Recommender systems with selfish users

Maria Halkidi & Iordanis Koutsopoulos

Knowledge and Information Systems
An International Journal

ISSN 0219-1377

Knowl Inf Syst
DOI 10.1007/s10115-020-01460-5

Springer

Knowledge and Information Systems
https://doi.org/10.1007/s10115-020-01460-5

REGULAR PAPER

Recommender systems with selfish users

Maria Halkidi[1] · **Iordanis Koutsopoulos**[2]

Received: 17 September 2018 / Revised: 6 March 2020 / Accepted: 7 March 2020

Abstract
Recommender systems are a fundamental component of contemporary social-media platforms and require feedback submitted from users in order to fulfill their goal. On the other hand, the raise of advocacy about user-controlled data repositories supports the selective submission of data by user through intelligent software agents residing at the user end. These agents are endowed with the task of protecting user privacy by applying a "soft filter" on personal data provided to the system. In this work, we pose the question: "how should the software agent control the user feedback submitted to a recommender system in a way that is most privacy preserving, while the user still enjoys most of the benefits of the recommender system?". We consider a set of such agents, each of which aims to protect the privacy of its serving user by submitting to the recommender system server a version of her real rating profile. The fact that issued recommendations to a user depend on the collective rating profiles by all agents gives rise to a novel game-theoretic setup that unveils the trade-off between privacy preservation of each user and the quality of recommendation they receive. Privacy is quantified through a distance metric between declared and an "initial" random rating profile; the latter is assumed to provide a "neutral" starting point for the disclosure of the real profile. We allow different users to have different perception of their privacy through a user-dependent utility function of this distance. The quality of recommendations for each user depends on submitted ratings of all users, including the ratings of the user to whom the recommendation is provided. We prove the existence of a Nash equilibrium point (NEP), and we derive conditions for that. We show that user strategies converge to the NEP after an iterative best-response strategy update sequence that involves circulation of aggregate quantities in the system and no revelation of real ratings. We also present various modes of user cooperation in rating declaration, by which users mutually benefit in terms of privacy. We evaluate and compare cooperative and selfish strategies in their performance in terms of privacy preservation and recommendation quality through real movie datasets.

Keywords Recommender systems · Game theory · Optimization

✉ Maria Halkidi
mhalk@unipi.gr

Iordanis Koutsopoulos
jordan@aueb.gr

[1] Department of Digital Systems, University of Piraeus, Piraeus, Greece

[2] Department of Informatics, Athens University of Economics and Busines, Athens, Greece

Published online: 28 March 2020

1 Introduction

Recommender systems are basic ingredients of contemporary social-media platforms [1,2]. The functionality of these systems relies on user-related feedback information that is received by the platform in the form of user records (e.g., book purchase, hotel stay, movie watch), binary preferences (like/not like) or rating of items or services [3]. More often than not, this information is provided by users on a voluntary basis, and sometimes it is abstracted as the preference profile of the user. User profiles are then aggregated by the system, and they are used as input to the recommender system algorithm that provides personalized recommendations to users about items to view or purchase.

The efficiency of a recommender system amounts to providing high-quality personalized recommendations to different users. The quality of recommendations for an individual user relies on the participation of other users in the rating process. Furthermore, even if that is not immediately apparent, each user can, to a certain extent, affect the quality of recommendations for herself by her own ratings and degree of participation in the process. To see this, consider the following example with two users, 1 and 2, where user 1 has viewed and rated item A, and user 2 has viewed and rated item B. Suppose that the recommender system recommends item B to user 1 if a certain metric pertaining to item B exceeds a threshold, otherwise it does not. This metric will depend on (i) how high the rating of user 2 for item B is, (ii) how similar item B is to item A, based on some similarity metric, (iii) how high the rating of user A is for item 1. Clearly, *whether or not item B will be recommended to user 1 depends on the ratings of both users for the items they have viewed.*

User feedback information is necessary to reap the benefits of recommender systems. However, this raises also substantiated concerns regarding the privacy of users, in the sense that certain aspects of the user profile are sensitive enough. Some users follow an extreme behavior so that they do not provide any information about their profile to the system and thus they receive very low-quality or no recommendations in the end [4].

With the raise of advocacy about personally controlled data spaces and repositories, there comes the perspective of intelligent software agents residing at the user end. These agents are endowed with the task of protecting user privacy by applying a "soft filter" on personal data provided to the system, while still reaping most of the benefits of the recommender system's service. This is the scenario we study in this paper. We consider a set of such agents, each of which aims to protect the privacy of its serving user in the best possible way by submitting to the recommender system server a distorted version of her real rating profile. We note that there exist other more drastic approaches to protect one's privacy, such as eliminating a rating from the rating profile. However, it is our intention here to study the impact of softer approaches to privacy protection first.

Our contribution Since recommendations to a user depend on the collective rating profiles by all agents, a novel game-theoretic setup arises. We address the fundamental trade-off between privacy preservation and high-quality recommendation, and the way it is shaped through selfish or collaborative profile perturbation strategies by users. Users submit their ratings about items they have viewed or experienced to the recommendation server. The server aggregates ratings and generates personalized recommendations for each user. We study the class of recommender systems in which the quality of recommendations to each user i depends on submitted rating profiles from all users, including user i. For example, recommender systems that are based on collaborative filtering (CF) (memory- or model-based methods) or hybrid recommender systems that combine collaborative filtering and content-based approaches fall within that class. Each user aims at declaring a rating profile

sufficiently far away from her true one so as to preserve her privacy as much as possible. Declaring a random profile the user is considered to maximize her privacy preservation. At the same time, she would like to incur a received recommendation as close as possible to the one she would get if she revealed her true rating profile. The contributions of our work to the literature are as follows:

- We model privacy through a distance metric between declared and an initial random rating profile; the latter is considered as an initial neutral profile for disclosure the real profile of users. We allow different users to have different perception of their privacy through a user-dependent utility function of this distance. We also model the quality of the recommendation through a vector distance-based model between the recommendations the user would get if she declares true profile or not. Having modeled that, we then define the strategy space of a user in terms of her declared rating profile to the recommendation server.
- We use game theory to model and study the interaction of users. We show the existence of a Nash equilibrium point (NEP), and we derive conditions and expressions of that. We also show that user strategies converge to the NEP after an iterative best-response strategy update sequence that involves circulation of aggregate quantities in the system and no revelation of real ratings.
- We present different modes of user cooperation through which users cooperate in declaring their profile so as to mutually benefit in terms of privacy.
- We use real movie rating datasets to evaluate the performance of the game-theoretic approach in terms of privacy preservation and recommendation quality, and we compare the performance of cooperative and competitive profile declaration strategies.

This paper and its preliminary conference version [5] are among the very first ones that apply game theory to address user interaction and selfishness in privacy preservation in recommender systems and to quantify the benefits of user cooperation versus competition. While game theory has been used in various contexts [6], only few works ([7,8]) leverage game theory in recommender systems. The work in [7] studies the game between users where each user applies a probability distribution over the possible ways in which a user may choose to rate items she has experienced. The IRGAN framework [8] aims to unify generative and discriminative models, via a minimax game in order to retrieve documents and predict their relevancy to a given query. Compared to that works, we explicitly introduce privacy objectives and metrics in order to capture the trade-off between recommendation quality and privacy preservation. We generalize the applicability of the approach to a larger class of recommender systems, we adhere to deterministic user strategies that might be considered closer to real life, and we propose cooperative policies as well.

The rest of the paper is organized as follows. In Sect. 2, we present the model and define the problem, and in Sect. 3, we exemplify the model for a CF recommender system that is used in our approach. In Sect. 4, we present an analysis of selfish and cooperative profile declaration policies. Section 5 presents results from real movie-rating datasets. In Sect. 6, we discuss related state-of-the-art works, and in Sect. 7, we conclude our study.

2 Model and problem statement

2.1 Ratings and recommendation

Consider a set $\mathcal{U}$ of N users and a set $\mathcal{I}$ of items available for recommendation. Each user i has already viewed, purchased or experienced a small subset of items $\mathcal{S}_i \subset \mathcal{I}$, where $|\mathcal{S}_i| << |\mathcal{U}|$, where $|\mathcal{A}|$ denotes the cardinality of set $\mathcal{A}$. We denote the vector of ratings of user i for the items it has viewed by $\mathbf{p}_i = (p_{ik} : k \in \mathcal{S}_i)$, where p_{ik} is the rating of user i for item $k \in \mathcal{S}_i$. Without loss of generality, we assume that p_{ik} is positive, continuous-valued and upper bounded, i.e., $0 \leq p_{ik} \leq P$. The vector of ratings $\mathbf{p}_i$ is private information for each user i, and we refer to it as the *private profile* or *private ratings vector* of user i. Thus, the private profile consists of the identities of viewed items and their ratings.

Each user i declares a rating for each item $k \in \mathcal{S}_i$ to the recommendation server. Let $\mathbf{q}_i = (q_{ik} : k \in \mathcal{S}_i)$ be the vector of *declared* ratings of user i, where q_{ik} is the declared rating of user i for item $k \in \mathcal{S}_i$. Thus, $\mathbf{q}_i$ can be different from $\mathbf{p}_i$. We refer to $\mathbf{q}_i$ as the *declared profile* or the *declared ratings vector* of user i. The declared profile consists of the same items as the private profile, i.e., $\mathbf{q}_i$ includes only items $k \in \mathcal{S}_i$. It is defined based on the disclosure strategy that the user has specified.

In this work, we adopt a privacy model similar to the one presented in [9]. According to this model, each user i selects an initial profile $\mathbf{t}_i$ that wishes to impersonate in case there is high tolerance to the recommendation error. The initial profile will provide a "neutral" starting point for the disclosure of the private user profile $\mathbf{p}_i$. For example, a user may want to exhibit very common interests, and therefore, t_i might be the average profile of the population or it might be the neutral value of the considered rating system. Then, the declared profile of user $\mathbf{q}_i$ coincides with the initial $\mathbf{t}_i$. However, the user may select to compromise her privacy and reveal her private profile in order that her recommendation requirements are satisfied.

