Multi-Objective Group Discovery on the Social Web

Behrooz Omidvar-Tehrani[1](✉), Sihem Amer-Yahia[2],
Pierre-Francois Dutot[2], and Denis Trystram[2]

[1] The Ohio State University, Columbus, USA
omidvar-tehrani.1@osu.edu
[2] Univ. Grenoble Alps, CNRS, Grenoble, France
{sihem.amer-yahia,pierre-francois.dutot,trystram}@imag.fr

Abstract. We are interested in discovering user groups from collaborative rating datasets of the form $\langle i, u, s\rangle$, where $i \in \mathcal{I}$, $u \in \mathcal{U}$, and s is the integer rating that user u has assigned to item i. Each user has a set of attributes that help find *labeled groups* such as *young computer scientists in France* and *American female designers*. We formalize the problem of finding user groups whose quality is optimized in multiple dimensions and show that it is NP-Complete. We develop α-MOMRI, an α-approximation algorithm, and h-MOMRI, a heuristic-based algorithm, for multi-objective optimization to find high quality groups. Our extensive experiments on real datasets from the social Web examine the performance of our algorithms and report cases where α-MOMRI and h-MOMRI are useful.

1 Introduction

Today's data scientists are faced with large volumes of data to explore. In particular, collaborative rating sites have become essential data resources to make decisions about mundane tasks such as purchasing a book, renting a movie or going to a restaurant. The availability of a number of datasets on the social Web, such as MovieLens, a movie rating site, LastFM, a music rating site and BookCrossing, a book rating site, appeals to scientists today who design algorithms that help analysts make better decisions on complex tasks such as crowd data sourcing (which users to ask ratings from), advertisers in determining which items to recommend to which users, and social scientists in validating hypotheses such as *young professionals are more inclined to buying self-help books*, on large datasets.

In practice, however, there does not exist analytics tools that enable the scalable, on-demand discovery of user groups. In this paper, we are given a dataset of rating records in the form $\langle i, u, s\rangle$, where $i \in \mathcal{I}$ (set of items), $u \in \mathcal{U}$ (set of users), and s is the integer rating that user u has assigned to item i. We define the notion of *user group* as a conjunction of demographic attributes over rating records, such as *rich young professionals* or *teachers who live in the countryside*. Given a dataset, e.g., ratings of Woody Allen movies, we formalize

P. Frasconi et al. (Eds.): ECML PKDD 2016, Part I, LNAI 9851, pp. 296–312, 2016.
DOI: 10.1007/978-3-319-46128-1_19

the problem of discovering *high quality* user groups. Quality is formulated as the optimization of two dimensions: *coverage* and *diversity*. Optimizing coverage ensures that most input records $\langle i, u, s \rangle$ will belong to at least one group in the output. Optimizing diversity ensures that found groups are as different as possible from each other, e.g., *males and females* or *young and old*, and unveils ratings by different users. User groups with high coverage and high diversity, can help analysts make a variety of decisions such as audience targeting in advertising or hypothesis validation in social science. Example 1 illustrates a common case in practice.[1]

Example 1. It is generally believed that romantic movies (e.g., *American Beauty*, 1999) are mostly watched by females. This observation is based on *demographic breakdown* reports on IMDb.[2] Anna, who is a social scientist, wants to validate this hypothesis by exploring diverse user groups that cover most ratings for *romance* genre movies. Such a group-centric examination would provide the following 3 user groups: *i. female reviewers from DC* (District of Columbia), *ii. young female reviewers*, and *iii. male teenager reviewers* with average ratings of 4.6, 3.7 and 3.1 (out of 5), respectively. By observing those groups, Anna finds that the hypothesis holds only for a sub-population of female reviewers, *middle-age* or *residents of DC*. Also the results show another group of *romance* genre lovers, *male teenagers*, which contradicts the hypothesis. Anna is confident in her observation (as the results has high coverage) and she can notice different aspects of her hypothesis (as results are diversified).

Beyond coverage and diversity, another interesting dimension of group quality is its *rating distribution*. As it has been argued in previous work [4], groups with *homogeneous* ratings may be more appealing to some applications, while groups with *polarized* ratings are preferred by others. Indeed the rating distribution in a group provides analysts with the ability to tune the quality of found groups according to specific needs. Example 1 is a good case for *homogeneity*. By reporting the average rating of 4.6 for young female reviewers, we know that most individuals in that group have high ratings. The following example shows how tuning the *rating distribution* of discovered groups leads to new discoveries when used alongside coverage and diversity.

Example 2. Following Example 1, Anna then looks at the *variance* of ratings in those groups and finds that *male teenager reviewers* has a higher variance comparing to two other groups. This potentially shows that not all male teenagers like romantic movies. Anna is more interested in a homogenous group, so she can either choose the second or third group or ask the system to find other groups specifically for males or teenagers.

Given an input set of rating records (e.g., Sci-Fi movies from the 90's, David Lynch movies, movies starring Scarlett Johansson), our problem is that of discovering a set of user groups. Even when the number of records is not very high,

[1] *We use this example as our running example throughout the paper.*

[2] http://www.imdb.com.

the number of possible groups that could be built may be very large. Indeed, the number of groups is exponential in the number of user attribute values and many groups are very small or empty. Therefore, given the ad-hoc and online nature of group discovery, our challenge is to *quickly* identify high quality user groups. We hence define desiderata that user groups should satisfy (local desiderata) and those that must be satisfied by the set of returned groups (global desiderata).

Local desiderata: *i. (Describability)* Each group should be easily understandable by the analyst. While this is difficult to satisfy through unsupervised clustering of ratings, it is easily enforced in our approach since each group must be formed by rating records of users that share at least one attribute value, which is used to describe that group. *ii. (Size)* Returning groups that contain too few rating records is not meaningful to the analyst. We hence need to impose a minimum size constraint on groups.

