Optimization of Multi-stakeholder Recommender Systems for Diversity and Coverage

Iordanis Koutsopoulos, Maria Halkidi

To cite this version:

Iordanis Koutsopoulos, Maria Halkidi. Optimization of Multi-stakeholder Recommender Systems for Diversity and Coverage. 17th IFIP International Conference on Artificial Intelligence Applications and Innovations (AIAI), Jun 2021, Hersonissos, Crete, Greece. pp.703-714, 10.1007/978-3-030-79150-6_55 . hal-03287702

HAL Id: hal-03287702
https://inria.hal.science/hal-03287702

Submitted on 15 Jul 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

Optimization of Multi-Stakeholder Recommender Systems for Diversity and Coverage*

Iordanis Koutsopoulos[1] and Maria Halkidi[2]

[1] Department of Informatics, Athens University of Economics and Business, Greece
jordan@aueb.gr

[2] Department of Digital Systems, University of Piraeus, Greece
mhalk@unipi.gr

Abstract. Multi-stakeholder recommender systems (RSs) are a major paradigm shift from current RSs because recommendations affect not only item consumers (end-users) but also item providers (owners). They also motivate the need for new performance metrics beyond recommendation quality that explicitly affect the latter. In this work, we introduce a framework for optimizing multi-stakeholder RSs under constraints on diversity and coverage. Our goal is to make recommendations to end-users while treating each item provider equally, by ensuring sufficient user base coverage and diverse profiles of users to which items are recommended. Namely, items of each provider should be recommended to a certain number of users that are also diverse enough in their preferences. The optimization objective is that the total average rating of recommended items is as close as possible to that of a baseline RS. The problem is formulated as a quadratically constrained integer program, which is NP-Hard and impractical to solve in the presence of big data and many providers. Interestingly, we show that when only the coverage constraint exists, an instance of the problem can be solved optimally in polynomial time through its Linear Programming relaxation, and this solution can be used to initialize a low-complexity heuristic algorithm. Data experiments show good performance and demonstrate the impact of these constraints on average rating of recommended items.

Keywords: Multi-stakeholder Recommender Systems · Optimization · Diversity · Coverage

1 Introduction

Recommender systems (RSs) provide personalized recommendations to users through web or mobile app interfaces, and they have permeated social media,

* This work was supported by the CHIST-ERA grant CHIST-ERA-18-SDCDN-004, through the General Secretariat for Research and Innovation (GSRI), grant number T11EPA4-00056.

e-commerce, e-service, entertainment (e.g. music), sharing economy, and other domains. The functionality of these systems relies on data such as (i) explicit or implicit user feedback in the form of user records (e.g. book purchases, hotel stays, movie watches), log-on site activity (e.g. clicks, searches, item views), binary preferences (like/not like) or rating of items or services; (ii) items' attributes such as title, price, description; (iii) contextual information such as device used, location, time, and more.

These data constitute the training dataset of the RS which is provided as input to the recommendation engine to train a Machine Learning (ML) model. The model predicts preferences of users for items other than the ones that she has already experienced. A basic performance metric to optimize is the Mean Squared Error (MSE) between the predicted ratings for a specific ML model and the true ratings of the training dataset, which characterizes the quality of recommendation. A regularization term is added to the MSE error term, and the optimization of the new objective aims to avoid overfitting and increase the flexibility of the model to generalize well in unseen data.

Given the model above, a recommendation algorithm evaluates the ratings for user-item pairs that have not been rated, it selects for each user the top-L items (where $L = 5, 10, 20$) with the highest predicted ratings for that user, and it recommends them under no other constraint. Hence, items are recommended separately to each user, and the set of recommended items to one user does not affect the sets of items recommended to others.

However, more often than not, RSs are embedded in an online service platform, an online retail store or a social-media site, and there exist other entities besides end-users that are interested in recommendations. Owners, producers, providers or advertisers of recommended items may have some agreement with the recommendation engine about item promotion or about certain user outreach through recommendations, in exchange for some payment. For example, different brand chains of restaurants or hotels, editors or publishing companies of books, production and distribution companies of movies, or owners of sponsored items have paid to have the items appear in users' recommendation lists.

