What and who with: A social approach to double-sided recommendation

Published version:

DOI:10.1016/j.ijhcs.2017.01.001

What and Who with: a Social Approach to Double-Sided Recommendation

Ilaria Lombardi[a,*], Fabiana Vernero[a]

[a]*Univ. of Turin - Dept. of Computer Science - C.so Svizzera, 185 - 10149 Torino (Italy)*

Abstract

Double-sided recommendations (DSR) have been recently introduced for an item and a group that the item is destined for. Herein we present an algorithm which takes inspiration from The Social Comparison Theory to recommend items that had an average positive evaluation from other users on the target user's social network. Other users' judgments are weighted according to the influence these users have on the target. Moreover, for each recommended item, we propose a group that encompasses all the target users' contacts who expressed a positive opinion on it.

Our data show that users consider double-sided recommendations more useful than traditional recommendations which provide equivalent information. It was observed that our "social" DSR algorithm performs better in the event recommendation domain than a content-based one which has already been recognised as providing a good performance, in terms of precision, recall, accuracy and F1.This result is strengthened by our demonstrating that the good performance DSRs provide also depends on their peculiar structure and not only on the fact that they include "social" information. The item-recommendation part also performed better than a user-based collaborative filtering algorithm. Lastly, we found that users' scores for recommended item-group packages can be better predicted by considering only the system scores for the recommended groups, at least in the domain of social and cultural events.

Keywords: recommender systems, group recommendation, social network, user model, content-based recommendation, double sided recommendations

[*]Please address correspondence to Ilaria Lombardi
Email addresses: lombardi@di.unito.it (Ilaria Lombardi),
vernerof@di.unito.it (Fabiana Vernero)

1. Introduction

Everybody makes daily decisions, be they small or large, as to what to do and consume. During this decision making process, the presence of other people is usually implicitly or explicitly considered as part of the experience they are planning. For example when a choice has to be made as to which restaurant to eat at, cinema to go to or what film to see, planning holidays or just a weekend, or even deciding what to cook for dinner or collective buying practices. Even if all these activities are part of our everyday life, organizing them properly may turn into a complex task, since reasoning on different levels is involved, like which items to choose for consumption, who to involve and what items to consume with which people. Indeed, although most of these activities may usually be carried out with a predefined group of people (e.g., a person may normally dine in the family), it is not always so: let's take the example of organizing a dinner party where candidate guests may have different food preferences or vary on their ideas of a social event (e.g., intimate vs. large receptions) and be on different terms with each other.

Recommender systems emerged as a way to help users make choices (Jameson et al., 2014) by generating personalized suggestions, especially when there is too much information for a human decision maker to deal with effectively (the so-called problem of "information overload"). Recommenders usually follow either a *content-based* or a *collaborative filtering* approach. Content-based recommenders exploit some representation of users' interests and preferences and suggest items that appear to match it well, based on the idea that users will stick to their preferences and will continue to like items similar to those they liked in the past. Differently, collaborative filtering recommenders take advantage of rating data expressed by large numbers of users and suggest items that have been rated positively by users whose rating behaviour is similar to that of the target. Therefore, the *ratio* will be that users who agreed on their judgement for some items in the past will most likely have the same opinions on other items too. Most recommender systems suggest single items to single target users (Adomavicius and Tuzhilin, 2005). More advanced recommenders can either provide recommendations to groups (Jameson and Smyth, 2007) or suggest complex items, for example sequences or packages of items (Chao et al., 2005), friends (Hsu et al., 2006) or pre-existing groups of people (Baatarjav et al., 2008; Carmagnola et al., 2009). A previous study of ours (Vernero, 2011) proposed the concept of double-sided recommendations (DSRs), envisaging the need for an advanced recommendation technique which generates suggestions made up of an item and the group that the item should be consumed with. This matched situations with a strong social connotation, where the suggested items are at least as likely to be used by groups as they are by single individuals (as is the case of *group recommendations*), but where a group cannot be determined *a priori*. Herein we present an algorithm able to generate DSRs (hereafter referred to as DSR algorithm) according to the so-called Social Comparison-based Recommendation Method, that we first introduced, in terms of high-level concepts and building blocks, in (Vernero, 2011). This method was chosen as it mainly uses information drawn from the target user's social network. In fact, relevant literature (Sinha and Swearingen, 2001), as well as our own previous research (Carmagnola et al., 2009), have evidenced how people tend to place a high value on recommendations based on the opinions and preferences of friends and other trusted users.