Global desiderata: *i. (Coverage)* Together, returned groups should cover most input rating records. While ideally we would like each input record to belong to at least one group, that is not always feasible due to other local and global desiderata associated with the set of returned groups. *ii. (Diversity)* Returned groups need to be different from each other in order to provide complementary information on users. *iii. (Rating Distribution)* Ratings in selected groups should follow a requested distribution (e.g., homogeneity). *iv. (Number of groups)* The number of returned groups should not be too high in order to provide the analyst with an at-a-glance understanding of the data.

A candidate solution is a group-set that verifies all above desiderata. Finding such a group-set is a hard problem because of two reasons. First the pool of candidate group-sets is very large as any possible combination of attribute value pairs can form a group, and any number of groups can form a group-set. By having only 20 attribute value pairs, we end up with $1,048,575$ groups (i.e., $(2^{20}) - 1$) and over 10^{12} group-sets of size 5 (i.e., $1,048,575$ choose 5). The second reason of hardness is that diversity, coverage and rating distribution are conflicting objectives (Sect. 5.1), i.e., optimizing one does not necessarily lead the best values for others. Thus the need for a Multi-Objective optimization approach that will not compromise one objective over another. Such an approach would return *the set of all candidate group-sets* that are not dominated by any other along all objectives.

In this paper, we propose α-MOMRI, an α-approximation algorithm for user group discovery that considers local and global desiderata and guarantees to find group-sets that are α-far from optimal ones. Since α-MOMRI relies on an exhaustive search in the space of all groups, we propose h-MOMRI, a heuristic that exploits the lattice formed by user groups and prunes exploration in order to speed up group-set discovery. Both our algorithms admit a set of rating records of the form $\langle i, u, s \rangle$ and a constrained Multi-Objective optimization formulation [5] and return group-sets that satisfy the formulation and are not dominated by any other group-set. The contributions of this paper are as follows.

1. We formalize specific quality dimensions (coverage, diversity and rating distribution) which we find to be the most natural for discovering user groups on the Social Web. We exploit the semantics of these objectives to go beyond a generic approach.
2. We formalize the problem of discovering user groups as a constrained Multi-Objective optimization problem with quality dimensions as objectives.
3. We develop α-MOMRI, an α-approximation algorithm for user group discovery. Returned group-sets are instances of Pareto plans and are guaranteed to be α-far from optimal ones.
4. We develop h-MOMRI, a heuristic-based algorithm that exploits the lattice formed by user groups to speed up group discovery.
5. In an extensive set of experiments on MovieLens and BookCrossing datasets, we analyze different solutions of α-MOMRI and h-MOMRI and show that high quality group-sets are returned by our approximation and very good response time is achieved by our heuristic.

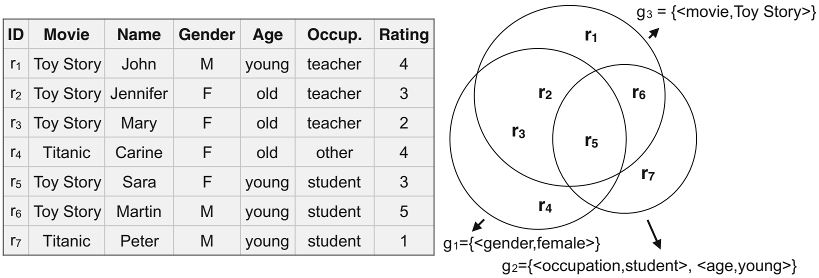
2 Data Model and Preliminaries

We model our database $\mathcal{D}$ as a triple $\langle \mathcal{I}, \mathcal{U}, \mathcal{R} \rangle$, representing the sets of items, reviewers and rating records respectively. Each rating record $r \in \mathcal{R}$ is itself a triple $\langle i, u, s \rangle$, where $i \in \mathcal{I}$, $u \in \mathcal{U}$, and s is the integer rating that reviewer u has assigned to item i. The values of s are application-dependent and do not affect our model.

$\mathcal{I}$ is associated with a set of attributes, denoted as $\mathcal{I}_A = \{ia_1, ia_2, \dots\}$, and each item $i \in \mathcal{I}$ is a tuple with $\mathcal{I}_A$ as its schema. In other words, $i = \langle iv_1, iv_2, \dots \rangle$, where each iv_j is a set of values for attribute ia_j. For example, for the movie *Kazaam* (1996) in MovieLens dataset, the set of attribute values are $\langle$`Paul M. Glaser`, $\{$`Comedy`, `Fantasy`$\}\rangle$ for the attribute schema $\langle$`director`, `genre`$\rangle$. Note that the attribute `genre` is multi-valued. We also have the schema $\mathcal{U}_A = \{ua_1, ua_2, \dots\}$ for reviewers, i.e., $u = \langle uv_1, uv_2, \dots \rangle \in \mathcal{U}$, where each uv_j is a value for attribute ua_j. As a result, each rating record, $r = \langle i, u, s \rangle$, is a tuple, $\langle iv_1, iv_2, \dots, uv_1, uv_2, \dots, s \rangle$, that concatenates the tuple for i, the tuple for u, and the numerical rating score s. The set of all attributes is denoted as $A = \{a_1, a_2, \dots\}$. We now define the notion of user group.