Our disclosure mechanism reveals the deviation of the user's *"neutral" profile* to the private one. Given the disclosure strategy specified by a user i, $\boldsymbol{\delta}_i = (\delta_{ik} : k \in \mathcal{S}_i)$, we define the user's declared profile $\mathbf{q_i}$ as the convex combination:

$$\mathbf{q}_i = (1 - \boldsymbol{\delta}_i) \cdot \mathbf{t}_i + \boldsymbol{\delta}_i \cdot \mathbf{p}_i \tag{1}$$

where $0 \leq \boldsymbol{\delta}_i \leq 1$.

The recommendation server is the repository of all ratings submitted by users. It collects declared user profiles, and it is responsible for issuing personalized recommendations to users. Let $\mathbf{P} = (\mathbf{p}_i : i \in \mathcal{U})$ be the ensemble of private ratings of users, and let $\mathbf{Q} = (\mathbf{q}_i : i \in \mathcal{U})$ be the ensemble of declared ratings of all users to the server. When the server needs to give a recommendation to user i, it takes into account the ensemble of ratings $\mathbf{Q}$ to compute a recommendation vector that includes ratings for items that user i has not viewed. Let $\mathbf{r}_i = (r_{i\ell} : \ell \notin \mathcal{S}_i)$ be the *recommendation vector* for user i.

The recommendation server takes all user ratings into account and employs a generic mapping $f_i(\cdot)$ to compute the recommendation vector for each user i. We denote the dependence of the recommendation for user i on declared ratings of all users, as $\mathbf{r}_i = f_i(\mathbf{Q}) = f_i(\mathbf{q}_1, \ldots, \mathbf{q}_N)$. Here, we implicitly assume that all users' ratings are taken into account. Usually, the recommendation server may take into account only a subset of users and their ratings in order to compute the recommendation for user i. In this work, we are not concerned with designing the mapping $f_i(\cdot)$. Instead, we assume that a given mapping is employed by the server, and this mapping is known to users. In the type of recommender systems we consider, the personalized recommendation $\mathbf{r}_i$ depends also on the rating vector $\mathbf{q}_i$ of user i.

Next, the recommendation vector is fed back to user i in a form that depends on the specific recommendation system. In general, part of the vector $\mathbf{r}_i$ is returned to user i. For instance, the server may return just one item, e.g., the one with highest computed rating $r_{i\ell}$ among those in set $\{\ell : \ell \notin \mathcal{S}_i\}$, or in general it may return the M highest-rated items from the set above. Without loss of generality, we assume that the entire vector of ratings $\mathbf{r}_i$ is returned to user i, possibly reordered, so that the highest-rated components appear first.

2.2 Privacy metric

For each user i, we define a metric that quantifies *privacy protection* or *preservation* for i. Privacy depends on the relative similarity between the private profile ($\mathbf{q}_i$) and the initial profile of user i ($\mathbf{t}_i$), where similarity may be measured with different types of measures (e.g., negative Euclidean, negative Mahalanobis, etc). We denote this similarity by a continuous function $g(\mathbf{t}_i, \mathbf{q}_i)$.

In general, different users may value privacy differently. In order to capture this differentiation, we define a continuous and non-decreasing utility function $u_i(g(\cdot))$ for each user i. Depending on the predisposition of users to privacy, $u_i(\cdot)$ may be concave, linear or convex function of its argument. For example, a convex function signals increasing privacy returns for the same unit deviation in privacy metric $g(\cdot)$. Thus, a user would feel happier in terms of privacy if his ratings changed say from 5 to 3 than if it was changed from 1 to 3. For instance, the function $u(\cdot)$ that quantifies privacy preservation for user i is taken to be,

$$u(g(\mathbf{t}_i, \mathbf{q}_i)) = -\sum_{k \in \mathcal{S}_i} p_{ik}(t_{ik} - q_{ik})^2 . \tag{2}$$

The metric above reflects the intuitive fact that privacy preservation increases as the similarity, $g(\mathbf{t}_i, \mathbf{q}_i)) = -\sum_{k\in\mathcal{S}_i}(t_{ik} - q_{ik})^2$ between the initial and the declared profiles increases. This distance is weighted by the private rating p_{ik} so as to capture the fact that, among items whose private and declared ratings have the same distance, it is preferable from a privacy preservation perspective to change the rating of items that are higher rated in the private rating profile.

Functions $u_i(\cdot)$ may be constructed through input provided by the user either by means of a priori settings of related web or mobile applications or via questionnaires.

2.3 Recommendation quality

Users would like to get good-quality recommendations for items that they have not viewed. The recommendation that a specific user receives depends on declared profiles of other users to the server, but also on the declared profile of this specific user. Even if a user declares her true private profile, the recommendations she would get would still depend on declared profiles of other users. Thus, the user may still receive sub-optimal recommendations while also compromising her privacy. The problem for each user i is to determine her disclosure strategy $\boldsymbol{\delta}_i$ so as to protect her privacy, while at the same time the quality of the recommendation is within tolerable bounds. The latter means that the user wants to calibrate her declared profile so as to receive recommendations *close to the ones she would receive if she had declared her true private profile*, regardless of the declaration policy of other users.

Let $\mathbf{q}_{-i} = (\mathbf{q}_1, \ldots, \mathbf{q}_{i-1}, \mathbf{q}_{i+1}, \ldots, \mathbf{q}_N)$ denotes the declared rating vector of all users except user i. Thus, it is $\mathbf{r}_i = f_i(\mathbf{q}_i, \mathbf{q}_{-i})$. Now, let $\tilde{\mathbf{r}}_i = f_i(\mathbf{p}_i, \mathbf{q}_{-i})$ be the resulting recommendation vector if user i had declared her true profile, while users other than i still

declared $\mathbf{q}_{-i}$. The need for adequate recommendation quality is described by the following constraint for user i:

$$(\tilde{\mathbf{r}}_i - \mathbf{r}_i)^2 \leq D \;\Leftrightarrow\; (f_i(\mathbf{p}_i, \mathbf{q}_{-i}) - f_i(\mathbf{q}_i, \mathbf{q}_{-i}))^2 \leq D\,, \tag{3}$$

where $(\tilde{\mathbf{r}}_i - \mathbf{r}_i)^2$ is the component-wise distance between rating vectors $\mathbf{r}_i$ and $\tilde{\mathbf{r}}_i$, and D is an upper bound that denotes the maximum tolerable distortion of the recommendation.

2.4 Optimization problem from a single user perspective

The user would like to submit rating profiles that do not reveal her real interests so as to protect her privacy as much as possible. An initial random profile provides a neutral starting point for the disclosure of users' real profile. Then, the user wishes to select a disclosure strategy so that the deviation of the declared profile from a random one is minimized. On the other hand, the user would like to make the declaration such that the quality of recommendation would not be affected too much, namely the recommendation would be close enough in distance (at most D) to the one she would get if she declared her real, private profile. The challenge arises because constraint (3) contains strategies $\mathbf{q}_{-i}$ of other users. The objective above can be formulated from the point of view of each user i as the optimization problem:

$$\max_{\mathbf{0}\leq\boldsymbol{\delta}_i\leq\mathbf{1}} u_i(g(\mathbf{t}_i, \mathbf{q}_i)), \tag{4}$$

$$\text{subject to: } \big(f_i(\mathbf{p}_i, \mathbf{q}_{-i}) - f_i(\mathbf{q}_i, \mathbf{q}_{-i})\big)^2 \leq D\,. \tag{5}$$

In other words, user i has to select her declared profile vector out of a set of feasible ones that satisfy (5). However, this set of feasible vectors is determined by declared profiles $\mathbf{q}_{-i}$ of other users. This feasible set is denoted by $F(\mathbf{q}_{-i})$. Each user tries to derive her strategy and solve the problem selfishly, i.e., she computes a declared profile $\mathbf{q}_i$ to maximize her own privacy, given the strategies (profile declarations) $\mathbf{q}_{-i}$ of others. This is precisely the game-theoretic setting, whereby the strategy of one user interacts with strategies of others.

The private profile $\mathbf{p}_i$ of each user i is given, fixed and private to user i, and it does not change. Rather, it is the declared profile $\mathbf{q}_i$ of each user i that will change depending on what user i declares. The interactions among users give rise to optimization problems (4)–(5). Each software agent tries to solve for its own user's declared profile, given that other users make their declarations $\mathbf{q}_j$, $j \neq i$.

2.4.1 Definition of NEP

A strategy profile $\mathbf{Q}^* = (\mathbf{q}_1^*, \ldots, \mathbf{q}_N^*)$ is called Nash equilibrium point (NEP) if for each user $i = 1, \ldots, N$, the following property holds:

$$u_i\big(g(\mathbf{t}_i, \mathbf{q}_i^*)\big) \geq \max_{\mathbf{q}_i \in F(\mathbf{q}_{-i}^*)} u_i\big(g(\mathbf{t}_i, \mathbf{q}_i)\big)\ \forall \mathbf{q}_i \neq \mathbf{q}_i^*\,. \tag{6}$$

In the NEP $(\mathbf{q}_1^*, \ldots, \mathbf{q}_N^*)$, no user i can further increase its privacy metric $g(\cdot)$ by unilaterally altering its declared profile to some $\mathbf{q}_i' \neq \mathbf{q}_i^*$, provided that other users stay with their NEP declared profiles.

3 A case study for the game-theoretic model

We now construct an example case study to present our model and study the problem. We consider a specific instance of a recommender system and specify example metrics for privacy and recommendation quality.