With the aim of validating our algorithm, we embraced a piece-wise approach and carried out a set of empirical evaluations able to make an assessment from various points of view and in different domains. Our goals were:

- to evaluate the usefulness of DSRs generated by our "social" DSR algorithm and to compare them with other, more traditional, kinds of recommendations;

- to understand whether the item-recommendation part of our DSR algorithm has comparable performance, in terms of standard measures, such as precision and recall, to a collaborative filtering algorithm, in the context of a large offline study;
- to understand whether, in the context of an online user study, our DSR algorithm performs better than a traditional content-based one which makes use of detailed information on the target user's preferences proved to have a good performance with a consolidated user model. The algorithm is described in detail by Carmagnola et al., 2008 and Gena et al., 2013 for information on its evaluation;
- to assess how much importance users give to the recommended item and group, respectively, when they are evaluating a DSR in a particular domain (we surmised that user preferences might depend on the type of suggestion they receive). This information is useful to understand how these two elements should contribute to the overall predicted score for a recommendation.
- to verify that the efficacy of DSRs does not depend solely on the fact that they provide information obtained from the target user's social network (to this aim, we compared the full DSR algorithm with its item-recommendation part).

Our empirical evaluations targeted three different domains, i.e. collaborative buying practices (in the context of a fictional system, first goal), music (second goal) and participation in social events (third, fourth and fifth goals) respectively. iCITY, a social adaptive recommender system we developed in the past, was used for the event domain, as a use-case.

Briefly, this paper makes a two-fold contribution: on the one hand, it proposes a new "social" algorithm for the generation of DSRs, starting from the concepts we introduced in a previous study of ours; on the other, it presents the results of the evaluations we carried out so as to assess it.

The paper is organized as follows: a presentation of the background information on DSRs is provided in Section 2. Section 3 reports on our work within pertinent research, whilst Section 4 presents our approach, detailing the algorithms. Section 5 discusses and describes our experiments and their results. Section 6 concludes the paper.

2. Background

The concept of DSRs was first introduced in Vernero (2011) to extend the scope of recommender systems to situations where the target user may need to be suggested, not only a personally appealing item, but also a group of people that the item could be consumed with. DSRs were formally defined as follows:

Definition 1 *Given a set of users U, a target user* $t \in U$*, a set of contacts of the target user* $N_t \subseteq U$ *and a set of candidate items I, we call a* double-sided recommendation (DSR) *a pair* $(i, G_t(i))$ *where* $i \in I$ *and* $G_t(i) \subseteq N_t$.

So as to generate the DRSs, we proposed a high-level framework identifying four macro-situations where this kind of suggestion may be appropriate.

Such situations differ depending on the contextual and occasional elements and/or according to the target users' personal preferences:

1. users are looking for an item to enjoy with some of their contacts. They value the opinions of their contacts and would like to know what they would do in their place;
2. users are interested in spending some time in good company and would like to find an item which can please all the people they will meet;
3. users are interested in enjoying a pleasant item and would like to know which of their contacts might keep them company;
4. users are interested in enjoying an item in company and the choice of both a suitable item and good company bear the same weight.

Each situation sets different priorities, either on the group or the item side. Our previous paper did not go into detail as to the specific solutions to the algorithms, but rather described an approach for each situation. The approaches proposed basically fall into two groups (Vernero, 2011):

- The **Social-Comparison-Based method** (matches: the first situation), which takes inspiration from psychological theories on social influence dynamics1 and previous research done by one of the authors (Carmagnola et al., 2009), generates suggestions based on the preferences of people within the social network of the target user. More specifically, it suggests consuming items that are appreciated, on average, by relevant contacts, in the company of those very same contacts who have given them a positive evaluation. Consequently, this method does not necessitate detailed information as to the target user's preferences or relationships among the members of his/her social network.
- **Component-Based methods** (match: the situations two to four) are three variants of the same overall process, but differ as to the importance assigned to items and groups. All Component-based methods are based on a) the identification of structural subcomponents (i.e., connected components) in the social network of the target user so as to build recommendable groups, and b) the exploitation of user models generated through a content-based approach to predict individual user preferences for candidate items.

Emphasis was placed on general concepts and methods in our previous paper, Vernero (2011). Conversely, herein we present a specific algorithm able to generate DSRs. Indeed, although algorithmic solutions were devised for the evaluation made in Vernero (2011), they did not represent the focus of our work and were, therefore, not published. The main novelties introduced in this paper, compared to the contents reported in Vernero (2011), are summarized in Table 1.