Definition 1 (User Group). *A group g is a set of rating records $\langle u, i, s \rangle$ described by a set of attribute value pairs shared among the reviewers and the items of those rating records. The description of a group g is defined as $\{\langle a_1, v_1 \rangle, \langle a_2, v_2 \rangle, \dots\}$ where each $a_i \in A$ (set of all attributes) and each v_i is a set of values for a_i. By $|g|$, we denote the number of rating records contained in g.*

For instance, the first group in Example 1, $g = \{\langle$`gender`, `female`$\rangle$, $\langle$`location`, `DC`$\rangle$, $\langle$`genre`, `romance`$\rangle\}$ contains rating records in MovieLens for romance movies whose reviewers are all females in DC. Note that is it au-naturel to combine item attributes (genre) and user attributes (location and gender) together. Figure 1 illustrates an example dataset with 7 rating records. The user

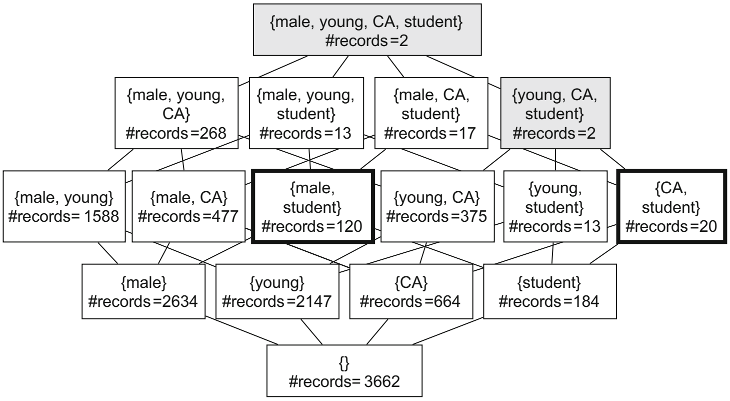


ID	Movie	Name	Gender	Age	Occup.	Rating
r_1	Toy Story	John	M	young	teacher	4
r_2	Toy Story	Jennifer	F	old	teacher	3
r_3	Toy Story	Mary	F	old	teacher	2
r_4	Titanic	Carine	F	old	other	4
r_5	Toy Story	Sara	F	young	student	3
r_6	Toy Story	Martin	M	young	student	5
r_7	Titanic	Peter	M	young	student	1

Fig. 1. Example dataset and group-set

group g_1 is for female reviewers with 4 rating records, and g_2 is for young students with 3 rating records. There exists one record in common between two mentioned user groups (r_5). Note that a user group differs from a *where-clause* SQL query, since our objectives and constraints are not expressible as SQL predicates.

Given a rating record $r = \langle v_1,\ v_2\ \ldots,\ v_k, s\rangle$, where each v_i is a set of values for its corresponding attribute in the schema A, and a group $g = \{\langle a_1, v_1\rangle,\ \langle a_2, v_2\rangle, \ldots, \langle a_n, v_n\rangle\}, n\ \leq\ k$, we say that g covers r, denoted as $r \lessdot g$, iff $\forall i\ \in\ [1, n]$, $\exists r.v_j$ such that v_j is a set of values for attribute $g.a_i$ and $g.v_j \subseteq r.v_i$. For example, the rating $\langle \texttt{female}, \texttt{DC}, \texttt{student}, 4\rangle$ is covered by the group $\{\langle \texttt{gender}, \texttt{female}\rangle,\ \langle \texttt{location}, \texttt{DC}\rangle\}$.



{male, young, CA, student} #records=2
{male, young, CA} #records=268
{male, young, student} #records=13
{male, CA, student} #records=17
{young, CA, student} #records=2
{male, young} #records= 1588
{male, CA} #records=477
{male, student} #records=120
{young, CA} #records=375
{young, student} #records=13
{CA, student} #records=20
{male} #records=2634
{young} #records=2147
{CA} #records=664
{student} #records=184
{} #records= 3662

Fig. 2. Partial lattice for the movie *Toy Story*

Similarly to data cubes, the set of all possible groups form a lattice where nodes correspond to groups and edges correspond to parent/child and ancestor/descendant relationships. A partial lattice for rating records of the movie *Toy Story* (1995) is illustrated in Fig. 2 where we have four reviewer attributes to analyze: `gender`, `age`, `location` (CA stands for California) and `occupation`.

For simplicity, exactly one distinct value per attribute is shown in the Figure. The complete lattice contains 15,582 attribute-value combinations.

2.1 Group Quality Dimensions

We now define three quality dimensions for groups, i.e., coverage, diversity and rating distribution. We are given a set of rating records $R \subseteq \mathcal{R}$ and a group-set G.

Coverage is a value between 0 and 1 and measures the percentage of rating records in R contained in groups in G.

$$coverage(G, R) = | \cup_{g \in G} (r \in R, r \lessdot g)|/|R| \quad (1)$$

For instance, in Fig. 1, $coverage(G, R) = 0.8$ where $G = \{g_1, g_2\}$ and R contains rating records for the movie *Toy Story*.

Diversity is a value between 0 and 1 that measures how distinct groups in group-set G are from each other. Diversity penalizes group-sets containing overlapping groups. To prioritize groups with few overlaps, the overlapping penalty is considered as an exponentiation with a negative exponent.

$$diversity(G, R) = 1/(1 + \Sigma_{g_1, g_2 \in G}|r \in R, r \lessdot g_1 \wedge r \lessdot g_2|) \quad (2)$$

For instance, in Fig. 1, $diversity(G, R) = 0.5$.