3.1 Use case specifics

3.1.1 Recommendation rating

We first discuss the model for aggregation functions $f_i(\cdot)$ that generate the recommendation rating. We consider a memory-based collaborative filtering (CF) recommendation approach (user–user *and* item–item). Each user declares her profile $\mathbf{q}_i$ for items $k \in \mathcal{S}_i$ to the recommendation server. The server then computes the following metrics $r_{i\ell}$ for non-viewed items by user i (but viewed by others), $\ell \notin \mathcal{S}_i$, $\ell \in \mathcal{S}_j$ for $j \neq i$ so as to include them in the recommendation vector that is sent to user i,

$$r_{i\ell} = \frac{1}{N-1} \cdot \sum_{\substack{j \neq i: \\ \ell \in \mathcal{S}_j, \ell \notin \mathcal{S}_i}} q_{j\ell} \cdot \frac{1}{|\mathcal{S}_i|} \sum_{k \in \mathcal{S}_i} \rho_{k\ell} q_{ik}, \tag{7}$$

where $\rho_{k\ell} \in [0, 1]$ is a metric that captures correlation (similarity) between items k and ℓ. Metric $r_{i\ell}$ takes into account ratings of users $j \neq i$ for viewed items in $\mathcal{S}_j$, correlation of these items with items in $\mathcal{S}_i$ and ratings of user i for items in $\mathcal{S}_i$. It also averages over users and over the number of items $|\mathcal{S}_i|$ viewed by user i. The server computes the metric above for all $\ell \notin \mathcal{S}_i$ and forms vector $\mathbf{r}_i$. The $|\mathcal{I}| \times |\mathcal{I}|$ correlation matrix of the pair-wise correlations between any two items is computed a priori, it is fixed and preloaded to the server, and it is known to user software agents. For example, if items are movies, the correlation between two movies could exist due to the existence of common features such as the movie theme, starring actors or director.

The recommendation metric above pertaining to user i implies a type of *CF* recommendation system. Indeed, the first term in the product in (7) implies a user–user *collaborative filtering* approach, where for each item ℓ under tentative recommendation to user i, the average of ratings of all other users that have viewed the item is computed. On the other hand, the second term can be viewed as a *item–item* CF recommendation approach, since it involves correlation between the item ℓ (candidate for recommendation) and other items that user i has viewed.

Remark 1 Our model can be modified and used for other models of recommender systems as well. For instance, in matrix factorization (MF)-based systems, there exists a rating matrix $\mathbf{R}$ with dimension $N \times |\mathcal{I}|$, where N is the number of users and $|\mathcal{I}|$ is the number of items. This matrix has some known entries that correspond to user ratings for certain items, while other entries are blank and are computed as the outcome of the recommendation algorithm. In matrix $\mathbf{R}$, the rating vector $\mathbf{q}_i$ of user i includes the non-blank elements of row i, while the rating vector $\mathbf{r}_i$ includes the blank elements of row i. The idea in MF is that $\mathbf{R}$ should be decomposed as $\mathbf{R} = \mathbf{U} \times \mathbf{V}^T$, where $\mathbf{U}$ is a $N \times L$ matrix and $\mathbf{V}$ is a $|\mathcal{I}| \times L$ matrix, where L is a number of latent features of items that are identified. The i-th row of U, u_i, compromises the profile of user i, while the j-th row of $\mathbf{V}$, v_j, compromises the profile of item j. If this factorization is achieved, the blank entries can be computed from the product $\mathbf{U} \times \mathbf{V}^T$.

Our model can be modified as follows to capture the MF approach. Declared ratings of a user i ($\mathbf{q}_i$) take place as the non-blank elements of row i in matrix $\mathbf{R}$. Given the declared ratings of all users ($\mathbf{Q}$), the recommendation server computes the profiles U and V by solving the following optimization problem:

$$\min_{\mathbf{U},\mathbf{V}} \sum_{i=1}^{N} \sum_{j=1}^{|\mathcal{I}|} (q_{ij} - u_i \cdot v_j)^2 + \lambda \sum_{i=1}^{N} ||u_i||_2^2 + \mu \sum_{j=1}^{|\mathcal{I}|} ||v_j||_2^2$$

If the rating q_{ij} change as a result of the strategy of each user, matrices $\mathbf{U}$ and $\mathbf{V}$ will change, and therefore, the recommendation ratings $\mathbf{r}_i$ will change.

We can in principle write an expression similar to that in (3) and formulate an optimization problem similar to (4)–(5), where $f_i(\mathbf{Q}) = u_i \cdot V^T$ to denote the mapping from $\mathbf{q}_i$ (non-blank elements of row i of $\mathbf{R}$) to $\mathbf{r}_i$ (blank elements of row i). Each user solves problem (4)–(5) to find the profile declaration vector that satisfies (5).

Remark 2 Besides averaging, other ways of aggregating user ratings may be used by the recommendation server. For example, weighted averaging with different weights may be applied. Then, considering that w_j is the weight of user j in the system, Eq. 7 can be written as

$$r_{i\ell} = \frac{1}{\sum_{\substack{j\neq i:\\ \ell\in\mathcal{S}_j, \ell\notin\mathcal{S}_i}} w_j} \cdot \sum_{\substack{j\neq i:\\ \ell\in\mathcal{S}_j, \ell\notin\mathcal{S}_i}} w_j q_{j\ell} \cdot \frac{1}{|\mathcal{S}_i|} \sum_{k\in\mathcal{S}_i} \rho_{k\ell} q_{ik}, \tag{8}$$

or, only ratings from a subset of users could be taken into account, e.g., only K users that have viewed common items with user i, where K is a parameter of the recommendation server. These K users are denoted by $\mathcal{U}_i$. In that case, the left-most part in (7) would be

$$\frac{1}{K} \sum_{\substack{j\in\mathcal{U}_i: |\mathcal{U}_i|=K\\ \mathcal{S}_i\cap\mathcal{S}_j\neq\emptyset}} q_{j\ell}\,.$$

3.1.2 Privacy metric

We consider negative Euclidean distance, and we define the function $g(\cdot)$ that quantifies privacy for user i as

$$g(\mathbf{t}_i, \mathbf{q}_i) = -\sum_{k\in\mathcal{S}_i} (t_{ik} - q_{ik})^2\,. \tag{9}$$

If we take the utility function to be linear, i.e., $u_i(x) = x$, then $u_i(g(\mathbf{t}_i, \mathbf{q}_i)) = g(\mathbf{t}_i, \mathbf{q}_i)$.

The metric above reflects the intuitive fact that privacy increases as the Euclidean distance between the declared and initial "random" profiles decreases.

Remark 3 Other privacy metrics besides (9) can also be used. For instance, for user i and item k, a factor of the form $(t_{ik} - \bar{r}_k)^2$ could be considered, where $\bar{r}_k$ is the average user rating for item k. Note that in this metric, factor $\bar{r}_k$ comprises strategies of multiple users. A game-theoretic model would still be applicable. In this case, user interaction arises both in the user objective through $\bar{r}_k$ and in the recommendation quality constraint as above.

3.1.3 Recommendation quality

Since users modify their private rating vector when they declare it to the server in an effort to maximize their privacy, they affect the quality of recommendation they receive. This quality is quantified through a measure of the difference between the recommendation that user i gets if she declares profile $\mathbf{q}_i$ and the one she would get if she declared the real profile $\mathbf{p}_i$, regardless of what other users do. Each user $j \neq i$ makes declarations $\mathbf{q}_j$. The constraint to be fulfilled for acceptable recommendation quality for user i is derived by using (5) and (7), and it is

$$\left[\sum_{k\in\mathcal{S}_i} \rho_{k\ell}(p_{ik} - q_{ik})\right]^2 \left[\frac{1}{(N-1)|\mathcal{S}_i|}\sum_{\substack{j\neq i:\\ \ell\in\mathcal{S}_j}} q_{j\ell}\right]^2 \leq D \;\; \forall l \notin S_i. \tag{10}$$

3.2 Information exchange between users and the recommendation server

An iterative process of data exchange between users and the recommendation server takes place. We envision a software agent at the side of each user i which acts on behalf of the user. The agent of user i is responsible for preserving privacy of user i and for delivering good-quality recommendations to i. The agent continuously sends queries for recommendation to the server. The steps of data exchange are summarized as follows:

– **STEP 0:** An initial (default) disclosure strategy $\boldsymbol{\delta}_i^{(0)}$ is used by each user.

For each user, $i = 1, \ldots, N$:

– **STEP 1:** At each iteration cycle $t > 0$, the server passes to each agent i the *aggregate* ratings of other users for items that user i has not viewed yet. These aggregate ratings are computed based on ratings that agents of other users have sent to the server at the same iteration cycle. That is, for each item $\ell \notin \mathcal{S}_i$ and users $j \neq i$, the server computes and passes to user i the quantity $\frac{1}{N-1}\sum_{j\neq i:\ell\in\mathcal{S}_j} q_{j\ell}^{(t)}$.

– **STEP 2:** The agent solves optimization problem $(\mathcal{P})$:

$$\max_{\mathbf{0}\leq\boldsymbol{\delta}_i^{(t)}\leq\mathbf{1}} g(\mathbf{t}_i, \mathbf{q}_i^{(t)}) = -\sum_{k\in\mathcal{S}_i} (t_{ik} - q_{ik}^{(t)})^2, \tag{11}$$

Based on Eq. 1 the problem $(\mathcal{P})$ can be written as follows:

$$\min_{\mathbf{0}\leq\boldsymbol{\delta}_i^{(t)}\leq\mathbf{1}} \sum_{k\in\mathcal{S}_i} \delta_{ik}^2 (t_{ik} - p_{ik}^{(t)})^2, \tag{12}$$

subject to constraint (10), which includes the aggregate $\frac{1}{N-1}\sum_{j\neq i:\ell\in\mathcal{S}_j} q_{j\ell}^{(t-1)}$ from the previous iteration. Thus, it computes her own disclosure strategy and her declared rating vector $\mathbf{q}_i^{(t)}$ for the current iteration t.