The terms *Two-sided markets* and *Multi-stakeholder Recommender Systems* [2] describe scenarios like the one above, where end-users (item consumers) and item providers need to be jointly taken into account when issuing recommendations. While item consumers are interested in good personalized experience through good-quality recommendations, item providers wish to receive good service by the recommendation engine as well. For example, different book publishers would like to have an adequate amount of exposure to users through recommendation. Likewise, different hosts in Airbnb would like to be represented in a fair manner in the recommendation lists of users. The consideration of the impact of the recommendation to item providers is a major paradigm shift from current RSs that focus only on rating prediction accuracy.

Our contribution. We address the problem of *optimizing Multi-stakeholder RSs* under constraints such as diversity and coverage that are specific to such RSs. We consider the setting where each item belongs to a set (class) of items of one owner or producer. Our goal is to make the recommendation of items to end-users and treat each provider equally by ensuring sufficient user base coverage and user diversity for their recommended items. That is, each set of items must be recommended to a certain number of users that are also diverse enough in terms of user profile. For example, in the case of hotel chains, a *coverage* constraint dictates that *each* hotel chain appears in the recommendation lists of a certain minimum number (or percentage) of users. A *diversity* constraint would imply that each hotel chain should be recommended to users with sufficiently diverse profiles and tastes so as to increase hotel reach and penetration.

Our optimization objective is to recommend items to users so that the total average rating of items that are recommended to users is affected by the minimum amount, compared to the ratings of recommended items from a baseline RS, subject to maintaining sufficient diversity and coverage for each provider.

The main challenge is that these constraints on the provider side lead to a *coupling* between the sets of items to be recommended to each user, and therefore these need to be decided *jointly* for all users. For example, for multiple hotel chains and a coverage constraint, each hotel chain should appear in the recommendation lists of a certain number of users, and thus the recommendation algorithm should jointly decide on the recommendations to all users rather than separately as in conventional RSs. This makes the recommendation problem much more composite and challenging compared to state of the art, and to the best of our knowledge, this problem has not been addressed. The existence of high-dimensional data and multiple involved stakeholders further exacerbates the problem. The contributions of our work to the literature are as follows:

- We introduce the problem of optimizing Multi-stakeholder Recommender Systems subject to diversity and coverage constraints and formulate it as a quadratically constrained integer program, which is NP-Hard and impractical to solve in the presence of big data and many providers. Items to be recommended are viewed as "resources", and the framework abstracts the recommendation problem as a resource allocation one.
- We show that when only the coverage constraint exists, an instance of the problem can be solved optimally in polynomial time through its Linear Programming (LP) relaxation, and we use this solution to appropriately initialize a low-complexity heuristic algorithm.
- We perform initial experiments with the publicly available Movielens dataset, which show good performance and demonstrate the impact of coverage and diversity constraints on the achieved average rating of recommended items through our approach.

The rest of the paper is organized as follows. In section 2, we present an overview of state of the art. In section 3 we provide the model and some derivations and definitions that are used next in section 4 to formulate and solve the problem. In section 5, we present initial experimental results on publicly available datasets, and in section 6 we conclude the paper. The terms "list of recommended items" and "recommendation list" refer to the set of items that are recommended to a user.

2 Related work

Diversity, coverage and positive surprise (i.e. serendipity) are recognized as critical aspects of user Quality of Experience (QoE) in RSs, and they admit various definitions and interpretations [6]. From the point of view of the user, diversity refers to recommending diverse items to her so as to eliminate the item bubble effect. On the other hand, coverage is the number of items that appear in the lists of recommended items to all users. A broad class of techniques to address diversity and other metrics (such as coverage, serendipity) uses re-ranking [3]. In these techniques, a baseline recommendation algorithm is used to generate predicted ratings, and these are sorted in decreasing order for each user. Next, these items are re-ordered according to some further performance objective. For example, if diversity is the goal, items that are ranked lower in terms of predicted rating but differ in their profile from those in the current top-L list, are placed in that list so as to increase diversity. Likewise, if item popularity is sought in the recommendation, each item rating is re-weighted by its popularity, i.e. the number of users that have rated it, and thus the relative position of items in the recommendation list changes. The work [14] falls within that class of works and places emphasis on genre diversity and coverage for a user.

In the work [4], the authors propose system-wide diversity metrics to simultaneously achieve diversification of the categories of items that each user sees and diversification of the types of users to which each item is recommended, while maintaining high recommendation quality. In [8], diversity metrics are considered for both the items and the users, and for single-user and group recommendations. The work [5] studies the problem of identifying k products that cover maximum number of consumers so as to maximize the probability of product purchase.