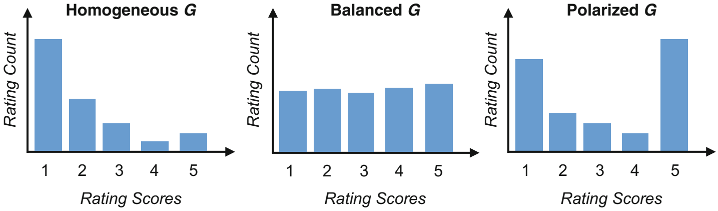


Fig. 3. Different rating distributions for a group-set

Rating Distribution. A group-set G may be characterized by its rating distribution. Figure 3 illustrates some examples of distributions. A rating distribution is a function over the set of ratings in the rating records of groups in G. Equation 3 shows an example of such a function which computes the average *diameter* of ratings. Other aggregation functions could be defined.

$$diameter(G) = avg_{g \in G}(max_{r \in g}(r.s) - min_{r' \in g}(r'.s)) \quad (3)$$

The two most common rating distributions are groups whose members have a consensus (homogeneous distribution, Fig. 3 left), and groups whose members have very different points of view (polarized distribution, Fig. 3 right). A small value of $diameter(G)$ leads a homogeneous group-set G and a high value leads a polarized group-set G. In Fig. 1, $diameter(G) = 3$.

The *diameter* function can capture homogeneity and polarization, but not some other distributions such as "balanced". A detailed discussion on different functions for capturing rating distributions is provided in our technical report [12].

2.2 Multi-Objective Optimization Principles

We propose to use the quality dimensions (coverage, diversity and rating distribution) defined as optimization objectives. When dealing with more than one dimension to optimize, there may be many incomparable group-sets. For instance, for a set of ratings R, we can form two group-sets, G_1 with $\mathit{coverage}(G_1, R) = 0.8$ and $\mathit{diversity}(G_1, R) = 0.4$ and G_2 with $\mathit{coverage}(G_2, R) = 0.5$ and $\mathit{diversity}(G_2, R) = 0.7$. Each group-set has its own advantage: the former has higher coverage and the latter has higher diversity. Another group-set G_3 with $\mathit{coverage}(G_3, R) = 0.5$ and $\mathit{diversity}(G_3, R) = 0.2$ has no advantage compared to G_1, hence it can be ignored. In other words, G_3 is dominated by G_1. In this section, we borrow the terminology of Multi-Objective optimization [5] and define these concepts.

Definition 2 (Plan). *Plan p_i, associated to a group-set G_i for a set of rating records $R \subseteq \mathcal{R}$, is a tuple*
$\langle |G_i|, coverage(G_i, R), diversity(G_i, R), diameter(G_i) \rangle$.

Definition 3 (Sub-plan). *Plan p_i is the sub-plan of another plan p_j if their associated group-sets satisfy $G_i \subseteq G_j$.*

Definition 4 (Dominance). *Plan p_1 dominates p_2 if p_1 has* better *or equivalent values than p_2 in every objective. The term* "better" *is equivalent to* "greater" *for* maximization *objectives (e.g., diversity, coverage and polarization), and* "lower" *for* minimization *ones (e.g., homogeneity). Furthermore, plan p_1 strictly dominates p_2 if p_1 dominates p_2 and the values of objectives for p_1 and p_2 are not equal.*

Definition 5 (Pareto Plan). *Plan p is Pareto if no other plan strictly dominates p. The set of all Pareto plans is denoted as $\mathcal{P}$.*

3 Problem Definition

We define our constrained Multi-Objective optimization problem as follows: for a given set of rating records R and integer constants σ and k (number of groups), the problem is to identify all group-sets, such that each group-set G satisfies:

- $\mathit{coverage}(G, R)$ is maximized;
- $\mathit{diversity}(G, R)$ is maximized;
- $\mathit{rDistb}(G)$ is optimized;
- $|G| \leq k$;
- $\forall g \in G : |g| \geq \sigma$.

Note that our problem focuses on *group-sets* in opposition to *individual groups*, which is a clear distinction from the literature. The last constraint in our problem states that a group g should contain at least σ rating records, an application-defined threshold. For example, if we fix σ to 10 rating records, the groups highlighted in gray in Fig. 2 will not be returned. Note that while we always maximize coverage and diversity, we may either minimize (e.g., in case of homogeneity) or maximize (e.g., in case of polarization) the diameter based on the analyst's needs. We state the complexity of our problem as follows.

Theorem 1. *The decision version of our problem is NP-Complete.*

Proof (sketch). It is shown in [4] that a single-objective optimization problem for user group discovery is NP-Complete by a reduction from the Exact 3-Set Cover problem (EC3). There, homogeneity is maximized and a threshold on coverage is satisfied. In our case, two new conflicting dimensions (diversity and coverage) are added. This means that the problem in [4] is a *special case* of ours, hence our problem is obviously harder. □

4 Algorithm

The main challenge in designing an algorithm for user group discovery, is the Multi-Objective nature of the problem. A Multi-Objective problem can be easily solved if it is possible to combine all objective dimensions into a single dimension (scalarization), or if optimizing one dimension leads an optimized value for other dimensions.

Both following transformations are infeasible for our problem because our objectives are *conflicting*, i.e., optimizing one does not necessarily lead to an optimized value for others (Sect. 5.1). For instance, a group-set may cover almost all input rating records but contains highly overlapping groups thereby hurting its diversity.

In this paper, we discuss 3 different algorithms for our problem: exhaustive, approximation and heuristic.

4.1 Exhaustive and Approximation Algorithms

The exhaustive algorithm starts by calculating Pareto plans for single groups. Then it iteratively calculates plans for group-sets containing more than one group by combining single groups. At each iteration, dominated plans are discarded. The algorithm combines sub-plans to obtain new plans and exploits the *optimality principle* (POO) for pruning [15]. This approach makes an exhaustive search over all combinations of user groups to find Pareto plans, i.e., both time and space consuming [6].

We propose to improve the complexity of the exhaustive algorithm with our *approximation-based* algorithm which makes less enumerations and guarantees the quality of results. Another way of improvement is *heuristic-based* which will be discussed in Sect. 4.2. For our approximation algorithm, we exploit the near-optimality principle (PONO) [15].