– **STEP 3:** Each agent i declares vector $\mathbf{q}_i^{(t)}$ to the server.

– **STEP 4:** The server uses these ratings to compose aggregated quantities for items that have not been viewed by other users. Go to Step 1. Repeat until convergence.

The system and interactions are depicted in Fig. 1. Agents (users) update their rating vectors in subsequent iterations. At each iteration cycle t, each agent i solves its own optimization problem based on *aggregate* ratings of other agents that have been passed to i by

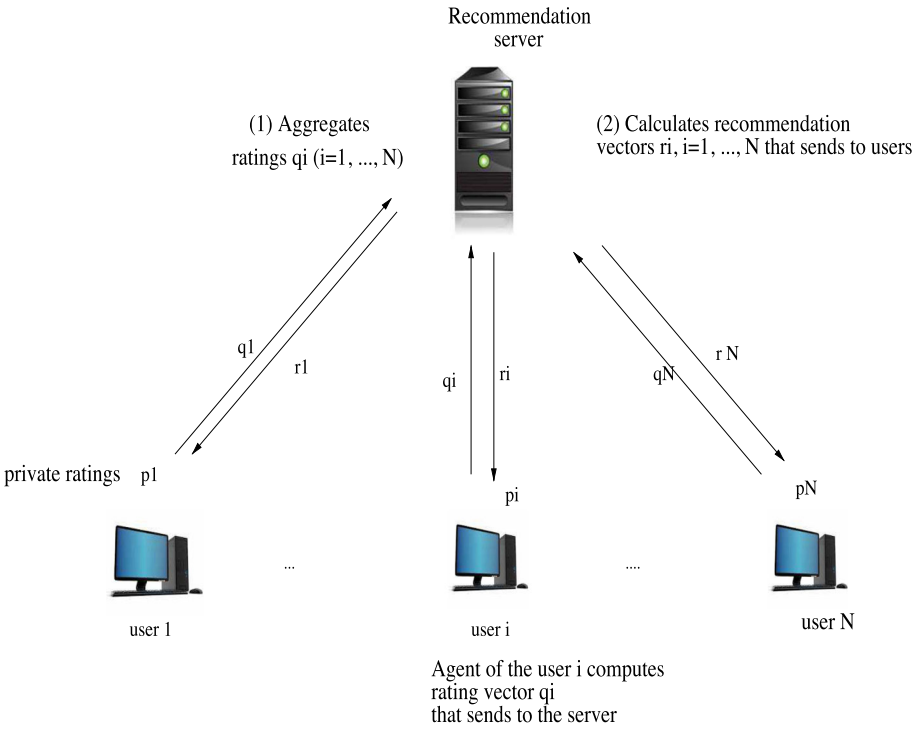


Fig. 1 Overview of the system architecture and data exchange process among users and the recommendation server at each iteration

the server at the end of the previous iteration cycle. User i declares her ratings to the recommendation server. The server collects user rating vectors, and it announces the aggregate rating to different users in order for the new iteration to start. As can be observed, the process involves circulation of aggregate quantities in the system and no revelation of real ratings.

Convergence to the NEP: The procedure described above involves a *Linear Programming* (LP) problem that is solved by each user at each iteration cycle. The iterative procedure above is an instance of iterative *best-response* for each user. It is known from game theory that for LP problems, the sequence of best-response updates that starts from any initial vector $\mathbf{q}_i^{(0)}$, $i = 1, \ldots, N$ converges to the NEP.

4 Selfish and cooperative profile declaration

4.1 Analysis of the game-theoretic interaction

We now present some analysis of our game-theoretic model for the case study of Sect. 3. We set $w_{ik} = (t_{ik} - p_{ik})$, and $\mathbf{w}_i = (w_{ik} : k \in \mathcal{S}_i)$. Problem $(\mathcal{P})$ (Eq. subject to (10)) that is solved by each user i at each iteration is written as

$$\min_{\mathbf{0}\le\boldsymbol{\delta_i}\le\mathbf{1}} \sum_{k\in\mathcal{S}_i} \delta_{ik}^2 w_{ik}^2,$$

$$\text{subject to:} \quad \sum_{k\in\mathcal{S}_i} \beta_{ik}(1-\delta_{ik}) w_{ik} \le \pm(N-1)\sqrt{D}, \tag{13}$$

$$0 \le \delta_{ik} \le 1, \forall k \in S_i$$

$$\text{with } \beta_{ik} = \frac{1}{|\mathcal{S}_i|} \sum_{\ell\notin\mathcal{S}_i} \rho_{k\ell} \sum_{j\neq i:\ell\in\mathcal{S}_j} q_{j\ell}, \tag{14}$$

and it is a quadratic programming problem.

We observe that the objective function is convex, and that the inequality constraint functions are affine. Then, a unique solution exists for the above optimization problem. Since the objective and constraint functions are differentiable and the linearity constraint qualification holds, KKT conditions are necessary and sufficient conditions for optimality. The application of these optimality conditions leads to the following Lagrangian cost,

$$J = -\sum_{k\in S_i} \delta_{ik}^2 w_{ik}^2 + \lambda_i \left[\sum_{k\in S_i} \beta_{ik}\delta_{ik} w_{ik} - \sum_{k\in S_i} \beta_{ik} w_{ik} \pm (N-1)\sqrt{D} \right] + \sum_{k\in S_i} \mu_{ik}(1-\delta_{ik}) + \sum_{k\in S_i} \nu_{ik}\delta_{ik} \tag{15}$$

Then, we have the following conditions

$$\frac{\partial J}{\partial \delta_{ik}} = 0 \Longrightarrow -2\delta_{ik} w_{ik}^2 + \lambda_i \beta_{ik} w_{ik} - \mu_{ik} + \nu_{ik} = 0 \tag{16}$$

$$\lambda_i \left[\sum_{k\in S_i} \beta_{ik}\delta_{ik} w_{ik} - \sum_{k\in S_i} \beta_{ik} w_{ik} \pm (N-1)\sqrt{D} \right] = 0 \tag{17}$$

$$\mu_{ik}(1-\delta_{ik}) = 0$$

$$\nu_{ik}\delta_{ik} = 0 \tag{18}$$

$$\lambda_i \ge 0,\ \mu_{ik} \ge 0,\ \nu_{ik} \ge 0,$$

$$0 \le \delta_{ik} \le 1. \tag{19}$$

In the sequel, we shall proceed to solve these equations. We consider the following possibilities for each i:

1. $\delta_{ik} = 0$, which corresponds to the case that the users declare the "initial" random profile $q_{ik} = t_{ik}$. Based on Eq. 18, we get that $\mu_{ik} = 0$. Also Eq. 16 gives: $\lambda_i = -\frac{\nu_{ik}}{\beta_{ik} w_{ik}}$. Then, we have the following cases:
   - if $\lambda_i = 0$, we get: $\nu_{ik} = 0$ and the following inequality should be satisfied $\sum_{k\in S_i} \beta_{ik} w_{ik} \le \pm(N-1)\sqrt{D}$
   - if $\lambda_i > 0$, we get: $\nu_{ik} > 0$ and $\sum_{k\in S_i} \beta_{ik} w_{ik} = \pm(N-1)\sqrt{D}$
2. $\delta_{ik} = 1$ which corresponds to the case that the users declare their private profile, $q_{ik} = p_{ik}$. Based on Eq. 18, we get that $\nu_{ik} = 0$. Also Eq. 16 gives: $\lambda_i = \frac{2w_{ik}^2 + \mu_{ik}}{\beta_{ik} w_{ik}}$. Then, we have the following cases:
   - if $\lambda_i = 0$, we get: $\mu_{ik} = -2w_{ik}^2 \le 0$ but μ_{ik} should be positive .
   - if $\lambda_i > 0$, we get: $\mu_{ik} > 0$, while Eq. 17 results in $D = 0$.

3. $0 < \delta_{ik} < 1$. In this case, we have that $\lambda_i > 0$, $\nu_{ik} = 0$, $\mu_{ik} = 0$. Based on Eqs. 16 and 18, we get : $\lambda_i = 2\frac{\sum_{k\in S_i}\beta_{ik}w_{ik}\pm(N-1)\sqrt{D}}{\sum_{k\in S_i}\beta_{ik}^2}$ and $\delta_{ik} = \frac{\beta_{ik}}{w_{ik}}\frac{\sum_{k\in S_i}\beta_{ik}w_{ik}\pm(N-1)\sqrt{D}}{\sum_{k\in S_i}\beta_{ik}^2}$

Considering the above cases for $\boldsymbol{\delta}_i$, we observe that the objective function is minimized for $\delta_{ik} = 0$. However, this case requires the satisfaction of the inequality $\sum_{k\in S_i}\beta_{ik}w_{ik} \leq \pm(N-1)\sqrt{D}$. Otherwise, each user i selects a disclosure strategy such as $0 < \delta_{ik} < 1$ and $\delta_{ik} = \frac{\beta_{ik}}{w_{ik}}\frac{\sum_{k\in S_i}\beta_{ik}w_{ik}\pm(N-1)\sqrt{D}}{\sum_{k\in S_i}\beta_{ik}^2}$.

4.2 Cooperative profile declaration

Now, we consider scenarios in which agents may cooperate in declaring their rating profiles in an effort to mutually benefit in preserving their respective users' privacy. We consider two possible cooperation scenarios.

4.2.1 First scenario: private rating exchange

The first scenario amounts to private rating profile exchange among cooperating users. Users are assumed to trust each other and agree to cooperate offline. By cooperation, we mean that these users exchange ratings for items that one has rated and the other has not. For items that both users have rated, each user keeps her own rating. As a result, both users appear to the server as having rated the same set of items, and each user has an enhanced set of viewed items. Next, each user solves for herself the optimization problem (13) and the iterative process proceeds as in Sect. 3.2.