3. Related work

The paper by Stefanidis and Pitoura (2013) is the most representative of the idea of DSRs proposed herein, which focused on the suggestion of vacation packages and

[1] According to the Social Comparison theory, which inspired this recommendation method, people who are in a state of uncertainty as to what they should be thinking or doing actively seek information about the opinions of others. (Festinger, 1954; Suls et al., 2002).

addressed the problem of forming a group of users which is an appropriate target for the recommended packages. The authors proposed a greedy group construction algorithm that took into account a number of constraints as to the users' liking for the item, as well as group composition. While DSRs are directly aimed at a single target user who will consume the suggested item with the recommended group, this paper provided a service to be used by travel agencies. However, travel agencies will then recommend a certain vacation package and the rest of the group of co-travellers to each member of the generated group, thus matching a DSR scenario perfectly.

The whole area of *group recommender systems* is also closely related to DSRs: indeed, such systems recommend items to a group of people on the basis of the idea that some types of items with a strong "social" characterization (for example vacations, movies or restaurants) are at least as likely to be consumed by groups as they are by individuals, which is also a starting point for double-sided recommendations. Recent examples of group recommender systems suggest movies and TV programmes (gRecs (Ntoutsi et al., 2012)), music (Groupfun (Popescu and Pu, 2012), home-cooking recipes (Berkovsky and Freyne, 2010) or even books (Kim et al., 2010). Moreover, Stefanidis et al. (2012) widen the scope of group recommendations by presenting a context-aware recommendation model which takes into account factors such as the weather and time period, as well as providing examples in the movie-recommendation domain.

The main difference between group and double-sided recommendations is that there is no target group available for the latter *a priori*: indeed, double- sided recommender systems define different group options automatically and these are then presented to the target user as part of the generated suggestions. To this aim, Boratto and Carta (2011) list four different types of groups which have been addressed in the literature on group recommender systems: *established* ones (i.e., persistent groups where members share common interests and have actively decided to join the group), *occasional* ones (i.e., groups formed by people who share a common goal at a specific time, such as planning a vacation), *random* ones (i.e., groups of individuals who simply happen to share an environment at a certain moment in time) and *automatically identified* ones. This last type fits into a borderline scenario where groups of people that, despite their common interests, are otherwise unrelated one to the other, are targeted rather than single individuals, because of technological constraints. One such case is that of Mobile IPTV systems with limited transmission capacity, where it would be impossible to create a personalized schedule for each user (Boratto et al., 2009). To a certain extent, it can be considered that DSRs target automatically-identified, occasional groups of users, who simply share an interest at a certain point in time, provided they are all part of the target user's social network.

The inclusion of social aspects, such as social relationships and personality types is constantly attracting more attention in the area of group recommenders. A simple way to take into account such aspects is that of treating group members differently, for example by giving higher priority to the preferences of some "special" group member, e.g., a guest or a person with particular needs (Ardissono et al., 2003). Considering that recommended items usually carry a different weight for different group members, Amer-Yahia et al. (2009) propose to formalize group disagreement as an integral part of group recommendation semantics. In particular, they suggest a consensus score for each item be defined, which is a result of both *group relevance* (how much such an item is liked by group members) and *group disagreement* (how much group members disagree with one other). Gartrell et al. (2010) took these concepts further, by scoring recommendations based on a group consensus function which takes into account relationship strength among group members, as well as expertise and interest dissimilarity. Another relevant approach is described by Recio-Garcia et al. (2009), where the authors try to anticipate the way people with different personalities (assertive or cooperative) may react to compromise recommendations by integrating a factor in the final item selection, which gives more importance to the preferences of assertive users. In later research, done by Sanchez et al. (2013), the authors extended their approach by taking into account also the social connections among group members. This was based on the assumption that users' preferences and choices may be biased by those expressed by close friends, either because they actively create their opinions according to those of other, trusted users, or because they may modify their pre-existing preferences on the basis of social influence dynamics (social trust). Similarly, Christensen and Schiaffino (2014) remark that, in real-life scenarios, group members are influenced by one another's opinions and are often willing to change their initial choices to favour the satisfaction of the group as a whole. Consequently, they propose a method which generates group suggestions by aggregating "influenced individual preferences", that is, preferences that take into consideration the type of relationship and interest similarity between every two members, as well as the network centrality of the party who plays the influencer's role (in fact, people having a central position in a social network are assumed to be more influential than peripheral ones). All the "socially aware" strategies for the aggregation of group member preferences reported herein can also be used in DSRs to assess the suitability of the recommended item to the group as a whole, with the aim of making group recommendations more realistic. For example, in our algorithm, we select recommendable items by balancing users' preferences according to their influence on the target user, an approach which is close to the ideas of "social trust" and "influenced individual preferences" expressed by (Sanchez et al., 2013) and (Christensen and Schiaffino, 2014), respectively.