Definition 6 (PONO). *Given a maximization objective f (e.g., diversity, coverage, polarization) and $\alpha \geq 1$, let p_1 be a plan with sub-plans p_{11} and p_{12}. Derive*

Algorithm 1. α-approximation MOMRI (α-MOMRI)

```
Input: $\sigma, k, \alpha > 1, R$
Output: Pareto result set $\mathcal{P}_\alpha$
$_\alpha \leftarrow \emptyset$
for all user groups $g$ whose size is at least $\sigma$ do
    $p_g \leftarrow construct\_plan(g)$
    if $p_g$ is not $\alpha$-dominated by any other plan in $\mathcal{P}_\alpha$ then $\mathcal{P}_\alpha.add(p_g)$
end
for $n \in [2, k]$ do
    for group-sets $G$ of size $n$ do
        $p_G \leftarrow construct\_plan(g_G)$
        if $p_G$ is not $\alpha$-dominated by any other plan in $\mathcal{P}_\alpha$ then $\mathcal{P}_\alpha.add(p_G)$
    end
end
return $\mathcal{P}_\alpha$
```

p_2 from p_1 by replacing p_{11} by p_{21} and p_{12} by p_{22} where p_{21} and p_{22} are subplans of p_2. Then $f(G_{21}) \geq f(G_{11}) \times \alpha$ and $f(G_{22}) \geq f(G_{12}) \times \alpha$ together imply $f(G_2) \geq f(G_1) \times \alpha$. The extension for a minimization objective is straightforward.

We have formally proved that all our objectives satisfy PONO. Proofs are provided in our technical report [12]. Note that among different definitions in the literature for coverage, diversity and rating distribution, we picked the ones that are most intuitive to our problem and that satisfy PONO. For instance, the rating distribution function in [4] does not satisfy PONO.

PONO overrides POO. Thus a new notion of dominance is introduced in Definition 7 to be in line with PONO.

Definition 7 (Approximated Dominance). *Let $\alpha \geq 1$ be the precision value, a plan p_1 α-dominates p_2 if for every maximization objective f (e.g., diversity, coverage, polarization), $f(G_1) \geq f(G_2) \times \alpha$. The extension for a minimization objective is straightforward.*

Definition 8 (Approximated Pareto Plan). *For a precision value α, plan p is an α-approximated Pareto plan if no other plan α-dominates p.*

Generating fewer plans makes a Multi-Objective optimization algorithm run faster [15]. This is because the execution time heavily depends on the number of generated plans. Thus a pruning strategy dictated by PONO is at the core of the α-MOMRI algorithm illustrated in Algorithm 1. In the special case of $\alpha = 1$, the algorithm operates exhaustively. If $\alpha > 1$, the algorithm prunes more and hence is faster. In the latter case, a new plan is only compared with all plans that generate the same result. But a new plan are only inserted into the buffer if no other plan approximately dominates it. This means that α-MOMRI tends to insert fewer plans than the exhaustive algorithm. Note that α-MOMRI

Algorithm 2. Heuristic MOMRI (h-MOMRI)

```
   Input: $\sigma, k, \alpha, R$
   Output: Result set $\mathcal{P}_h$
1  $\mathcal{P}_h \leftarrow \emptyset$
2  $\mathcal{N} \leftarrow$ Set of intervals on diversity values
3  for $n$ times do
4      $G_s \leftarrow random\_groupset(k, \sigma)$
5      $G_s^* \leftarrow SHC(G_s)$
6      $interval \leftarrow get\_interval(G_s^*)$
7      $\mathcal{N}[interval].add(G_s^*)$
8  end
9  for $interval \in \mathcal{N}$ do
10     Keep non-dominated plans in interval and add them to $\mathcal{P}_h$
11 end
12 $\mathcal{P}_h \leftarrow optimize\_diameter(\mathcal{P}_h)$
13 return $\mathcal{P}_h$
```

is objective-independent. In the future, we plan to extend the scope of group discovery to other objectives (as listed in [7]).

4.2 Heuristic Algorithm

A heuristic algorithm has obviously its own advantages and disadvantages. Of course a heuristic algorithm does not provide any approximation guarantee. Eventually, it returns a subset of Pareto set. Nevertheless, the fact that it generates a subset of Pareto makes it faster.

Algorithm 2 illustrates our heuristic algorithm. The algorithm starts by making n different iterations on finding optimal points to avoid local optima (lines 3 to 8). At each iteration, the algorithm begins with a random group-set G_s with k groups whose size is at least σ (line 4). Then a *Shotgun Hill Climbing* [14] local search approach (*SHC*) is executed (Algorithm 3) to find the group-set with optimal value starting from G_s (line 5). *SHC* maximizes coverage. Diversity is already divided into intervals $\mathcal{N}$ for each of which a buffer is associated. The resulting group-set of *SHC* is placed in the buffer whose interval matches the diversity value of the group-set (line 7). Finally, n different solutions are distributed in different interval buffers. The algorithm then iterates over interval buffers to prune dominated plans (lines 9 to 11). Based on Definition 4, a plan is pruned and removed from its buffer if it is dominated by other plans. Finally, for each interval, we report one unique solution that has the maximum/minimum value for diameter based on the requested distribution (line 12).

SHC operates on a generalization/specialization lattice of groups (as in Fig. 2). Navigation of this lattice in a downward fashion satisfies monotonicity property for coverage: given any two groups g_1 and g_2 where g_1 is the parent of g_2, the coverage of g_1 is no smaller than the coverage of g_2. Note that in a bi-objective context, *SHC* can optimize each one of coverage and diversity.