Multi-stakeholder recommender systems (MSRSs) and two-sided markets have been an active research area recently, with a broad range of application areas and a promising roadmap of directions [16]. A taxonomy of multi-stakeholder RSs with respect to item providers, item consumers and their preferences as well as possible side stakeholders is given in [2]. A comprehensive survey of the area with emphasis on the recent trends of people recommendation, value-aware recommendation and fairness aspects is presented in [1]. The work in [13] formulates the problem of maximizing the rating value of recommended items subject to provider constraints as an integer programming problem that is solved using Lagrangian relaxation and subgradient methods. Coverage is studied also in [7] with the goal to find an average rating of recommended items that is as close as possible to that of a baseline RS and to balance the rating deviation from a baseline RS across users and across items.

Fairness towards item providers and consumers is an active area in MSRSs. In [11], fairness for the providers' side is addressed in the form of recommendation updates so that providers are guaranteed similar amounts of exposure to users. Fairness for both providers and consumers has also been addressed in the context of ride-hailing platforms [12], where repeated matching of providers (drivers) to consumers (ride requesters) is performed so that in the long run, similar utilities across providers and across consumers are achieved. Different notions of fairness inspired by classical economics theory are evaluated for providers and consumers of a RS in [10]. The tradeoff between providing relevant recommendations to users and guaranteeing fair representations of different music artists in users' recommendation lists is addressed in [9].

Compared to the current literature we explicitly define a novel diversity and coverage metric which give rise to non-linear and linear constraints respectively. In fact, we show that the coverage constraint helps to simplify the diversity one. Contrary to [13], we formulate the problem as a quadratically constrained integer program. We also show that if only the coverage constraint exists, an instance of the problem can be solved in polynomial time through its LP relaxation, since the constraint matrix becomes unimodular.

3 Model

We consider a set $\mathcal{I}$ of items and a set $\mathcal{U}$ of users. We assume that a baseline recommendation system (e.g. Collaborative Filtering) generates a list of recommended items $\mathcal{L}_u$ for each user u, where $|\mathcal{L}_u| = L$, with L typically taking values $1, 2, 5, 10, 20$. Denote by $\mathcal{L} = (\mathcal{L}_u : u \in \mathcal{U})$ the output of the baseline RS in terms of the list of recommended items to each user u. Let $\mathcal{U}_i$ be the subset of users to which item $i \in \mathcal{I}$ is recommended according to the baseline RS algorithm, and let $|\mathcal{U}_i|$ be its cardinality. For each user u and item i, let r_{iu} denote the predicted rating of item i for user u according to the baseline RS algorithm.

There exist C predefined item sets (classes). A set (class) stands for a different provider, for example a different book publisher if items are books, or a different production company if items are movies. Each item is assumed to belong to exactly one class. Denote by $\mathcal{C}$ the set of C classes. Let $\mathcal{I}_c$ be the set of items in class c, for $c = 1, \dots, C$.

3.1 Deviation from baseline recommendations

We are interested to find a new recommendation policy with an output $\mathcal{L}' = (\mathcal{L}'_u : u \in \mathcal{U})$, where $\mathcal{L}'_u$ is the *new* list of recommended items to user u. These *new* lists of recommended items should satisfy the coverage and diversity constraints. That is, items from *each* class are recommended to a large enough number of users, and these users have diverse enough profiles. The definitions of coverage and diversity will be provided in the sequel, in subsections 3.2 and 3.3.

Let $\mathbf{x} = (x_{iu} : i \in \mathcal{I}, u \in \mathcal{U})$ denote this new recommendation policy, where for each item i and user u, the binary variable $x_{iu} = 1$ if item i is recommended to user u, and $x_{iu} = 0$ otherwise. Namely $x_{iu} = 1$ if $i \in \mathcal{L}'_u$, and 0 if not.

When performing the update in the list of recommended items $\mathcal{L} \to \mathcal{L}'$, we would like to minimize the cost of the impact of this change to users. For each user u, this cost is defined as the average difference between the sum of ratings of items in list $\mathcal{L}_u$ generated by the baseline RS, and the sum of ratings in the new list $\mathcal{L}'_u$. For example, consider a user u and $|\mathcal{L}_u| = 2$ so that the baseline RS recommends two items A, B with predicted ratings 4.8 and 4.6, on a scale from 1 to 5. Assume that the new recommendation list that attempts to satisfy coverage and diversity constraints comprises two items other than A,B, with predicted ratings 4.7 and 4.2 respectively. Then, the average cost incurred to the user is $\frac{1}{2}[(4.8-4.7)+(4.6-4.2)] = 0.25$.

