of recommendation lists, this reduces to a combinatorial joint optimization of user and supplier utility. However, the standard test / training evaluation scenario is not a good match to the application of such a metric in a live system. In a live system, the system is in the position of compensating at the present time for biases that occurred in prior recommendations. The system does not have the luxury of going back in time to change what recommendation lists were delivered to users in the past. What is called for in a live recommendation setting is an on-line algorithm that tries to achieve balance across suppliers over time [1].

Our proposed algorithm is a filtering mechanism applied to the recommendation lists returned by the recommender system. We create a hash map h keyed by the studio names containing as values the number of times a studio's movies have been shown so far. After a time period is complete, we update this map so that a running total is maintained in each entry $h(d)$.

During each time period, we use the historical coverage information stored in h as follows. First, we generate a recommendation list for the user of size K where K is larger than our desired recommendation size n. (In the experiments below, it is $5n$. So, n is 10 and K is 50). From this list, we remove all items with predicted rating less than or equal to 3. Then, we divide the recommendations into two lists, *seen* and *unseen*: the seen list consists of the items from studios where $h(d) > 0$, and the unseen list has the items from studios where $h(d) = 0$. We fill the recommendation list with items from the unseen list first. If there are fewer than n items on the unseen list, then the top items from the seen list are added until it reaches size n.

5. EVALUATION

The coverage-oriented algorithm above is designed to improve supplier coverage and owner utility over time by compensating for past popularity bias through reordering new recommendations. As this algorithm can only be evaluated in a time-sensitive way [3, 10], we evaluate the utility of a given recommendation algorithm given a history of rating data as follows.

1. Divide the data into k epochs by time (for example,

months).

2. For each epoch j, train the recommender on the preceding $j-1$ epochs.

3. Collect all of the tuples $\langle v, i, r\rangle$ (user, item and rating) for a given target user v in the data for epoch j. We will count this as a visit for testing purposes.

4. Use the recommender to compute a recommendation list of size n for user v.

5. Compute the utility for the list by treating the user's actual ratings from the test data as our ρ values.[1]

6. After iterating through all users, the utility vector for the epoch is calculated and can be added to the running total.

7. At the end of all epochs, we can produce the total utility vector for the system.

6. EXPERIMENTS

The dataset we use in our experiments is the well-known Movielens 1M rating database that contains 1,000,209 anonymous ratings of approximately 3,900 movies made by 6,040 MovieLens users who joined MovieLens in 2000 [7]. Following the procedure described in [3], we use the rating time stamps to split the data into 30 day epochs, starting at January 1, 2000 and ending at December 26, 2000: twelve 30-day epochs. The ratings in the data extend beyond this date and these ratings – approximately 6% of the total – are discarded and not used in our evaluation. We gathered the name of the studio releasing each movie from IMDb.

We use Apache Mahout as our recommender system implementation, modified by the inclusion of our utility metric and our coverage-oriented re-ranking scheme. For this preliminary work, we are using only user-based collaborative filtering with and without coverage-based re-ranking. For our experiments, we compute the utility assuming a recommendation list of size 10 and report utility at each epoch. We also compute precision at 10 and report the average over all active users for each epoch.

Table 1: Comparison of utility values for the users, suppliers and the system between the baseline algorithm and the greedy re-ranking approach

Algorithm	Users	suppliers	System
Baseline	1200	7473	6427
Re-ranking	1155	8053	6765
Total Gain (Loss)	(45)	580	338
% Gain (Loss)	(3.8%)	7.8%	5.3%

The table above summarizes the results for the experiment. We are not making any claim to the comparability of utilities across the three different dimensions, so the columns should be considered independently. There is a small utility loss for users and greater gains for the other stakeholders. The re-ranking algorithm was able to increase the item coverage (the number of different movies recommended) within the 11 epochs by about 15% from 1400 to 1605. It substantially increased the number of different studios represented among the recommendations by 37% from 545 to 746, working as designed to achieve a more equitable distribution of recommendations across studios.

Figure 1a shows the utility gain, the added utility for each group of stakeholders using the greedy algorithm over the baseline collaborative one. As might be expected with an algorithm focused on better studio coverage, the owners' collective utility is increased the most. The largest gain appears in the first epoch, where there is the largest number of unseen studios and then peaks again in the eighth epoch in tandem with precision shown below. The gain tapers to the end of the evaluation period because at this point there are few unseen studios and both algorithms return the same results. Figure 1b represents the same data in cumulative form.

The loss of user utility is a function of the small decrease in precision as shown in Figure 2. The left side of the figure shows the precision for each algorithm and the right side shows the differences at each epoch. The large loss in Epoch 2 corresponds exactly with the large increase in owner utility at the same period. In this epoch, the re-ranking algorithm is ignoring all of the studios with movies recommended in the first epoch; knowing the popularity bias of collaborative filtering, these are probably the most popular movies, which often come from larger studios with large movie inventories. These studios are filtered out, leaving less well-known movies to be shown in Epoch 2. A more sophisticated re-ranking scheme could take average rating and size of inventory into account when integrating the seen and unseen lists. The precision peak in month 8 may be due to a change in the characteristics of the users entering the system that this time, or it may be due to the benefits of larger training corpus. Note also that there are three epochs where the precision of the re-ranking algorithm exceeds the baseline by a small amount.