Algorithm 3. Shotgun Hill Climbing (*SHC*) Algorithm

```
Input: Group-set $G$, $R$
Output: Optimized group-set $G^*$
$G^* \leftarrow \emptyset$
while true do
    $\mathcal{C} \leftarrow \emptyset$
    for $g \in G$ and each lattice-based parent $g'$ of $g$ do
        $G' \leftarrow G - \{g\} + \{g\}'$
        $\mathcal{C}.add(G', coverage(G', R))$
    end
    let $(G'_m, coverage(G'_m, R))$ be the pair with maximum coverage
    if $coverage(G'_m, R) \leq coverage(G, R)$ then
        $G^* \leftarrow G$
        return $G^*$
    end
    $G \leftarrow G'_m$
end
```

However, to benefit from the monotonicity property, we use *SHC* to optimize coverage. *SHC* verifies all local neighbors of a group for an improvement of coverage. If no improvement is achieved, it stops and returns the current group-set. Nevertheless, if we optimize diversity using *SHC*, navigation in the generalization/specialization lattice is nothing but a random walk over the space of groups.

For instance, consider the input group-set $G_s = \{g_1, g_2\}$ where $g_1 = \{\langle$`gender`, `male`$\rangle$, $\langle$`occupation`, `student`$\rangle\}$ and $g_2 = \{\langle$`location`, `CA`$\rangle$, $\langle$`occupation`, `student`$\rangle\}$. These two groups are marked in bold boxes in Fig. 2. We obtain a coverage of 0.79 for G_s. Keeping g_2 fixed, the resulting combinations by swapping g_1 with its parents are either $g_3 = \{\langle$`gender`, `male`$\rangle\}$ or $g_4 = \{\langle$`occupation`, `student`$\rangle\}$. For instance, the coverage of $G'_s = \{g_2, g_3\}$ is 0.81. As we observe an improvement, we iterate on this new group-set G'_s to improve coverage.

A detailed discussion on complexity analysis of our proposed algorithms is provided in our technical report [12].

5 Experiments

In this section, we first validate the need for Multi-Objective optimization. Then we compare α-MOMRI and h-MOMRI on the quality of returned groups and the scalability of those algorithms.

We consider two different rating datasets for our study: MovieLens and BookCrossing. Due to lack of space, we only show results on MovieLens. An exhaustive set of results is presented in our technical report [12]. Both datasets have approximately the same number of ratings. BookCrossing has one order of magnitude more users and items. We consider a 5-star rating system for both datasets.

Table 1. Input sets of rating records

Profile	Movie in MovieLens
Highest number of ratings	American Beauty
Lowest number of ratings	Celtic Pride
Highest average rating	Sanjuro
Lowest average rating	Kazaam

MovieLens contains four user attributes: `gender`, `age`, `occupation` and `zipcode`. We convert the numeric age into four categorical attribute values, namely `teenager` (under 18), `young` (18 to 35), `middle-age` (35 to 55) and `old` (over 55). There are 21 different occupations listed in MovieLens e.g., student, artist, doctor, lawyer, etc. We convert zipcodes to states in the USA (or to `foreign`, if not in USA) by using the USPS zip code lookup.[3] We also enriched MovieLens by crawling IMDb[4] using the OMDb API[5] to obtain following item attributes: `director`, `writer` and `release year` and `genre`.

We implement our prototype system using JDK 1.8.0. All scalability experiments are conducted on an 2.4 GHz Intel Core i5 with 8 GB of memory on OS X 10.9.5 operating system.

For our experiments, we consider four different sets of input rating records described in Table 1. Each item contains at least 50 ratings. We assume that it is straightforward to analyze less than 50 ratings, manually. We also fix $\sigma = 10$ as this value is a border line between frequent ratings and the long tail [12].

5.1 Need for Multi-Objective Optimization

What is the added value of Multi-Objective optimization? We compare first MOMRI with MRI [4], a single-objective approach for group discovery which some authors of this work have already proposed. MRI minimizes *diameter* and considers a lower bound on coverage *min_c*. Given a set of rating records R for the movie *American Beauty* in MovieLens, $k = 3$, $min\_c = 0.7$, one of the returned group-sets by MRI is $G_{MRI} = \{g_1, g_2, g_3\}$ where $g_1 = \{\langle \texttt{gender}, \texttt{female}\rangle, \langle \texttt{age}, \texttt{young}\rangle\}$, $g_2 = \{\langle \texttt{occupation}, \texttt{student}\rangle, \langle \texttt{age}, \texttt{young}\rangle\}$ and $g_3 = \{\langle \texttt{gender}, \texttt{male}\rangle, \langle \texttt{occupation}, \texttt{student}\rangle\}$. The objective values for G_{MRI} are as follows: $coverage(G_{MRI}, R) = 0.81$, $diversity(G_{MRI}, R) = 0.03$ and $diameter(G_{MRI}, R) = 0.13$. However, as diversity is not optimized, there exists huge overlap in groups: many young reviewers are also students.

In the same context, one returned group-set by MOMRI is the one we already discussed in Example 1: $G_{MOMRI} = \{g_4, g_5, g_6\}$ where $g_4 = \{\langle \texttt{gender}, \texttt{female}\rangle, \langle \texttt{age}, \texttt{young}\rangle\}$, $g_5 = \{\langle \texttt{age}, \texttt{young}\rangle, \langle \texttt{location}, \texttt{DC}\rangle\}$ and $g_6 = \{\langle \texttt{gender}, \texttt{male}\rangle, \langle \texttt{age}, \texttt{teen} - \texttt{ager}\rangle\}$. The objective values for G_{MOMRI}

[3] http://zip4.usps.com.

[4] http://www.imdb.com.

[5] http://www.omdbapi.com.

are as follows: *coverage*(G_{MOMRI}, R)=0.79, *diversity*(G_{MOMRI}, R)=0.33 and *diameter*$(G_{MOMRI}, R) = 0.11$. This group-set has optimized values on all objectives. Specifically, it has a high diversity as only 2 female reviewers for *American Beauty* are both young and residents of DC. It also shows that *min_c* in MRI is a hard constraint and can easily miss a promising result which has a very high coverage but does not meet the threshold.