4.2.2 Second scenario: joint profile declaration decision

In a second cooperation scenario, all users or a subset of them coordinate by jointly deciding on their declared ratings to the server. A global objective $G(\mathbf{T}, \mathbf{Q})$ needs to be defined for the set of collaborating users $\mathcal{U}$. For instance, $G(\mathbf{T}, \mathbf{Q}) = \sum_{i\in\mathcal{U}} u_i\big(g(\mathbf{t}_i, \mathbf{q}_i)\big)$ denotes the total amount of privacy of users. Or it could be $G(\mathbf{P}, \mathbf{Q}) = \min_{i\in\mathcal{U}} u_i\big((\mathbf{t}_i, \mathbf{q}_i)\big)$, denoting the least privacy of a user. Users decide jointly so as to optimize the global objective.

A *feasible cooperation regime* for the N users in $\mathcal{U}$ is a *joint* profile declaration strategy $\mathbf{Q}^0 = (\mathbf{q}_1^0, \ldots, \mathbf{q}_N^0)$ such that

$$\mathbf{q}_i^0 \in F(\mathbf{q}_{-i}^0), \text{ , and } u_i\big(g(\mathbf{t}_i, \mathbf{q}_i^0)\big) \geq u_i\big(g(\mathbf{t}_i, \mathbf{q}_i^*)\big), \text{ , } \forall\, i \in \mathcal{U}, \tag{20}$$

where $\mathbf{Q}^* = (\mathbf{q}_i^* : i = 1, \ldots, N)$ is the NEP. That is, a cooperation regime is feasible if: (i) it belongs to the set of feasible vectors, as specified by constraint (3) for all users, (ii) each user has a privacy stemming from cooperation that is at least as much as the one she has at the NEP. This latter requirement renders cooperation meaningful for each user and provides the incentive to the user to participate in the coordinated effort.

A first goal in this cooperation scenario is to jointly find the set of feasible cooperation regimes, call it $\mathcal{F}_c$. If $\mathcal{F}_c \neq \emptyset$, then there exists at least one joint strategy $\mathbf{Q}^0$ such that all users are privacy-wise better off compared to the NEP, and this strategy can be found by solving the set of inequalities in (20). If $\mathcal{F}_c \neq \emptyset$, a further goal is to find the cooperation regime among the feasible ones that maximizes the global privacy objective $G(\mathbf{T}, \mathbf{Q})$ or a regime that guarantees certain properties of the privacy vector $(u_i\big(g(\mathbf{t}_1, \mathbf{q}_1)\big), \ldots, u_N\big(g(\mathbf{t}_N, \mathbf{q}_N)\big))$.

Let us consider a simple example with $N = 2$ users. Assume that user 1 has viewed items in set $\mathcal{I}_1 = \{A, B\}$ and has private rating vector (p_{1A}, p_{1B}), while user 2 has viewed items in $\mathcal{I}_2 = \{B, C\}$ with private rating vector (p_{2B}, p_{2C}). Let $w_{ik} = (t_{ik} - p_{ik})$ for $i = 1, 2$ and $k \in \{A, B, C\}$. If users 1 and 2 cooperate, they try to maximize their *sum* privacy subject to the constraints that (i) the recommendation error *for each user* is less than a threshold, (ii) each user privacy is at least as much as the one at the NEP. The global optimization problem is

$$\min_{\delta_{1A},\delta_{1B},\delta_{2B},\delta_{2C}} \delta_{1A}^2 w_{1A}^2 + \delta_{1B}^2 w_{1B}^2 + \delta_{2B}^2 w_{2B}^2 + \delta_{2C}^2 w_{2C}^2 \tag{21}$$

$$\text{subject to: } \rho_{AC} q_{2C}(\delta_{1A} - 1) w_{1C} + \rho_{BC} q_{2C}(\delta_{1B} - 1) w_{1B} \leq \sqrt{D}, \tag{22}$$

$$\rho_{AB} q_{1A}(\delta_{2B} - 1) w_{2B} + \rho_{AC} q_{1A}(\delta_{2C} - 1) w_{2C} \leq \sqrt{D},$$

$$\delta_{1A}^2 w_{1A}^2 + \delta_{1B}^2 w_{1B}^2 \geq \delta_{1A}^{*2} w_{1A}^2 + \delta_{1B}^{*2} w_{1B}^2,$$

$$\delta_{2B}^2 w_{2B}^2 + \delta_{2C}^2 w_{2C}^2 \geq \delta_{2B}^{*2} w_{2B}^2 + \delta_{2C}^{*2} w_{2C}^2$$

$$0 \leq \delta_{1A} \leq 1, 0 \leq \delta_{1\mathbf{B}} \leq 1, 0 \leq \delta_{2A} \leq 1, 0 \leq \delta_{2\mathbf{B}} \leq 1 \tag{23}$$

where $(\delta_{1A}^*, \delta_{1B}^*, \delta_{2B}^*, \delta_{2C}^*)$ is the NEP. The problem is a nonlinear one, and it can be solved through the conditions of the Karush–Kuhn–Tucker (KKT) theorem.

5 Experimental study

In this section, we apply our methods to real datasets and evaluate the performance of our game-theoretic and cooperative approaches in terms of privacy and recommendation quality.

In our experimental study, we used a recommender system that is based on the CF model discussed in Sect. 3.1.1.

5.1 Datasets

The experiments have been conducted on publicly available datasets of movie ratings that are commonly used in recommender system research. The original datasets have been properly filtered to address common problems in recommender systems (e.g., sparsity of information) and select appropriate experimental datasets.

We have used two Movielens datasets (Movielens100K, Movielens20M) provided by GroupLens [1]. Both datasets contain users that have rated at least 20 movies. Text files describing movie genres according to the Internet Movie Database (IMDB) were used in order to compute Jaccard similarity coefficient. The Movielens100K dataset contains 100,000 ratings (in the range 1–5) from 943 users on 1682 movies. The Movielens20M consists of 20 million ratings given by 138,000 users to 27,000 movies. The original dataset has been filtered selecting the movies with at least 20 ratings.

The third dataset is an extension of MovieLens10M and was also published by the Grouplens Research Group. It was released in the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec), 2011. It consists of 855,598 ratings (in the range 0.5–5 with step 0.5) from 2,113 users on 10,197 movies. We have randomly selected a portion of the dataset consisting of 2,000 movies and 992 users.

Table 1 summarizes the characteristics of the datasets that we used in experiments.

[1] https://grouplens.org/

Table 1 Dataset characteristics

Datasets	Description (ratings)	Users	Movies	Density	Average rating
MovieLens100K	100,000	943	1682	6.3%	3.53
MovieLens20M	19,799,049	138,493	10,239	2%	3.5
HetRec2011	152,629	992	2000	8.3%	3.49

5.2 Accuracy evaluation metrics

Two widely used evaluation metrics that measure the error between the rating that the user gave (which is stored in the test set) and the predicted value (which is derived by our approach) are the *mean absolute error* (MAE) and the *root mean square error* (RMSE). The *mean absolute error* is defined as the average absolute difference between the real rating and the predicted one, averaged over the set of ratings, that is,

$$\mathrm{MAE} = \frac{\sum_{r_i \in \mathcal{R}_{\mathrm{test}}} |\mathrm{pr}_i - r_i|}{|R_{\mathrm{test}}|}, \tag{24}$$

where pr_i is the predicted rating, r_i is the real rating, and i is an index running through the ratings set $\mathcal{R}_{\mathrm{test}}$. The *Root Mean Square Error* is defined as

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{r_i \in \mathcal{R}_{\mathrm{test}}} (\mathrm{pr}_i - r_i)^2}{|R_{\mathrm{test}}|}}. \tag{25}$$

Precision shows the percentage of recommended items that are of interest to the user, thus she has rated them with 4 or 5 in the test set, while *Recall* shows the percentage of items that have been recommended out of the ones that the user likes.

5.3 Results

5.3.1 Recommendation quality and privacy

We divided the datasets into training and test sets, following the rule 80–20%. Our goal is to examine the accuracy with which items in the test set were predicted as good recommendations. The recommendations of our approach in the experimental part are based on (7). We assume that if our method performs well for items in the test set, it will most probably perform well for new and unknown items to the user. In order to compute $\rho_{k\ell} \in [0, 1]$ the correlation between items k and ℓ, we use three known similarity measures: *Pearson's correlation, Euclidean distance* and *Jaccard similarity coefficient*.

Tables 2 and 3 depict experimental results regarding the accuracy of recommendations. We evaluated the performance of our approach using the evaluation and correlation metrics above, while we set the value of the maximum distortion tolerance in recommendation to $D = 2$.