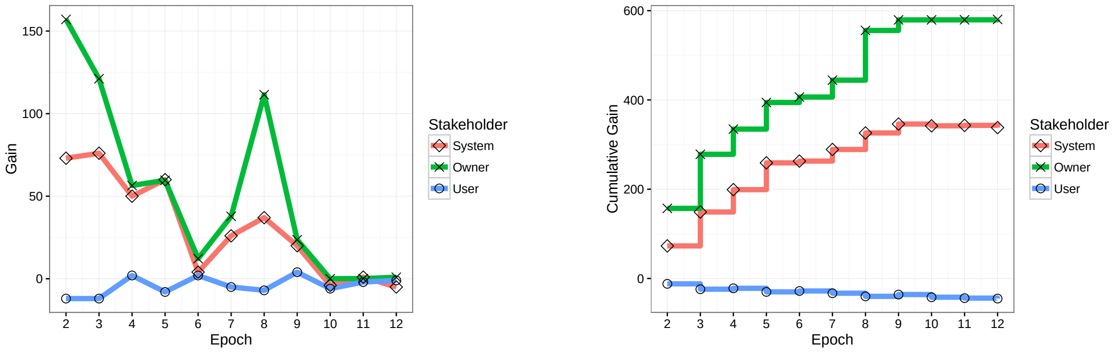
7. RELATED WORK

The concept of multiple stakeholders in recommender systems is suggested in a number of prior research works. As discussed above, researchers on reciprocal recommendation have looked at bi-lateral considerations to ensure that a recommendation is acceptable to both parties in the transaction [20].Similar ideas have appeared in work on group recommender systems where the goal is to find recommendation(s) that can maximize the utility of every stakeholder, which in this case are users in the group [12].

There is also some research that apply multiple constraints to recommendation generation such as [5]. Constraints can be understood as encoding utilities; however, this work does not take a multi-stakeholder approach. A more explicit utility-theoretic approach is taken by [17] in which a user's job seeking propensity is combined with their fit for a job description in ranking recruitment candidates in LinkedIn. This paper found that the combined utility approach yielded higher engagement rates than similarity alone.

There is a substantial literature in real-time targeted advertising in which advertisers' expected revenue and / or available budget are incorporated into the decision to deliver personalized advertising to a user. See [21] for a survey of some of this work. The BALANCE algorithm is designed to achieve balanced budget draw-down in an online advertis-

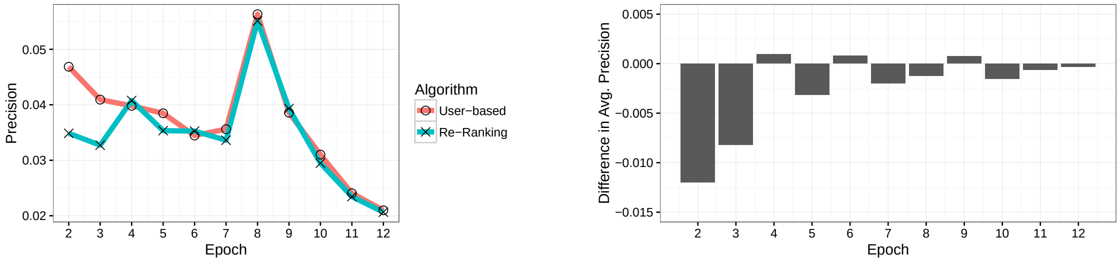
[1] As typical in this type of evaluation, we will ignore retrieved movies that are unrated because their utility is unknown. One could also imagine supplying an estimated average utility in such cases.



(a)

(b)

Figure 1: Utility gain



(a)

(b)

Figure 2: Precision@10

ing setting [9], and served as an inspiration for the coverage-oriented algorithm above.

Our approach is novel in that we explicitly represent the different stakeholders in the recommendation process and formalize their utilities. Our approach is sufficiently general that a wide variety of recommendation scenarios can be represented including reciprocal recommendation, budget management, and others.

There is a large body of recent work in recommender systems on incorporating diversity, novelty and other metrics into recommendation generation and evaluation. See, for example, [18, 23, 4, 19]. In a sense, all of these efforts can be understood as attempts to extend the notion of the utility of a recommendation (or a recommenation list) beyond simple rating prediction. Although we do not do so here, our model also provides a framework in which such considerations can be explicitly represented as utilities and accounted for in evaluation.

Similarly, multiple utilities may be in play when users' short-term preferences and their long-term well-being may have different associated utility functions. For example, lifestyle recommenders have been developed to encourage users to engage in healthful activities [11, 16]. In such systems, it is important not to recommend items that are too distant from the user's preferences – even if they would maximize health. The goal to be persuasive requires that the user's immediate context and preferences be honored. Athough these lifestyle recommenders, to date, have not taken a utility-oriented approach they can be understood in these terms.