For the entire user set $\mathcal{U}$, the total average cost is expressed as a function of policy $\mathbf{x}$ as follows:

$$\mathrm{Cost}_{\mathcal{L}\to\mathcal{L}'}(\mathbf{x}) = \frac{1}{|\mathcal{U}|}\sum_{u\in\mathcal{U}}\frac{1}{L}\left(\sum_{i\in\mathcal{L}_u} r_{iu} - \sum_{i\in\mathcal{I}} r_{iu}x_{iu}\right) \tag{1}$$

Note that $\mathrm{Cost}_{\mathcal{L}\to\mathcal{L}'}(\mathbf{x}) \geq 0$, for any policy $\mathbf{x}$, since the first term is always larger than the second one, because it is the result of the baseline RS which recommends to each user the L items with highest rating.

3.2 User coverage

In an effort to treat the C providers equally, a first requirement is that the new recommendation algorithm should recommend items from each class $c\in\mathcal{C}$ to an adequate number of users i.e. make sure that items from each provider appear in the recommendation lists of enough users. This requirement is realistic and arises because of bilateral agreements between the provider and the recommendation platform.

User coverage for a class of items c can be defined in various ways, e.g. as the number of users to which items of class c are recommended, expressed as $\sum_{u\in\mathcal{U}}\min\{1,\sum_{i\in\mathcal{I}_c} x_{iu}\}$. Namely, user coverage for class c is the number of users to which at least one item of class c is recommended. If there are more than one items of class c recommended to a user u, i.e. if $\sum_{i\in\mathcal{I}_c} x_{iu} > 1$, then this user would count as one user in the coverage.

In this work, we adopt a simpler form of user coverage. For each item $i\in\mathcal{I}_c$ separately, we count the number of users to whom item i is recommended. The average per-item user coverage for items of class c is then given as

$$\overline{\mathrm{Cov}}_c(\mathbf{x}) = \frac{1}{|\mathcal{I}_c|}\sum_{i\in\mathcal{I}_c}\sum_{u\in\mathcal{U}} x_{iu}\,. \tag{2}$$

Let K_c denote the constraint on the average number of times that an item of class c is assigned to a user, which is specified by the agreement between provider c and the recommendation platform. Namely, it is $\overline{\mathrm{Cov}}_c(\mathbf{x}) = K_c$. Hence, the total number of times that an item of class c is assigned to a user is $K_c|\mathcal{I}_c|$, and K_c takes values in $[0,\ldots,|\mathcal{U}|]$.

3.3 User diversity

Another aspect of the agreement between a provider and the platform may concern user diversity. In this case, the platform should recommend the set of items $\mathcal{I}_c$ of each class (provider) c to a set of users that is as diverse as possible, in an effort to further expand item reach in the user base. For example, a book publisher may be interested in having a novel recommended to readers of diverse tastes or age groups, or an Airbnb accommodation RS may wish to recommend different accommodations owned by one host to diverse groups of users in terms of interests, and so on.

In this work, we abstract the profile similarity between a pair of users (u, v) as w_{uv}. This can be computed through *cosine similarity or the Pearson correlation coefficient* over a set of predefined features. Let d_{uv} be the *dissimilarity* between users u, v, defined as $d_{uv} = 1 - w_{uv}$, so that $0 \leq d_{uv} \leq 2$.

For items of class c, the per-item average diversity of users to which these items are recommended is

$$\mathrm{Div}_c(\mathbf{x}) = \frac{1}{|\mathcal{I}_c|} \sum_{i \in \mathcal{I}_c} \sum_{u \in \mathcal{U}} \sum_{v \in \mathcal{U}: v \neq u} d_{uv} x_{iu} x_{iv} \,. \tag{3}$$

For example, consider a class with 2 items, item i_1 and i_2 and assume 3 users, u_1, u_2 and u_3. Suppose that i_1 is recommended to users u_1, u_2, and i_2 is recommended to all users u_1, u_2, u_3. Then, the average user diversity is $\frac{1}{2}(2d_{u_1u_2} + d_{u_2u_3} + d_{u_1u_3})$.