We already discussed that consistency of objectives transforms the multi-objective problem into a single-objective one that is trivial to solve (Sect. 4). In this experiment, we verify if our objectives (defined in Sect. 2.1) are consistent. We maximize coverage and observe how values of diversity and diameter evolve. To maximize coverage, we use Algorithm 3. Figure 4 illustrates the results for different sets of input rating records in Table 1. Each point illustrates the objective values for each of 20 runs. Note that this experiment is independent of the heuristic and the approximation algorithms.

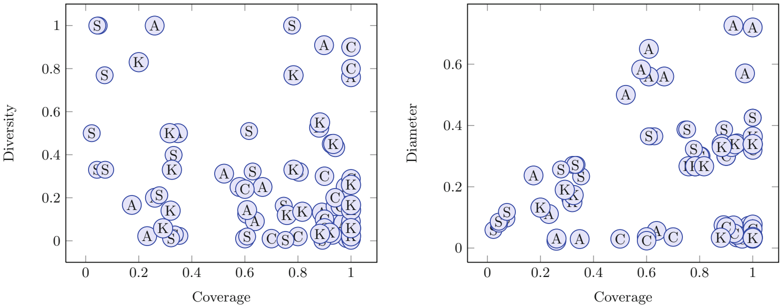


Fig. 4. Conflicting objectives on MovieLens. Movie title initials are illustrated on points.

We observe that in general, no correlation exists between the optimized value of coverage and other objectives. Thus each objective should be optimized independently. The same result was obtained for BookCrossing [12].

5.2 Comparison of Algorithms

In this section, we compare h-MOMRI and α-MOMRI. Our hypothesis is that h-MOMRI has a manageable solution space size compared to α-MOMRI which leads to a reduced execution time.

First we compare the quality of algorithms regarding the dominance of solutions. In Multi-Objective optimization, if for two algorithms X and Y, the majority of X's solutions dominate Y's, it means that X is able to produce solutions with higher quality than Y. In this experiment, we make the same comparison between α-MOMRI and h-MOMRI. For this experiment, we need to compare each pair of α-MOMRI and h-MOMRI solutions. We count the number of times each algorithm dominates the other in pairwise comparison of their results. We consider $\alpha = 1.15$ for α-MOMRI and *nbintervals* $= 40$ for h-MOMRI. We denote the set of α-MOMRI solutions as $\mathcal{P}_\alpha$ and the set of h-MOMRI solutions as $\mathcal{P}_h$. We observe that for all sets of input rating records in Table 1, at least 62 % of solutions in $\mathcal{P}_h$ are dominated by solutions in $\mathcal{P}_\alpha$. This is because α-MOMRI generates the complete set of α-approximated Pareto plans, while h-MOMRI produces a subset. For instance, for the movie *American Beauty*, α-MOMRI produces 16 times more solutions than the heuristic algorithm. Evidently the solutions in $\mathcal{P}_h$ are either as good as $\mathcal{P}_\alpha$'s or worse. Our results show that although α-MOMRI presents a huge set of all Pareto plans, h-MOMRI can return an acceptable representative subset where almost half of solutions are as good as the set $\mathcal{P}_\alpha$.

Now we compare α-MOMRI and h-MOMRI concerning their performance and the number of solutions they produce. We consider 3 different instances for each algorithm: for α-MOMRI, we consider instances with $\alpha = 2$ (A), $\alpha = 1.5$ (B) and $\alpha = 1.15$ (C), and for h-MOMRI, we consider instances with 5 (D), 10 (E) and 40 (F) intervals. We run this experiment with 4 items having the highest amount of rating records as items with fewer records exhibit similar behavior.

Figure 5 illustrates the results. As expected, in general the number of solutions produced by h-MOMRI is one order of magnitude less than α-MOMRI in both datasets. In both algorithms, the number of ratings records play an important role and increases the number of solutions. In [12], it is shown that the time performance of both algorithms is a function of the group space size. A data-centric observation in Fig. 5 reveals that more rating records lead more groups, hence worse performance (which is the case for *American Beauty*).

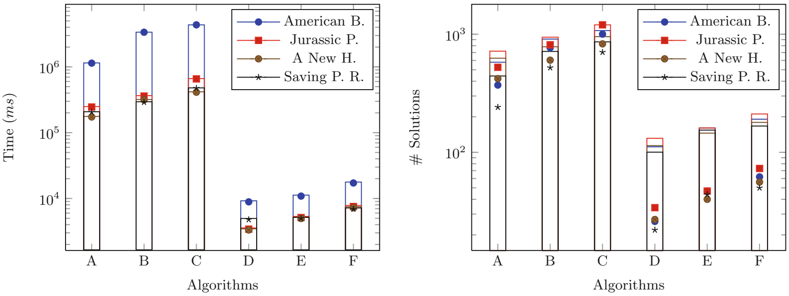


Fig. 5. Comparison of α-MOMRI and h-MOMRI algorithms in execution time (left) and # solutions (right) on MovieLens

Choosing between α-MOMRI and h-MOMRI. Both α-MOMRI and h-MOMRI are useful for analysts in different scenarios. α-MOMRI can be used in an *offline* context to produce an exhaustive set of user groups with a precision defined by α for further analysis. For instance, a movie rating website (like IMDb) can index user groups generated offline and execute various user queries like *'what are interesting groups of female teenagers who have rated romantic* *movies'*. On the other hand, in an *online* or *streaming* context, h-MOMRI is beneficial because it can immediately produce a representative subset of results. For instance, in a movie rating website an analyst can quickly observe interesting user groups of comedy and romantic movies.