Table 2 shows that the mean error is in general less than 1. Thus, our approach provides recommendations that do not diverge significantly from the users' interests. Moreover, Table 3 shows that for the MovieLens100K dataset, 57–68% of positive predictions were correct. Also

Table 2 Accuracy metrics results

	Correlation	Accuracy	
		MAE	RMSE
MovieLens100K	Pearson's correlation	0.7	1.39
	Euclidean distance	0.84	1.52
MovieLens20M	Pearson's correlation	0.68	1.2
	Euclidean distance	0.8	1.49
HetRec2011	Pearson's correlation	0.59	1.26
	Euclidean distance	1.04	1.72
	Jaccard coefficient	0.82	1.57

Table 3 Precision, recall and F-measure results

	Correlation	Precision	Recall	F-measure
MovieLens100K	Pearson's correlation	0.68	0.25	0.37
	Euclidean distance	0.57	0.36	0.44
MovieLens20M	Pearson's correlation	0.73	0.31	0.42
	Euclidean distance	0.62	0.43	0.52
HetRec2011	Pearson's correlation	0.63	0.19	0.29
	Euclidean distance	0.47	0.07	0.10
	Jaccard coefficient	0.62	0.23	0.33

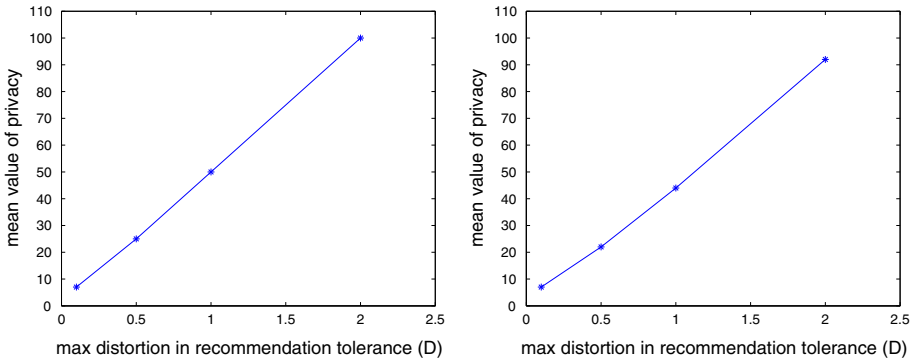
our approach achieves to recommend 25–36% of items that are related to users' interests. The experiments with the HetRec2011 dataset show that the users were satisfied by 47–63% of the recommended items. Similarly, in the case of MovieLens20M, 60–70% of recommendation corresponds to user interests. The larger dataset gives slightly better error for sure (MAE, RMSE) as expected (the more the users/movies are, the better the estimation is).

The differences we observe on the accuracy among various datasets lead us to conjecture that the error depends on properties of the dataset but also on the metric we use for correlation (Pearson, Euclidean, Jaccard). However, in all cases, the experimental results show that our method attains item recommendations that are related to users' interests although significant changes were made to ratings for privacy reasons.

Furthermore, in order to evaluate the user privacy in our experimental study, we adopted an Euclidean-based distance metric, and we assess the divergence of declared profile from the real one. That is,

$$d(\mathbf{p}_i, \mathbf{q}_i) = \sum_{k \in \mathcal{S}_i} (p_{ik} - q_{ik})^2. \tag{26}$$

Figures 2 and 3 depict the relation between maximum distortion tolerance in recommendation, D and average privacy ($\frac{1}{N} \sum_{i=1}^{N} d(\mathbf{p}_i, \mathbf{q}_i)$, where $d(\cdot)$ is the distance metric [Eq. 26)] of all users in the case of Movielens datasets (Movielens100K, Movielens20M) and HetRec2011, respectively. For certain values of D, i.e., $D = 0.1, 0.5, 1, 2$, we can see that privacy and D behave in the similar way. This happens because for a more stringent constraint on the expected recommendation error, user privacy is smaller, namely the user has to reveal more information about her personal interests and ratings.



(a) MovieLens100K dataset **(b)** Movielens20M dataset

Fig. 2 Relationship between recommendation quality tolerance parameter D and average user privacy (Movielens datasets)

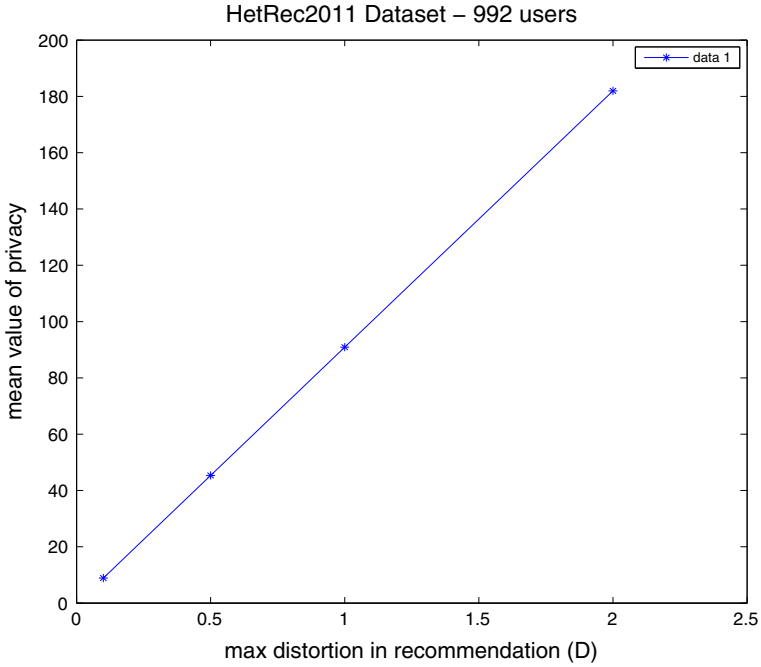


Fig. 3 Relationship between D and average user privacy (HetRec2011 dataset)

5.3.2 Game-theoretic user interaction

As discussed above, our approach involves an iterative data exchange between users and the recommendation server, which converges after a number of iterations. We considered a specific user, and we studied convergence of our approach for different correlation metrics, as well as for different values of D.

The convergence occurs irrespective of the dataset and of the scale of it. Thus, the evaluation metrics used in experimental evaluation of our model, namely the proposed privacy metric

and the accuracy of the recommendations are not affected by the volume of data collections. Furthermore, the game-theoretic interaction is executed with local relatively computations based on locally available information by each agent and with limited circulated information; all these make the approach scalable.

Figure 4a depicts the convergence of our approach considering the Movielens100K dataset and different metrics to measure item correlation, while D is set to 2. Convergence of the best-response mechanism is guaranteed after 4–7 iterations, depending on the correlation metric used. We note that the ultimate value of user privacy metric depends on that correlation metric. Figure 4b depicts the convergence of our approach for different values of D. As the tolerance for recommendation error increases, privacy increases as well. Consequently, if a user is more tolerant to large recommendation errors, she does not have to reveal much information. Experiments with the other datasets (Sect. 5.1) did not depict significant differences in convergence and in the privacy metric in plots of Fig. 4.

5.3.3 Cooperative and selfish profile declaration

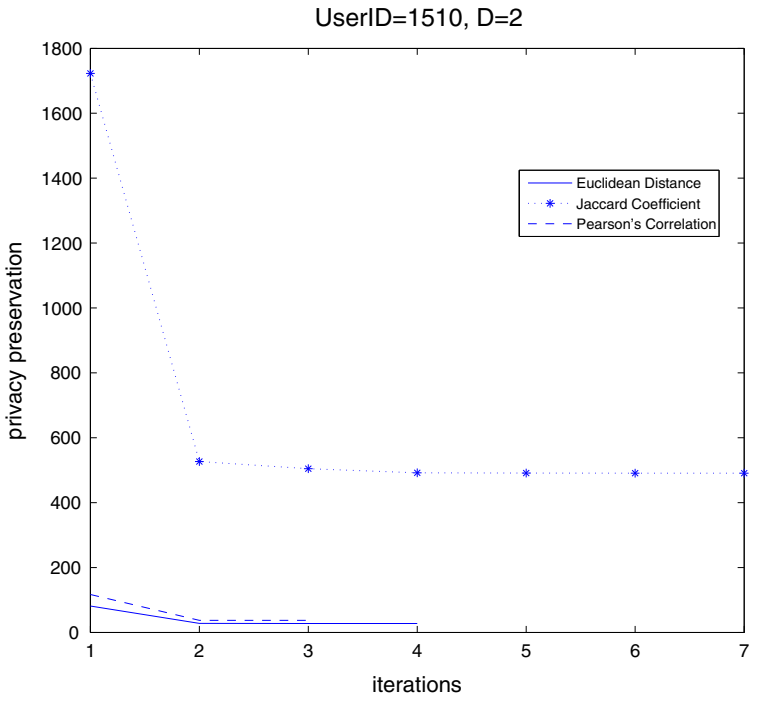
In this subsection, we discuss the results of our experimental study for the cooperation strategies presented in Sect. 4.

First cooperation scenario: We evaluate the performance of our approach for the first cooperation scenario according to which users that decide to cooperate exchange ratings on items that one has rated and the other has not. Thus, cooperating users appear to have rated the same number of items. Experiments on the considered datasets show that privacy for users who cooperate and exchange data is larger than the one when they act independently. Figures 5 and 6 show results for a group of 10 users from MovieLens100K when users 1 and 2 cooperate. It can be observed that the privacy metrics of users 1 and 2 have increased significantly after their cooperation.

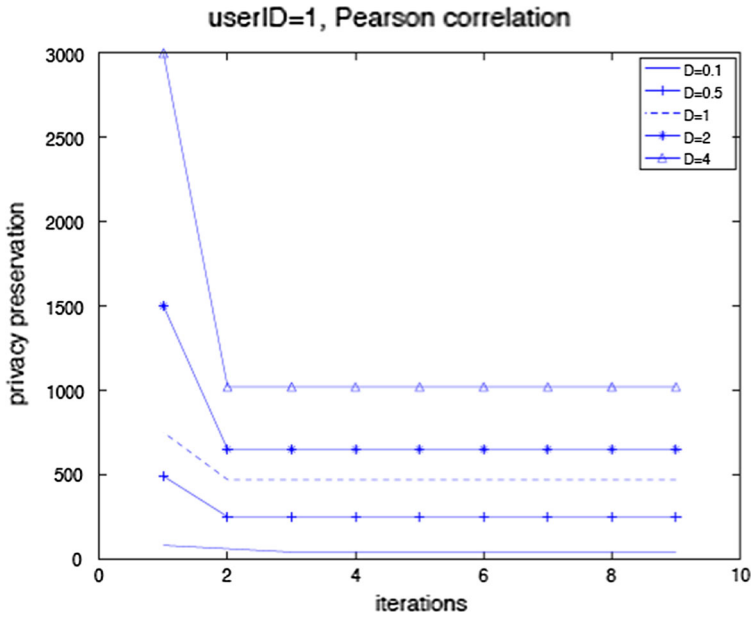
Second cooperation scenario: We also carried out experiments for the second cooperation scenario, according to which users jointly decide on the ratings to declare for their items. We selected 100 sets of 10 users from the MovieLens100K dataset, and for each subset of 10 users, we randomly chose one pair of users to cooperate. For each user, we randomly selected 5 of his available ratings in the MovieLens dataset. We evaluated the performance of the cooperative approach in terms of privacy after averaging over 100 times, one for each set of users.