We can normalize diversity by dividing with the number of pairs of users to which items of class c are assigned, which is

$$\frac{1}{2}\big(\sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}_c} x_{iu}\big)\big(\sum_{u \in \mathcal{U}} \sum_{i \in \mathcal{I}_c} x_{iu} - 1\big) \,. \tag{4}$$

E.g. in the example above, we can further divide with $\frac{1}{2} \times 5 \times 4 = 10$, where 5 is the total number of users to which an item of the class is assigned (i.e. item i_1 to users u_1, u_2, and item i_2 to users u_1, u_2, u_3).

Therefore, the average normalized diversity (per item and per-user pair) is written as

$$\overline{\mathrm{Div}}_c(\mathbf{x}) = \frac{2}{|\mathcal{I}_c|} \frac{\sum_{i \in \mathcal{I}_c} \sum_{u \in \mathcal{U}} \sum_{v \in \mathcal{U}: v \neq u} d_{uv} x_{iu} x_{iv}}{\left(\sum_{i \in \mathcal{I}_c} \sum_{u \in \mathcal{U}} x_{iu}\right) \times \left(\sum_{i \in \mathcal{I}_c} \sum_{u \in \mathcal{U}} x_{iu} - 1\right)} \,. \tag{5}$$

Because of the coverage constraint above, this is simplified to

$$\overline{\mathrm{Div}}_c(\mathbf{x}) = 2 \times \frac{\sum_{i \in \mathcal{I}_c} \sum_{u \in \mathcal{U}} \sum_{v \in \mathcal{U}: v \neq u} d_{uv} x_{iu} x_{iv}}{K_c |\mathcal{I}_c|^2 \big(K_c |\mathcal{I}_c| - 1\big)} \,, \tag{6}$$

and for $K_c|\mathcal{I}_c| >> 1$, this can be approximated as

$$\overline{\mathrm{Div}}_c(\mathbf{x}) \simeq \frac{2}{K_c^2 |\mathcal{I}_c|^3} \sum_{i \in \mathcal{I}_c} \sum_{u \in \mathcal{U}} \sum_{v \in \mathcal{U}: v \neq u} d_{uv} x_{iu} x_{iv} \,. \tag{7}$$

We define the constraint $\overline{\mathrm{Div}}_c(\mathbf{x}) = D_c$, where D_c is a specified average per-item and per-user pair diversity that should be satisfied for recommendations of items of class c.

4 Problem formulation and solution

The objective is to generate new lists of recommended items $\{\mathcal{L}'_u\}$ for each user $u \in \mathcal{U}$ so as to minimize the cost function (1). This objective is equivalent to:

$$\max_{\mathbf{x}} \frac{1}{L|\mathcal{U}|} \sum_{u\in\mathcal{U}} \sum_{i\in\mathcal{I}} r_{iu} x_{iu} \,, \tag{8}$$

subject to the following constraints:

$$\sum_{c} \sum_{i\in\mathcal{I}_c} x_{iu} = L \,, \text{ for each user } u \tag{9}$$

$$\overline{\mathrm{Div}}_c(\mathbf{x}) = D_c \,, \text{ for each provider } c \,, \tag{10}$$

$$\overline{\mathrm{Cov}}_c(\mathbf{x}) = K_c, \text{ for each provider } c \,, \tag{11}$$

and $x_{iu} \in \{0, 1\}$ for each i, u. Constraint (9) says that L items should be recommended to each user u, since $|\mathcal{L}'_u| = L$ for each u. Constraint (10) says that for each item class (provider) c, an average per-item and per-user pair user diversity D_c should be satisfied. Finally, constraint (11) says that for each provider c, the total number of times an item of class c is assigned to a user should be $K_c|\mathcal{I}_c|$, where $K_c \in [0, \dots, |\mathcal{U}|]$. Note that $K_c/|\mathcal{U}|$ is the minimum percentage of users to which items of class c are recommended.

We refer to problem (8)-(11) as problem (**P**). This is a Quadratically constrained Integer Program (QCIP), since the objective is linear in $\mathbf{x}$ and constraints (10) are quadratic in $\mathbf{x}$. This problem is NP-Hard.