6 Related Work

To the best of our knowledge, no approach has proposed and formalized the problem of discovering user groups for collaborative rating datasets by considering multiple *independent* and *conflicting* quality dimensions. Recent studies[6] have shown an interest in reporting statistics about pre-defined groups, as opposed to our work where we look to discover high-quality user groups on the fly. However our work does relate to a number of others in its aim and optimization mechanism.

Multi-Objective Optimization. There exist different approaches to solve a multi-objective problem [15,16]. We already discussed that Scalarization does not work in our case (Sect. 5.1). Another popular method is ϵ-constraints [13] where one objective is optimized and others are considered as constraints. The approach in [4] can be seen as a relaxed ϵ-constraints version of our problem. Another approach is Multi-Level Optimization [11] which needs a meaningful hierarchy between objectives. In our case, all objectives are independent and conflicting, hence using this mechanism is not feasible.

User Group Discovery. User groups can be discovered by clustering methods [1–3,9] where a single objective is optimized. Multi-Objective clustering [8,10] is an improvement where clusters are obtained from n different clustering algorithms. This guarantees clusters with high quality in multiple dimensions. This is a two-step approach where *i*. each clustering algorithm, applied to one quality dimension, generates its own set of clusters, *ii*. a *goodness* measure picks target clusters by combining results of all algorithms. However, the definition of a goodness measure is subjective and does not guarantee that all desired objectives are optimized. Also MOMRI scans data only once as the pruning technique in α-MOMRI considers all objectives at the same time and determines if a candidate group-set should or not be kept for further comparisons. On the other hand, clustering methods often lead to information overload. Using h-MOMRI, the analyst receives a manageable subset of high quality results in a reasonable time. More (precise) results are returned by reducing α for α-MOMRI or increasing *nbintervals* for h-MOMRI.

7 Conclusion and Future Work

In this paper, we investigated the question of finding the best group-sets that characterize a database of rating records of the form $\langle i, u, s\rangle$, where $i \in \mathcal{I}$, $u \in \mathcal{U}$,

[6] http://blog.testmunk.com/how-teens-really-use-apps/.

and s is the integer rating that user u has assigned to item i. We showed that the problem of finding high-quality group-sets is NP-Complete and proposed a constrained Multi-Objective formulation. Our formulation incorporates local and global group desiderata. We proposed two algorithms that find group-sets as instances of Pareto plans. The first one α-MOMRI, is an α-approximation algorithm and the second, h-MOMRI, is a heuristic-based algorithm. Our extensive experiments on MovieLens and BookCrossing datasets show that our approximation finds high quality groups and that our heuristic is very fast without compromising quality.

Our work can be improved in many ways. In particular, we plan to perform an extensive user study to be able to evaluate the quality of returned group-sets. An online poll (about movies or books) could be used to build a ground-truth and will be used to evaluate the usefulness of our group-sets. Also, we plan to investigate an extensive analysis of rating distributions for our algorithms using some dispersion measures.

References

1. Agrawal, R., Gehrke, J., Gunopulos, D., Raghavan, P.: Automatic subspace clustering of high dimensional data for data mining applications. ACM (1998)
2. Agrawal, R., Imielinski, T., Swami, A.N.: Mining association rules between sets of items in large databases. In: SIGMOD (1993)
3. Amiri, B., Hossain, L., Crowford, J.: A multiobjective hybrid evolutionary algorithm for clustering in social networks. In: Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation. ACM (2012)
4. Das, M., Amer-Yahia, S., Das, G., Yu, C.: Mri: meaningful interpretations of collaborative ratings. In: VLDB (2011)
5. Dutot, P.F., Rzadca, K., Saule, E., Trystram, D.: Multi-objective Scheduling, chap. 9. Chapman and Hall/CRC Press (2009)
6. Ganguly, S., Hasan, W., Krishnamurthy, R.: Query optimization forparallel execution, vol. 21. ACM (1992)
7. Geng, L., Hamilton, H.J.: Interestingness measures for data mining: a survey. ACM Comput. Surv. (CSUR) **38**(3), 9 (2006)
8. Jiamthapthaksin, R., Eick, C.F., Vilalta, R.: A framework for multi-objective clustering and its application to co-location mining. In: Huang, R., Yang, Q., Pei, J., Gama, J., Meng, X., Li, X. (eds.) ADMA 2009. LNCS (LNAI), pp. 188–199. Springer, Heidelberg (2009). doi:10.1007/978-3-642-03348-3_20
9. Kargar, M., An, A., Zihayat, M.: Efficient bi-objective team formation in social networks. In: Flach, P.A., Bie, T., Cristianini, N. (eds.) ECML PKDD 2012. LNCS (LNAI), pp. 483–498. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33486-3_31
10. Law, M.H., Topchy, A.P., Jain, A.K.: Multiobjective data clustering. In: Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2004, vol. 2, pp. II–424. IEEE (2004)
11. Migdalas, A., Pardalos, P.M., Värbrand, P.: Multilevel optimization: algorithms and applications, vol. 20. Springer Science & Business Media (1997)
12. Omidvar-Tehrani, B., Amer-Yahia, S., Dutot, P.F., Trystram, D.: Multi-objective group discovery on the social web. Research Report RR-LIG-052, LIG, Grenoble, France (2016)

13. Papadimitriou, C.H., Yannakakis, M.: On the approximability of trade-offs and optimal access of web sources. In: FOCS (2000)
14. Russell, S.J., Norvig, P.: Probabilistic reasoning. Artificial intelligence: a modern approach (2003)
15. Trummer, I., Koch, C.: Approximation schemes for many-objectivequery optimization. In: Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data. ACM (2014)
16. Tsaggouris, G., Zaroliagis, C.: Multiobjective optimization: Improved fptas for shortest paths and non-linear objectives with applications. Theory Comput. Syst. **45**(1), 162–186 (2009)