Table 4 presents the average privacy of the two users when they cooperate compared to when they act selfishly in a game-theoretic fashion. We observe that the privacy of each user significantly increases when users cooperate. For small values of D, user privacy under cooperation is between 3 and 10 times more than the one under competition. For larger values of D, the privacy under cooperation is still 25% more than the one under competition.

Figure 7 depicts the total average privacy for a pair of users with respect to recommendation error tolerance D for cooperating and for competing users. The averaging is again performed for 100 experiments. Total user privacy increases as D increases both for cooperation and for the game-theoretic competition. The benefit in terms of total user privacy stemming from cooperation compared to competition is seen to *decrease* as the value of D increases. This can be explained since a larger tolerance in recommendation error reduces the effect of user conflict in the game-theoretic interaction. One could also see this readily by comparing the dependence of total user privacy on D for both cases. For the case of competition, it was shown that total user privacy was proportional to D, while for the case of cooperation, privacy was shown to be at most proportional to $\sqrt{D}$. From the plot, the intuition becomes clear that



(a)



(b)

Fig. 4 Convergence of our approach for different **a** correlation metrics, **b** values of D

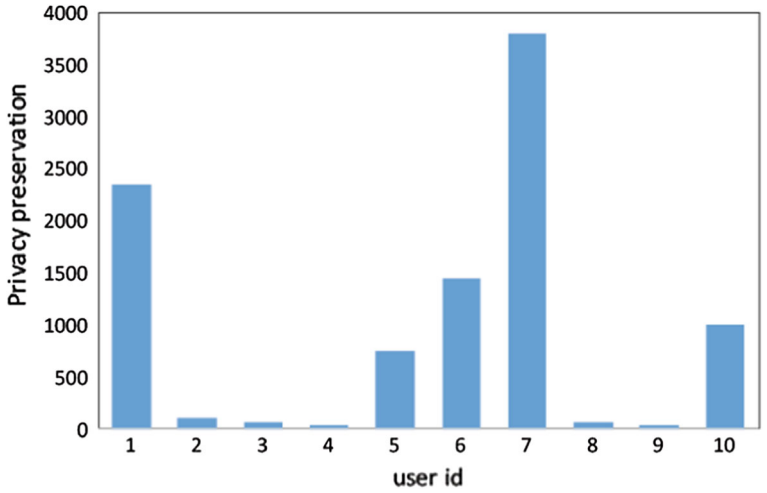


Fig. 5 First cooperation scenario: privacy of users 1, 2 before cooperation

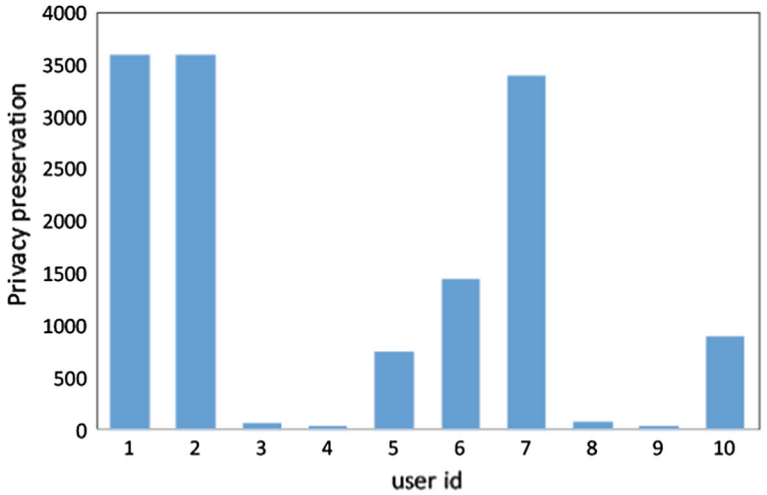


Fig. 6 First cooperation scenario: privacy of users 1, 2 after cooperation

it makes more sense to cooperate when the value of D is low, i.e., when there exist stringent constraints on the recommendation error.

6 Related work

Recommender systems automatize the generation of recommendations based on data analytics techniques [10]. The approaches proposed in the literature can be classified as follows: (i) collaborative filtering, (ii) content-based ones and (iii) hybrid approaches. In *collaborative filtering* (CF) systems, recommendations to a user are performed based on past ratings of other users. Specifically, neighborhood-based CF approaches assume that users with corre-

Table 4 Privacy of individual users in the case of cooperation and competition

	Distortion in recommendation tolerance (D)				
	0.1	0.5	1	1.5	2
Cooperation					
User 1	30.96	31.17	40.18	41.5	45.56
User 2	30.45	35.71	37.68	43.2	44.04
Competition					
User 1	3.74	10.26	16.85	30.12	37.44
User 2	3.73	10.22	16.37	31.17	37.56

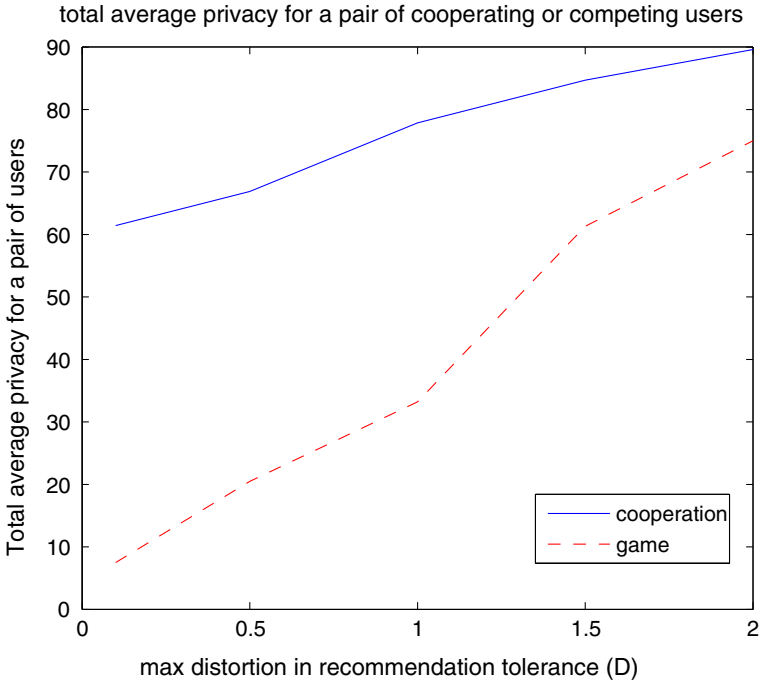


Fig. 7 Total privacy for a pair of users in the case of cooperation and competition

lated interests will most probably like similar items. The Pearson's correlation coefficient is the most widely used measure of similarity between the ratings of two users [11], but there exist other measures that are also used [12]. An extension to neighborhood-based CF is the item-to-item CF approach, which matches a user's rated items to similar items rather than similar users [13,14]. On the other hand, *content-based* methods focus on a specific user and provide recommendations for new items based on their similarity with items that the user has liked in the past. Balabanovic et al. [15] assume that user preferences can be treated as a query, and that unrated objects are scored based on their similarity to this query. An alternative approach is proposed in [16], which treats recommendation as a classification problem. Finally, *hybrid* approaches aim to leverage advantages of both content-based and collaborative filtering ones. Cotter et al. [17] propose a simple approach that collects results

of both content-based and CF approaches and merges these results to produce the final recommendation. Melville et al. [18] propose a framework that uses content-based predictions to convert a sparse user-rating matrix into a full rating one, and subsequently, it employs a CF method to provide recommendations.

Since recommendation servers need to have access to user preferences in order to predict other items of interest to users, privacy of users is put at risk. A number of different methods such as *cryptographic, obfuscation, perturbation and probabilistic methods* have been developed to address privacy in recommender systems. Erkin et al. [19] introduce a recommendation system that allows users to jointly compute their profile data and receive recommendations without sharing true data with other parties. In [20], they proposed another approach that combines matrix factorization and cryptographic methods to provide recommendations without learning the real user ratings. Another category of works refers to approaches that store user profiles locally and run the recommender system in a distributed fashion. Miller et al. [21] propose the PocketLens algorithm for CF in a distributed environment. Their approach requires only the circulation of similarity measures in the system, and thus, it protects user privacy by keeping their profiles secret.

Perturbation methods modify users' real rating profile by a selected distribution. Polat and Du [22] propose a randomized perturbation technique to protect user privacy in CF approaches. Similar, Bilge et al. [23] consider that a uniform distribution noise is added to the real users' ratings before they are used in the recommendation procedure. An approach that uses the deviation between two items as the adding noise is proposed in [24].

The work proposed in [25] addresses the problem of protecting user privacy through substituting the centralized CF system by a virtual peer-to-peer one. Also user profiles are partially modified by adding some degree of uncertainty. Although these methods almost eliminate user privacy losses, they require high degree of cooperation among users so as to achieve accurate recommendations. A similar approach that exploits the idea of group-based privacy-preserving recommender systems is presented in [26]. Users within a group exchange their preferences with other members in their group, and only the exchanged information is sent to the service provider.

A hybrid CF model is used for providing suggestions to group members. The service provider makes recommendations to groups, while recommendations to individuals (group members) are made locally. In [27], a privacy-preserving collaborative filtering method is proposed that exploits microaggregation to perturb data. Also, it uses the idea of organizing users to groups in order to achieve k-anonymity.