4.1 An interesting special case

Consider problem (**P**) without the diversity constraints (10), and without constraints $x_{iu} \in \{0, 1\}$. Call this problem (**P1**). Namely, problem (**P1**) is that of maximizing (8) subject to constraints (9), (11) and $x_{iu} \in \mathbb{Z}_+$ i.e. a positive integer for all i, u. Now, consider the Linear Program (LP) relaxation (**P1**') of (**P1**), with the same objective and constraints, but with continuous-valued variables $x_{iu} \in \mathbb{R}_+$ for all i, u. All these constraints can be written succinctly in matrix and vector form with the following linear form:

$$\mathbf{A}\mathbf{x} = \mathbf{b} \,, \tag{12}$$

where $\mathbf{A}$ is a $n \times m$ matrix, with $n = |\mathcal{U}| + C$, $m = |\mathcal{U}||\mathcal{I}|$, and $\mathbf{b}$ is a vector with n elements with the first $|\mathcal{U}|$ elements equal to L, and the last C elements equal to $K_1|\mathcal{I}_1|, \dots, K_C|\mathcal{I}_C|$.

We call a constraint matrix $\mathbf{A}$ of a LP *totally unimodular* (TU) if each square sub-matrix of $\mathbf{A}$ has determinant $+1$, -1 or 0. According to a theorem from [15, Sec. 3.2], a matrix $\mathbf{A}$ is TU if the following three conditions hold: (i) matrix elements $a_{ij} \in \{+1, -1, 0\}$ for all i, j; (ii) each matrix column contains at most two nonzero elements, and (iii) there exists a partition of the set of rows in two subsets $\mathcal{M}_1$ and $\mathcal{M}_2$, and each column j with two nonzero coefficients satisfies $\sum_{i\in\mathcal{M}_1} a_{ij} - \sum_{i\in\mathcal{M}_2} a_{ij} = 0$.

Matrix $\mathbf{A}$ satisfies all conditions above, and therefore it is TU. Next, we have the following proposition, from [15, Sec.3.2]: An LP problem with feasible set $\{\mathbf{Ax} = \mathbf{b},\ \mathbf{x} \in \mathbb{R}_+\}$ has an integral optimal solution for all integer vectors $\mathbf{b}$ for which it has an optimal value, if and only if $\mathbf{A}$ is TU. In our case, vector $\mathbf{b}$ is an integer vector since L, $\{K_1, \ldots, K_C\}$ and $\{|\mathcal{I}_1|, \ldots, |\mathcal{I}_C|\}$ are positive integers by definition.

Therefore, the optimal solution of the LP problem (**P1**') is in integer form. Thus, the optimal solution to (**P1**') (e.g. through the Simplex algorithm) is the same as the optimal solution to (**P1**), i.e. with the coverage constraints satisfied. However, one issue is that $x_{iu} \in \mathbb{Z}_+$ in problem (**P1**), and thus in the optimal solution of (**P1**), there will exist items i and users u for which $x_{iu} > 1$. We deal with this issue in the sequel.

4.2 Heuristic algorithm for problem (P)

When the size of problem (**P**) is large, namely the number of items and item providers is large, the numerical evaluation of the solution becomes difficult. We consider the following low-complexity heuristic algorithm which uses the solution of the LP problem (**P1'**) as follows.

We start by finding the optimal solution to (**P1'**). This solution is feasible in terms of coverage for each provider, and it is also feasible in terms of recommending L items to each user. However, constraints $x_{iu} \in \{0, 1\}$ may not be satisfied, i.e. the algorithm may recommend the same item to the same user more than once. Furthermore, the diversity constraints may not be satisfied because they have not been taken into account in (**P1'**). If this is the case, we need to choose an item j from a provider c that has been repeatedly recommended to a user u, and substitute it with a non-recommended item k of the same provider so as to maintain coverage feasibility and the L recommended items to each user. Then, we assign item k to u.

We make the choice of j, u, c, k as follows. We choose a provider c, a user u, a recommended item j to user u to substitute, and a not yet recommended item $k \in \mathcal{I}_c$ (so as to substitute j with k) for which the rating r_{ku} is as close as possible to r_{ju}, and the diversity increase is large if it is recommended to user u. We continue the substitutions by appropriately choosing j, u, c and i in that fashion until each item is recommended at most once to the same user. Next, in order to reach feasibility in the diversity constraints, we do item substitutions across providers, by switching items from different providers between users, while keeping the coverage constraints and $|\mathcal{L}_u| = L$ satisfied for each user, by giving priority to those switches that most improve diversity.