However, our "social" approach was explicitly inspired by SoNARS, an algorithm described by Carmagnola et al. (2009), which targets single users, providing them with suggestions of items that really reflect the trend of their social networks. SoNARS itself uses a variant of collaborative filtering to predict the target user's preferences, where members of the target user's social network are considered rather than generic users and a relevance measure, which considers the number and type of social interactions (relationship strength), is used to balance the impact different people have on the final prediction. Differently from the algorithm proposed herein, SoNARS was more inspired by the idea that simply taking part in social relationships may make individuals modify their attitudes and behavior, than by the fact that some kinds of items are usually consumed with other people. Moreover, our DSR approach extends SoNARS algorithm by considering also the "value" of actions and not only their number and weight in our computation of the relevance items and other people have on the target user. Lastly, we also introduce interest similarity alongside relationship strength, as a determinant of user influence in the recommendation generation process.

A similar approach to that adopted by SoNARS was presented in (Guy et al., 2009), where the authors compared recommendations derived from the target users' similarity network, as in traditional collaborative filtering, to recommendations derived from their familiarity network, that included the people they know. A social network aggregation system called SONAR was used to extract information on relationships between people and people, on the one hand and people and items, on the other, from different sources.

Apart from their freshness, candidate items were scored according to the number of people in the target users' social network who were somehow related to them and to the strength of the relationships between each of these people and a) the target user, and b) the item itself. The authors reported that recommendations generated from the familiarity network consistently achieved the best performance. Moreover, there was a statistically significant difference as to recommendations based on the similarity network whenever case explanations that evidenced which people were related to each item suggested were provided, compared to when they were not.

Martinez-Cruz et al. (Mart´ınez-Cruz et al., 2015) also departed from the traditional collaborative filtering approaches when they proposed the exploitation of trust networks. They suggested that trust scores for any two users can either be explicitly assigned by the users themselves, or estimated on the basis of trust scores found along the paths that connect them . Running an offline experiment on a large dataset from Epinions.com, they observed that recommendations based on their trust network outperformed traditional collaborative filtering algorithms, whether item or user-based, as far as accuracy and coverage (i.e., the proportion of target items an algorithm can generate a prediction for) are concerned. Noteworthy is the fact that, in comparison to double-sided or group recommendations, although the notion of trust refers to user reliability or expertise, here it does not imply an actual social contact.

Recommendations generated according to the preferences and activities of actual acquaintances have become very popular with social networks. For example, Berkovsky et al. (2012) proposed an algorithm that suggests the use of personalized news feeds to visitors of an online diet portal, where candidate news items are scored according to a linear combination of the relevance scores for the users they reference and the actions they mention (e.g., posting a comment). Comparing personalized and non-personalized news feeds, the authors found that, not only does personalization provide users with more interesting contents, but it also promotes user activity within the social network itself. In a more recent paper by Guy et al. (2015), emphasis was placed on an enterprise social network to assess a personalization model which suggests news items based on the entities (e.g., blog posts or wiki pages) people and keywords they are related to, or mentioned, which they have in common with the target user's profile. Comparing their approach to a simpler algorithm that suggested news items because they were related to either popular people or popular entities, the authors observed that personalized recommendations were significantly more interesting. However, recommended popular items were more often unknown to the target users and led to more serendipitous suggestions.

Table 1
Comparison between Vernero (2011) and the current paper: novelties and contributions.

Vernero (2011)	Current paper
Introduction of the DSR concept	
Presentation of 4 instances of the DSR problem	Focus on the problem instance where users value the opinion of their contacts
Description of 4 general methods able to generate DSRs	Presentation of an algorithm to generate DSRs according to the Social-Comparison-Based method
Evaluation goals:	Evaluation goals:
• Testing DSR usefulness;	• Testing DSR usefulness in comparison to single-sided recommendations;
• Comparing the performances of the 4 methods.	• Comparing the performances of the DSR algorithm with single-sided recommendation algorithms;
	• Assessing the contribution of the recommended item and group in determining user preferences for DSRs.

4. Double Sided Recommendation - a Social Approach

In the Social Comparison-based method (Vernero, 2011), target users are recommended items that were positively evaluated, on average, by other users in their social network, e.g., contacts. The users' judgment is weighted according to the influence they have on the target user, which is a function of the similarity and relevance of a certain contact for the target. For each recommended item, a group made up of all -and exclusively- the target user's contacts who expressed a positive opinion about it, is suggested for the generation of DSRs. Moreover, aiming at explaining recommendations and leveraging social influence, all the contacts who evaluated the recommended item are listed.