8. CONCLUSION

There is increasing dissatisfaction with one-dimensional, accuracy-oriented, evaluation of recommender systems. In addition, real-world recommendation applications frequently require that recommender systems be sensitive to business needs and context. A utility-oriented approach allows us to represent the such concerns explicitly and make clear our modeling assumptions about the relative benefits of different aspects of recommender systems. A multi-stakeholder approach highlights the multiple actors involved in a given recommender system configuration and to allows the concerns of each to be represented and accounted for in evaluation and design.

This paper represents a preliminary examination of some of the consequences of this approach. Many simplifying assumptions have been required to formulate and conduct these experiments. However, we have shown that it is possible to make simple modifications to recommendation algorithms (in our case, coverage-oriented re-ranking) and yield utility improvements for suppliers and system owners, with minor losses in precision for users.

9. ACKNOWLEDGMENTS

This work is supported in part by the National Science Foundation under grant IIS-1423368.

References

[1] Susanne Albers and Stefano Leonardi. “On-line algorithms”. In: *ACM Computing Surveys (CSUR)* 31.3es (1999), p. 4.

[2] Erik Brynjolfsson, Yu Jeffrey Hu, and Michael D Smith. “From niches to riches: Anatomy of the long tail”. In: *Sloan Management Review* 47.4 (2006), pp. 67–71.

[3] Robin Burke. “Evaluating the dynamic properties of recommendation algorithms”. In: *Proceedings of the fourth ACM conference on Recommender systems*. ACM. 2010, pp. 225–228.

[4] Pablo Castells, Saúl Vargas, and Jun Wang. “Novelty and diversity metrics for recommender systems: choice, discovery and relevance”. In: (2011).

[5] Alexander Felfernig and Alfred Kiener. “Knowledge-based interactive selling of financial services with FSAdvisor”. In: *Proceedings of the National Conference on Artificial Intelligence*. Vol. 20. 3. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999. 2005, p. 1475.

[6] Sunil Gupta et al. “Modeling customer lifetime value”. In: *Journal of service research* 9.2 (2006), pp. 139–155.

[7] F Maxwell Harper and Joseph A Konstan. “The MovieLens Datasets: History and Context”. In: *ACM Transactions on Interactive Intelligent Systems (TiiS)* 5.4 (2015), p. 19.

[8] Ganesh Iyer, David Soberman, and J Miguel Villas-Boas. “The targeting of advertising”. In: *Marketing Science* 24.3 (2005), pp. 461–476.

[9] Bala Kalyanasundaram and Kirk R Pruhs. “An optimal deterministic algorithm for online b-matching”. In: *Theoretical Computer Science* 233.1 (2000), pp. 319–325.

[10] Neal Lathia et al. “Temporal diversity in recommender systems”. In: *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*. ACM. 2010, pp. 210–217.

[11] Yuzhong Lin et al. “Motivate: Towards context-aware recommendation mobile system for healthy living”. In: *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2011 5th International Conference on*. IEEE. 2011, pp. 250–253.

[12] Judith Masthoff. “Group recommender systems: Combining individual models”. In: *Recommender systems handbook*. Springer, 2011, pp. 677–702.

[13] Eli Pariser. *The filter bubble: How the new personalized web is changing what we read and how we think*. Penguin, 2011.

[14] Yoon-Joo Park and Alexander Tuzhilin. “The long tail of recommender systems and how to leverage it”. In: *Proceedings of the 2008 ACM conference on Recommender systems*. ACM. 2008, pp. 11–18.

[15] Luiz Pizzato et al. “RECON: a reciprocal recommender for online dating”. In: *Proceedings of the fourth ACM conference on Recommender systems*. ACM. 2010, pp. 207–214.

[16] Victor Ponce et al. “QueFaire: context-aware in-person social activity recommendation system for active aging”. In: *Inclusive Smart Cities and e-Health*. Springer, 2015, pp. 64–75.

[17] Mario Rodriguez, Christian Posse, and Ethan Zhang. “Multiple objective optimization in recommender systems”. In: *Proceedings of the sixth ACM conference on Recommender systems*. ACM. 2012, pp. 11–18.

[18] Barry Smyth and Paul McClave. “Similarity vs. diversity”. In: *Case-Based Reasoning Research and Development*. Springer, 2001, pp. 347–361.

[19] Saúl Vargas and Pablo Castells. “Rank and relevance in novelty and diversity metrics for recommender systems”. In: *Proceedings of the fifth ACM conference on Recommender systems*. ACM. 2011, pp. 109–116.

[20] Peng Xia et al. “Reciprocal Recommendation System for Online Dating”. In: *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*. ACM. 2015, pp. 234–241.

[21] Shuai Yuan et al. “Internet Advertising: An Interplay among Advertisers, Online Publishers, Ad Exchanges and Web Users”. In: *arXiv preprint arXiv:1206.1754* (2012).

[22] Weinan Zhang, Shuai Yuan, and Jun Wang. “Optimal real-time bidding for display advertising”. In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2014, pp. 1077–1086.

[23] Cai-Nicolas Ziegler et al. “Improving recommendation lists through topic diversification”. In: *Proceedings of the 14th international conference on World Wide Web*. ACM. 2005, pp. 22–32.