5 Data experiments

We experiment with the publicly available `Movielens ml-latest-small` dataset to evaluate the effect of diversity and coverage constraints on the average

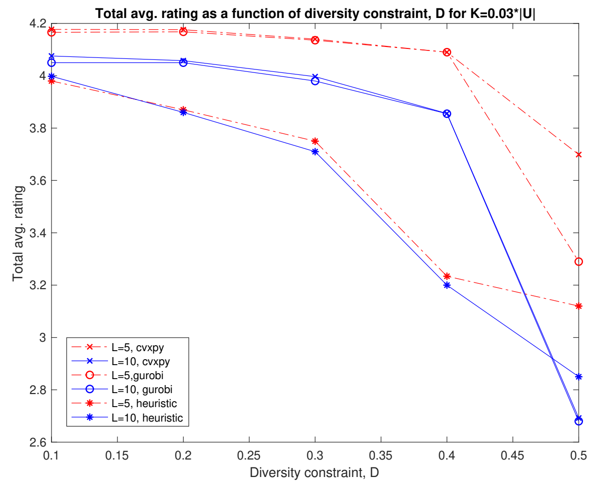

Fig. 1. Total average rating of recommendations for different solution approaches and different number of recommended items, L for $K = 0.03 \cdot |\mathcal{U}|$.

rating, by solving problem (**P**). The dataset contains $100,000$ ratings from 671 users and $9,125$ movies, and ratings are on a 5-star scale with 0.5-star increment. We consider only those users that have rated at least two movies. We set $K_c = K$ and $D_c = D$ for all classes (providers) c. The movies in the dataset are then assigned randomly to one provider out of $C = 5$ or $C = 10$ providers.

Next, we compute the similarity w_{uv} between user pairs (u, v). For each user pair (u, v), we record which movies have been rated by both users, and we compute similarity w_{uv} as the similarity between user ratings for this common set of movies, by using Pearson correlation. We then produce the baseline RS lists for users through item-item Collaborative Filtering (CF). The output of item-item CF is a user-item ratings matrix. We take as number of recommended items to each user u, $|\mathcal{L}_u| = L = 5$ or 10.

Next, we numerically solve problem (**P**) in the following versions:

- A Non-Linear Programming problem with continuous variables $x_{iu} \in [0, 1]$, by using the Python `cvxpy` solver. This gives an upper bound on the total rating achieved by any algorithm that has integer variables $\mathbf{x}$.
- A Quadratically constrained Integer Programming (QCIP) problem with discrete variables $x_{iu} \in \{0, 1\}$, by using the Gurobi solver.
- The heuristic algorithm of subsection 4.2 above.

Figures 1 and 2 show initial comparative results of these approaches as a function of the diversity constraint D for user coverage percentages $K = 3\%$ and $K = 10\%$, respectively. For all approaches, as both constraints become more stringent, the achieved average rating of recommended items decreases. The solver achieves an integer feasible solution with objective function value

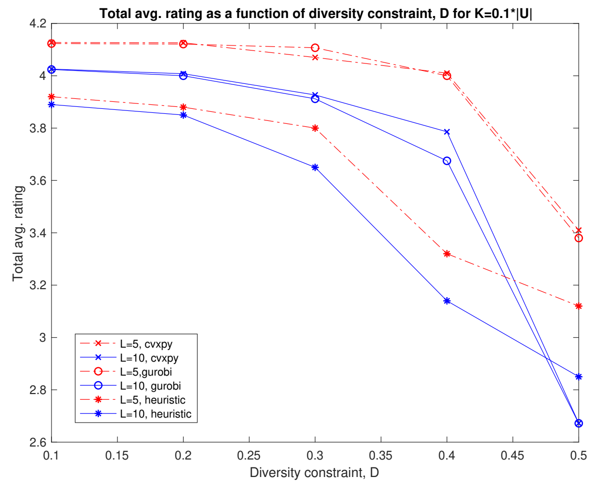

Fig. 2. Total average rating of recommendations for different solution approaches and different number of recommended items, L for $K = 0.1 \cdot |\mathcal{U}|$.

that is indeed very close to that of the LP. The heuristic algorithm achieves satisfactory rating performance for moderate values of D. From the plots above, it becomes apparent that the diversity constraint has larger impact than the coverage constraint on the reduction of average rating of recommended items.