Starting from the ideas described by Vernero (2011), we propose a “social” DSR algorithm which:

1. selects items to recommend based on a score called collective item relevance (see Section 4.2.2), which takes into account the opinions (item relevance, Section 4.1.2) expressed by the target user’s contacts who “reviewed” a certain item (reviewers) and their influence on the target user (user influence, Section 4.2.1).
2. for each recommended item, it:
    a. provides an assessment of how relevant the reviewer’s opinions are on average for the target user (*social confidence*, Section 4.2.3);
    b. generates a recommended group of people who the item could be enjoyed with by selecting the reviewers who have expressed a positive opinion about it (see Section 4.3).
    c. assigns a score to the recommended group which indicates how good it is for the target user (group score, Section 4.3);
3. orders the so-obtained DSRs (item plus group) based on their total score (see Section 4.4), which combines collective item relevance and group score.

We shall explain how each of these measures can be calculated. As a use case, reference will be made to a fictional system which supports collaborative buying practices, where users can perform several actions to express their liking or disliking for a certain item or user, and, in particular:

- on items: $\Theta_i = \{$Bookmark, Rate, Tag, Comment$\}$

- on other users: $\Theta_u = \Theta_i \cup \{$Message, Friend$\}$.

Moreover, we assume that user preferences for categories of items are stored in a user model as a distribution of values which correspond to categories in a taxonomy representing an overlay model of the domain2 (Carmagnola et al., 2008). Each category in the taxonomy identifies a type of food (e.g., wine, cheese, fruit, cereals). All our examples will make reference to the fictional DUCKBURG world (users, actions, relationships among users) presented in Tables 2-3-4.

[2] Methods to create and update the user model are beyond the scope of this paper.

Table 2
User actions for the characters in the fictional DUCKBURG world.

	Pies	Candies	Peanuts	Cheese	Salad	Donuts	Apples	Fish
Gus Goose	♥ ★ 4 ▬ 5	♥ ★ 4 ▬ 5			★ 1	♥ ★ 4		
Louie	♥ ★ 4 ▬ 5 ✎ 3	♥ ★ 4 ▬ 5 ✎ 3	▬ 3		★ 1	♥ ★ 3 ▬ 1		
Donald Duck	♥ ★ 4	♥ ★ 4 ▬ 5				♥ ★ 4 ▬ 5		
Daisy	♥ ★ 4 ✎ 2	★ 3	★ 2		♥	★ 4 ▬ 5	♥ ★ 4 ▬ 5 ✎ 3	★ 3 ✎ 1
Scrooge	♥			♥		♥	♥	

♥ bookmark; ★ rate; ▬ number of tags; ✎ number of comments.

Inferring Item and User Relevance from User Actions

As detailed by Kobsa et al. (2001), user actions can be considered an indicator of their interests and preferences. This section shows how we infer item relevance and user relevance from user actions. *Item relevance* indicates how important an item i is for a user u who “reviewed” it, i.e., performed one or more actions on it (hereafter these users will be referred to as “reviewers”), while user relevance measures the strength of the social relationship between the target user t and one of her contacts u.

Table 3
User relevance for characters in the fictional DUCKBURG world.

	Gus Goose	Louie	Donald Duck	Daisy	Scrooge
Gus Goose	–	1	1	1	1
Louie	1	–	1	1	1
Donald Duck	1	1	–	1	1
Daisy	1	1	1	–	1
Scrooge	1	1	1	1	–

Table 4
Users' interest for domain categories in the fictional DUCKBURG world.

	Pies	Candies	Peanuts	Cheese	Salad	Donuts	Apples	Fish
Gus Goose	0.3	0.3			0.1	0.3		
Louie		0.6	0.1			0.3		
Donald Duck	0.5					0.5		
Daisy	0.15		0.05		0.3		0.3	0.2
Scrooge	0.4			0.3		0.2	0.1	

User Actions

Both a value and a weight can be distinguished for each type of action. The action-type value synthesizes user opinions as to the elements acted- upon: for example, assuming that a four-point (1 to 4) rating scale is adopted, users are expected to prefer items they rated as 4 rather than items they rated as 2. On the contrary, the action-type weight relates to action effectiveness as an indicator of user interests. Differently from the action-type value, which is specific to a certain user-element couple, the weight depends exclusively on the type of action.