In [28], a new measure is introduced to estimate the similarity between two users without compromising user privacy. A randomly generated set of ratings is shared between two users, and then users estimate the number of concordant, discordant and tied pairs of ratings between their own profiles and the randomly generated ones. An alternative method for preserving privacy is discussed in [29]. Users create communities, and each user seeks recommendations from the most appropriate community. Each community computes a public aggregation of user profiles without violating individual profile privacy based on distributed singular value decomposition (SVD) of the user rating matrix. A distributed mechanism that focuses on obfuscating user–item connections is proposed in [30]. Each user arbitrarily selects to contact others over time and modifies her local profile off-line through an aggregation process. Users periodically synchronize their profiles at the server with their local ones. The authors in [31] present a recommender system that uses an adaptive obfuscation mechanism. The main idea is that users do not disclose a set of item ratings, and they ask the system to send predictions for these items. Based on received recommendations, each user adjusts the obfuscation level for new items so as to receive recommendations with the desired level of accuracy. Finally,

a privacy preservation approach for CF that uses k-means bisecting algorithm is proposed in [32]. It handles the problem of preserving individual privacy by randomly perturbing users' rating and by randomly filling some fraction of unknown ratings.

7 Conclusion

We introduced a game-theoretic framework for capturing the interaction and conflicting interests of users for privacy preservation in recommender systems. The game-theoretic model arises because the declared rating of one user affects the recommendation quality of all users. We demonstrated both in theory and by experimenting with real datasets the existence of stable declared rating vectors (NEP). We also showed that users may increase their privacy substantially if they cooperate in declaring their ratings rather than acting selfishly. In our work, we assumed a given model, abstracted as function $f_i(\cdot)$ in (3), with which the recommender system computes ratings. We used a CF recommendation as an example since this function can be written in analytic form. Our approach can also be applied to other recommendation models such as matrix factorization-based ones.

In this work, the action of users toward privacy included only the distortion of the true rating profile. A natural next step would be an enhanced strategy space where some viewed items could be concealed and not rated by the user, while for some others a distorted rating could be used. Furthermore, cooperative privacy preservation could involve the application of notions from cooperative game theory to study how users decide to form coalitions and mutually protect their privacy.

Our work can be viewed as an essential first step toward building more general models and setups where users strategically manage their personal information in order to preserve privacy in the presence of a central entity that aims to collect personal data so as to deliver some form of service after data processing. In our case, this central entity is the recommendation server that computes rating vectors and provides recommendations. Other settings where the same problem and model arise are also applicable such as that of a mobile app platform that collects personal medical, activity, location or other user data, and after some learning phase, it offers activity recognition, medical advice or location-based services.

Acknowledgements The authors wish to thank Mrs. Vassiliki Georgoudi and Mr. Panagiotis Spentzouris for their help with part of the numerical experiments.

References

1. Pennacchiotti M, Silvestri F, Vahabi H, Venturini R (2012) Making your interests follow you on Twitter. In: Proceedings of CIKM
2. Kempe D, Kleinberg JM, Tardos E (2003) Maximizing the spread of influence through a social network. In: Proceedings of KDD
3. Bobadilla J, Ortega F, Hernando A, Gutiérrez A (2013) Recommender systems survey. Knowl-Based Syst 46:109–132
4. Jeckmans A, Beye M, Erkin Z, Hartel PH, Lagendijk RL, Tang Q (2012) Privacy in recommender systems. In: Ramzan N et al (eds) Social media retrieval. Springer, London, pp 263–281
5. Halkidi M, Koutsopoulos I (2011) A game theoretic framework for data privacy preservation in recommender systems. In: Proceedings of the PKDD
6. Nisan N, Roughgarden T, Tardos E, Vazirani VV (2007) Algorithmic game theory. Cambridge University Press, Cambridge

7. Xu L, Jiang C, Chen Y, Ren Y, Liu KJR (2014) User participation game in collaborative filtering. In: Proceedings of the IEEE conferences signal and information processing (GlobalSIP)
8. Wang J, Yu L, Zhang W, Yu G, Xu Y, Wang B, Zhang P, Zhang D (2017) IRGAN: a minimax game for unifying generative and discriminative information retrieval models. In: Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval (SIGIR'17). ACM, New York, NY, USA. pp 515–524
9. Parra-Arnau J: Optimized, direct sale of privacy in personal-data marketplaces. arXiv:1701.00740
10. Melville P, Sindhwani V (2010) Recommender Systems, Encyclopedia of Machine Learning. Springer, New York
11. Resnick P, Iacovou N, Sushak M, Bergstrom M, Reidl J (1994) Grouplens: an open architecture for collaborative filtering of NETnews. In: Proceedings of computer supported cooperative work conference
12. Su X, Khoshgoftaar TM (2009) A survey of collaborative filtering techniques. Advances in Artificial Intelligence, Jan 2009
13. Linden G, Smith B, York J (2003) Amazon.com recommendations: item-to-item collaborative filtering. IEEE Intern Comput 7(1):76–80
14. Sarwar B, Karypis G, Konstan J, Reidl J (2001) Item-based collaborative filtering recommendation algorithms. In: Proceedings of the international conference on WWW
15. Balabanovic M, Shoham Y (1997) Content-based collaborative recommendation. Commun ACM 40(3):66–72
16. Mooney RJ, Roy L (2000) Content-based book recommending using learning for text categorization. In: Proceedings of ACM conference on digital libraries
17. Cotter P, Smyth B (2000) PTV: intelligent personalized TV guides. In: Proceedings of AAAI/IAAI,
18. Mellville P, Mooney RJ, Nagarajan R (2002) Content-boosted collaborative filtering for improved recommendations. In: Proceedings of the national conference on artificial intelligence
19. Erkin Z, Beye M, Veugen T, Lagendijk RL (2010) Privacy enhanced recommender system. In: 31st symposium on information theory in the Benelux, WIC 2010. IEEE Benelux Information Theory Chapter, pp 35–42
20. Nikolaenko V, Ioannidis S, Weinsberg U, Joye M, Taft N, Boneh D (2013) Privacy-preserving matrix factorization. In: Proceedings of the 2013 ACM SIGSAC conference on computer and communications security. New York, NY, USA
21. Miller B, Konstan JA, Riedl J (2004) Pocketlens: toward a personal recommender system. ACM Trans Inf Syst 22(3):437–476
22. Polat H, Du W (2003) Privacy-preserving collaborative filtering using randomized perturbation techniques. In: Proceedings of the international conference on data mining (ICDM)
23. Bilge A, Polat H (2012) An improved privacy-preserving DWT-based collaborative filtering scheme. Expert Syst Appl 39(3):3841–3854
24. Basu A, Vaidya J, Kikuchi H (2012) Perturbation based privacy preserving slope one predictors for collaborative filtering. In: Dimitrakos T, Moona R, Patel D, McKnight DH (eds) Trust management VI. Springer, Berlin, pp 17–35
25. Berkovsky S, Eytani Y, Kuflik T, Ricci F (2007) Enhancing privacy and preserving accuracy of a distributed collaborative filtering. In: Proceedings of the ACM RecSys
26. Shang S, Huiy Y, Huiz P, Cuff P, Kulkarni S (2013) Privacy preserving recommendation system based on groups. arXiv:1305.0540
27. Casino F, Domigo-Ferrer J, Patsakis C, Puig D, Solanas A (2015) A k-anonymous approach to privacy preserving collaborative filtering. J Comput Syst Sci 81(6):1000–1011
28. Lathia N, Hailes S, Capra L (2007) Private distributed collaborative filtering using estimated concordance measures. In: Proceedings of ACM RecSys
29. Canny J (2002) Collaborative filtering with privacy. In: Proceedings of the IEEE symposium on security and privacy
30. Shokri R, Pedarsani P, Theodorakopoulos G, Hubaux JP (2009) Preserving privacy in collaborative filtering through distributed aggregation of offline profiles. In: Proceedings of ACM RecSys
31. Kandappu T, Friedman A, Boreli R, Sivaraman V (2014) PrivacyCanary: privacy-aware recommenders with adaptive input obfuscation. In Proceedings of the MASCOTS
32. Bilge A, Polat H (2013) A scalable privacy-preserving recommendation scheme via bisecting k-means clustering. Inf Process Manag 49(4):912–927

Maria Halkidi is an Associate Professor at the Department of Digital Systems of University of Piraeus, Greece, since 2019. She obtained her degree in Informatics from University of Piraeus in 1997 and her M.Sc. and PhD degrees from Athens University of Economics & Business (AUEB) in 1999 and 2003, respectively. She has conducted research in AUEB participating in several national and European-funded projects and in University of California at Riverside as a Marie-Curie fellow funded by EU. Her research interests span diverse research areas of data and knowledge mining with emphasis on cluster validity, stream mining, social networks analysis, recommender systems. She has also participated in many program committees of international conferences in the areas of data mining and machine learning. She is a member of IEEE and ACM.

Iordanis Koutsopoulos is an Associate Professor at the Department of Informatics of Athens University of Economics and Business (AUEB) in Athens, Greece, since 2016. He received the Diploma degree in Electrical and Computer Engineering from the National Technical University of Athens (NTUA), Greece, in 1997 and the M.Sc. and Ph.D. degrees in Electrical and Computer Engineering from the University of Maryland, College Park (UMCP), MD, USA, in 1999 and 2002, respectively. He has served as Assistant Professor (2013-2016) with AUEB and Assistant Professor (2010-2013) and Lecturer (2005-2010) with the Department of Electrical and Computer Engineering, University of Thessaly. He received the single-investigator European Research Council (ERC) competition runner-up award for the project "RECITAL: Resource Management for Self-coordinated Autonomic Wireless Networks" (2012-2015). He is a senior member of IEEE. His research interests are in the general area of network control and optimization and in applications of machine learning, with application areas such as mobile crowdsensing, wireless networks, social networks, online platforms, smart energy grid and cloud computing systems.