6 Conclusion

We study the problem of optimizing multi-stakeholder recommender systems subject to user coverage and diversity constraints for different item providers that need to be treated equally in terms of guarantees for these metrics. The diversity constraint places quadratic constraints to the problem and makes it hard to solve. This study is a first step that serves as a proof-of-concept validation, and we intend to precisely quantify the various tradeoffs and impact of different parameters and constraints, through a larger dataset.

Looking ahead, the problem of equal treatment of item providers has many more angles to reveal, and we have only scratched the surface here. We viewed the problem through the lens of resource allocation where “resources” are the items to recommend. This parallelism to resource allocation may inspire different notions of fairness. For example, the average rating of recommendations for each provider should be assessed as well, and fair treatment may imply deciding on average ratings for each provider in a max-min fair sense, proportionally to the provider significance, or in an envy-free manner, whereby no provider would be willing to swap its user base and recommendations with those of another provider. A more fine-grained approach could value differently the rank (i.e.

the position) of an item in the recommendation list. Other metrics could be considered for providers, such as serendipity.

References

1. Abdollahpouri, H., Adomavicius, G., Burke, R., Guy, I., Jannach, D., Kamishima, T., Krasnodebski, J., Pizzato, L.A.: Beyond personalization: Research directions in multistakeholder recommendation. ArXiv: abs/1905.01986 (2019)
2. Abdollahpouri, H., Burke, R.D.: Multi-stakeholder recommendation and its connection to multi-sided fairness. ArXiv: abs/1907.13158 (2019)
3. Adomavicius, G., Kwon, Y.: Improving aggregate recommendation diversity using ranking-based techniques. IEEE Transactions on Knowledge and Data Engineering **24**(5), 896–911 (2012)
4. Antikacioglu, A., Bajpai, T., Ravi, R.: A new system-wide diversity measure for recommendations with efficient algorithms. SIAM J. Math. Data Sci. **1**(4), 759–779 (2019)
5. Hammar, M., Karlsson, R., Nilsson, B.J.: Using maximum coverage to optimize recommendation systems in e-commerce. In: Proceedings of the 7th ACM Conference on Recommender Systems (2013)
6. Kaminskas, M., Bridge, D.: Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems. ACM Trans. Interact. Intell. Syst. **7**(1) (2016)
7. Koutsopoulos, I., Halkidi, M.: Efficient and fair item coverage in recommender systems. In: IEEE 4th Intl Conf on Big Data Intelligence (DataCom) (2018)
8. Kyriakidi, M., Stefanidis, K., Ioannidis, Y.: On achieving diversity in recommender systems. In: Proceedings of the ExploreDB (2017)
9. Mehrotra, R., McInerney, J., Bouchard, H., Lalmas, M., Diaz, F.: Towards a fair marketplace: Counterfactual evaluation of the trade-off between relevance, fairness and satisfaction in recommendation systems. In: Proceedings of the ACM Int. Conf. on Information and Knowledge Management (2018)
10. Patro, G.K., Biswas, A., Ganguly, N., Gummadi, K.P., Chakraborty, A.: Fairrec: Two-sided fairness for personalized recommendations in two-sided platforms. In: Proceedings of The Web Conference (2020)
11. Patro, G.K., Chakraborty, A., Ganguly, N., Gummadi, K.P.: Incremental fairness in two-sided market platforms: On smoothly updating recommendations. ArXiv: abs/1909.10005 (2019)
12. Sühr, T., Biega, A.J., Zehlike, M., Gummadi, K.P., Chakraborty, A.: Two-sided fairness for repeated matchings in two-sided markets: A case study of a ride-hailing platform. In: Proceedings of the ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining (2019)
13. Sürer, O., Burke, R., Malthouse, E.C.: Multistakeholder recommendation with provider constraints. In: Proceedings of the 12th ACM Conference on Recommender Systems (2018)
14. Vargas, S., Baltrunas, L., Karatzoglou, A., Castells, P.: Coverage, redundancy and size-awareness in genre diversity for recommender systems. In: Proceedings of the ACM Conference on Recommender Systems (2014)
15. Wolsey, L.A.: Integer Programming. Wiley-Interscience, USA (1998)
16. Zheng, Y.: Multi-stakeholder recommendations: Case studies, methods and challenges. In: Proceedings of the ACM Conf. on Recommender Systems (2019)