Action-type value. The action-type value derives from the number of actions of a certain type a user performs on a certain element and from the value of each single action (action value). The following rules were defined so as to assign values to user actions:

rating: the action value corresponds to the rating itself;

bookmarking/befriending : the action value corresponds to the maximum possible rating value (in our fictional system, 4), since these actions usually indicate high appreciation[3];

commenting/tagging/sending a message[4]: if the element was also rated by the same user, then the action value corresponds to the rating. Otherwise, if it was bookmarked (and/or made friend in case of users), then the action value corresponds to the maximum possible rating value. Otherwise, the action value corresponds to the average value intended as a neutral value. This is the average between 1 and 4, i.e. 2.5 in our fictional system.

Table 5
Maximum action value, threshold action numbers and assigned weights for action types.

Action type	Maximum value v_θ	Threshold n_θ	Weight α_θ
Rating	4	1	0.6
Bookmarking	4	1	0.9
Befriending	4	1	1
Tagging	4	2	0.8
Commenting	4	2	0.8
Sending message	4	2	1

Moreover, in the case of tagging, sending a message or commenting, users can perform more than one action on the same item or user and the number of actions itself can be considered an indicator of user opinions. For example, *ceteris paribus* users are expected to be more interested in items they have made 10 comments for, rather than items they only made one comment for.

Therefore, using these argumentations as a starting point, we calculated an action-type value, ranging from 0 to 1 for each action type $\theta \in \Theta_i \cup \Theta_u$, collecting the values of single actions and normalizing them by comparing them to the product of the maximum action value (v_θ) and a number of actions that we assume to be a threshold beyond which the user interest is ascertained (n_θ). Our system sets those values at 1 for most actions and at 2 for the actions of tagging, commenting and sending messages, which can be performed repeatedly on the same item (see Table 5). For example, consider the tag action: we assume that 2 tags with action value 4 are a clear indication of a user's maximum interest in a certain element.

Formally, the *action-type value* is defined as follows:

Definition 2 *Given an action type θ, the* action-type value *of an element* e *for a user* u *is*

$$\nu_{\mathrm{u}}^{\theta}(\mathrm{e}) = min\left(1,\ \frac{\sum_{\mathrm{a}\in\mathcal{A}_{\mathrm{u}}^{\theta}(\mathrm{e})} \nu(a)}{v_\theta n_\theta}\right).$$

3 Note that, in some domains, bookmarking an item or befriending a user might not always indicate real interest. For example, Twitter troll accounts might interact with other users to make them visible for others to attack. However, we do not consider malicious users as valuable targets for our algorithm.

4 Assigning a value to comments, messages and tags would require some sort of sentiment analysis to determine whether they express a positive or a negative opinion. However, to simplify things, we assumed that users behave coherently, therefore, the value of these kinds of actions may be inherited from other actions that were performed by the same user on the same item (or user).

where

- $A^{\theta}_{\mathrm{u}}(\mathrm{e})$ *is the set of the actions of type θ that user* u *performed on element* e;

- $v(\mathrm{a})$ *is the action value for action* a*;*

- v_{θ} *is the maximum action value that a user can assign to an action of type θ;*

- n_{θ} *is the number of actions of type θ that we assume gives evidence of the user's interest in the element.*

As an example, consider the item Donuts and the user LOUIE in the fictional DUCKBURG world, who bookmarked, assigned a rating of 3 and added a tag to Donuts (see Table 2).

$$\nu_{\mathbf{LOUIE}}^{\mathrm{Bookmark}}(\mathrm{Donuts}) = min\left(1, \frac{4}{4 \cdot 1}\right) = 1$$

$$\nu_{\mathbf{LOUIE}}^{\mathrm{Rate}}(\mathrm{Donuts}) = min\left(1, \frac{3}{4 \cdot 1}\right) = 0.75$$

$$\nu_{\mathbf{LOUIE}}^{\mathrm{Tag}}(\mathrm{Donuts}) = min\left(1, \frac{3}{4 \cdot 2}\right) = 0.38$$

Action-type weight. According to Kobsa et al. (2001), different types of actions are more or less reliable indicators of users' actual interests in a certain element. For example, a rating which requires the users to make less effort is usually considered less informative than is a comment. Therefore, we assigned different weights $\alpha \in [0, 1]$ to them, so that more informative actions have a greater impact on item and user relevance than do less informative ones. Table 5 shows the specific actions and weights used, which were chosen according to our past experience with the SoNARS algorithm (Carmagnola et al., 2009) and iCITY (Carmagnola et al., 2008) and which have already been tested for our previous paper (Vernero, 2011), although they were not published.

Item and user relevance

On the basis of *action-type values* and *weights* we propose a method to estimate how relevant an element, be it an item or another user, is for a single user.

Definition 3 *For a user* u ∈ *U, the* relevance *of an element* e ∈ I∪ (U \{u}) *is:*

$$\mathrm{REL}(\mathrm{e}, \mathrm{u}) = \frac{\sum_{\theta \in \Theta} \alpha_{\theta} \nu_{\mathrm{u}}^{\theta}(\mathrm{e})}{\sum_{\theta \in \Theta} \alpha_{\theta}}.$$

where

- Θ *is the set of actions that the user* u *can perform on* e*. It is* Θ_i *if* e *is an item, and it is* Θ_u *if* e *is a user;*

- α_θ *is the weight of the action type θ as described in Table 5;*
- v_u^θ (e) *is the value assigned to the actions of type θ performed by the user* u *on the element* e *(see Eq.* (1)*).*

Note that relevance is an asymmetric measure, therefore, in the case of two users, u and t, rel(u, t) might differ from rel(t, u).

As an example, the relevance of the Donuts for LOUIE and the other Duckburgers (refer to Table 2) is:

$$\text{REL(Donuts, LOUIE)} = \frac{0.9 \cdot 1 + 0.6 \cdot 0.75 + 0.8 \cdot 0.13}{0.9 + 0.6 + 0.8 + 0.8} = 0.47$$

$$\text{REL(Donuts, GUS GOOSE)} = 0.48$$

$$\text{REL(Donuts, DONALD)} = 0.74$$

$$\text{REL(Donuts, DAISY)} = 0.45$$

Recommending Items

This section will explain how to select items to include in DSRs on the basis of a measure called *collective item relevance*. This estimates the importance item i has for the target user t by combining *item relevance* (see Section 4.1) and *user influence* taken for each one of the target user's contacts who "reviewed" item i.

User influence

The *user influence* $\text{infl}_t(u)$ indicates how much importance should be placed on the opinion of a certain contact u in determining a suggestion for the target t. *User influence* ranges [0,1] and is simply computed as an average of (i) *user relevance* rel(u, t), indicating how relevant u is to t (see Section 4.1.2, Eq. (2) and (ii) *pairwise similarity*.

Pairwise similarity (range: from 0 to 1) indicates to what degree two users (in our case, the target user t and a contact u in her/his network) can be considered to have similar interests in domain categories. Based on our previous experience with iCITY (Carmagnola et al., 2008) and according to extensive evidence extrapolated from international literature (Schafer et al., 2007), pairwise similarity was defined as a variation of the formula for the standard deviation, where the values stored in user's u user model and indicating her interests substitute the expected values.

Definition 4 *The* pairwise similarity *between a user* u *and a user* t *is*

$$\text{SIM}_t(u) = 1 - \sqrt{\frac{\sum_{c \in C} (int_t(c) - int_u(c))^2}{|C|}}.$$

where:

- *C is the set of the domain categories the user model is defined on;*
- $int_t(c)$ *is the interest the user* t *has for the category c as defined in her user model;*
- $int_u(c)$ *is the interest the user* u *has for the category c as defined in her user model;*

If a user has no interest in a category, the corresponding default interest value is 0. Note that, since standard deviation is a dissimilarity measure, which reaches its minimum value, 0, if the interests of two users are perfectly equivalent, the complementary value is calculated by subtracting the standard-deviation based measure from 1 so as to obtain a real similarity measure. Moreover, $sim_t(u) = sim_u(t)$.

For example, for SCROOGE and the other Duckburgers the *pairwise similarity* is:

$$\text{SIM}_{\text{SCROOGE}}(\mathbb{LOUIE}) = 1 - \sqrt{\frac{0.16 + 0.36 + 0.01 + 0.09 + 0 + 0.01 + 0.01 + 0}{8}} = 0.72.$$

$$\text{SIM}_{\text{SCROOGE}}(\mathbb{GUS\ GOOSE}) = 0.83$$

$$\text{SIM}_{\text{SCROOGE}}(\mathbb{DONALD}) = 0.84$$

$$\text{SIM}_{\text{SCROOGE}}(\mathbb{DAISY}) = 0.79.$$

Lastly, we consider that user relevance and pairwise similarity are equally important in determining user influence and this measure only reaches its maximum if both its components have also reached their maximum. The influence a user u has on a user t is calculated as follows:

$$\text{INFL}_t(u) = \frac{\text{REL}(u, t) + \text{SIM}_t(u)}{2}.$$

In our example, the citizens of the same town are friends and all the Duckburg inhabitants interact with one another so that their mutual relevance is maximal, i.e. 1. The *user influence* of each Duckburger on the target user SCROOGE is:

$$\text{INFL}_{\text{SCROOGE}}(\mathbb{LOUIE}) = \frac{1 + 0.72}{2} = 0.86$$

$$\text{INFL}_{\text{SCROOGE}}(\mathbb{GUS\ GOOSE}) = 0.92$$

$$\text{INFL}_{\text{SCROOGE}}(\mathbb{DONALD}) = 0.92$$

$$\text{INFL}_{\text{SCROOGE}}(\mathbb{DAISY}) = 0.90$$

The following section shows how *item relevance* and *user influence* can be combined to obtain *collective item relevance*.

Collective Item Relevance

The *collective relevance* for an item i is a value ranging from 0 to 1 and, similarly to what happens in user-based collaborative filtering recommendations, summarizes the opinion that the network of contacts of the target user t has on i. It is calculated for each item $i \in I$ so as to determine if i should be recommended to t or not. More specifically, whenever the collective relevance of the item i exceeds a given threshold, i is included in the recommendation list.

Definition 5 *The* collective relevance *of an item* i *for the network of a target user* t *is*

$$\overline{\text{REL}}(i, t) = \frac{\sum_{u \in \mathcal{N}_t} \text{INFL}_t(u)\text{REL}(i, u)}{\sum_{u \in \mathcal{N}_t} \text{INFL}_t(u)}.$$

where:

- N_t *is the set of users that have some kind of user-to-user relationship with the user* t*;*

- rel(i, u) *is the relevance the item* i *has for the user* u *as defined in Eq.* (2)*;*

- $\text{infl}_t(u)$ *is the* user influence*, i.e. a weight with a value that depends on the importance of* u *for the target user* t*, as explained in Section 4.2.1.*

For example the *collective item relevance* of Donuts for SCROOGE is:

$$\overline{\text{REL}}(\text{Donuts}, \textbf{SCROOGE}) = \frac{0.86{\cdot}0.53 + 0.92{\cdot}0.48 + 0.92{\cdot}0.74 + 0.89{\cdot}0.45}{0.86 + 0.92 + 0.92 + 0.89} = 0.55.$$

Noteworthy is the fact that only judgments of users who actually exert some influence on the target user (i.e. with $\text{infl}_t(u) > 0$) contribute to the collective item relevance.

Items having *collective item relevance* greater than a certain threshold are selected for recommendation, while the others are discarded as they are not considered to be relevant for the target user. We set such a threshold at the average of not null collective item relevance, as suggested by Basu et al. (1998). The *reviewers' group* includes all the target user's contacts that reviewed the item.

Social Confidence

Social confidence is a score ranging from 0 to 1 that accompanies the list of reviewers for a certain item. Defined as a function of the reviewers' *user influence* on the target user, it shows the level of confidence associated to the corresponding item recommendation, based on the idea that suggestions depending on the opinions of reviewers who have very similar tastes to the target user's and are very close to them, are more likely to be accurate.

Definition 6 *Given a set* $R^i_t \subset N_t$*, which is the set of target user's contacts that performed at least one action on* i*, its* social confidence *for the target user* t *is:*

$$\text{CONF}_t(i) = \frac{\sum_{r \in \mathcal{R}^i_t} \text{INFL}_t(r)}{|\mathcal{R}^i_t|}.$$

Recommending groups

Our algorithm suggests a group of people belonging to the target user's social network for each recommended item that can be enjoyed with them. More specifically, recommended group members are those reviewers who have expressed a positive evaluation on the corresponding item. The threshold set to consider an individual evaluation as "positive" was set at 0.6[5].

Definition 7 *Given an item* i *recommended to a target user* t*, the corresponding* recommended group *is:*

5 In a scale ranging from 0 to 1, 0.6 corresponds to sufficiency in the commonsense view.

$$\mathcal{G}_t(\mathrm{i}) = \{r \in \mathcal{R}_t^{\mathrm{i}} | \mathrm{REL}(\mathrm{i}, r) \geq 0.6\}.$$

Note that, in real-world scenarios, including all the reviewers who have expressed a positive evaluation may make the recommended group too numerous. Therefore, in such cases, considering further constraints for group member selection may help limit the group size, whilst, at the same time, improve group cohesiveness. For example, it may be preferable to include candidate members who are directly related to other group members or who have similar interests to the rest of the group, based on the fact that recommendations have been proven to be more satisfying when group members have similar preferences (Baltrunas et al., 2010). However, in this case, we decided not to add further constraints, as in our previous research (Vernero, 2011), we observed that the recommended groups usually had an affordable size in relatively small-scale evaluations.

Given a recommended group $G_t(i)$, a *group score* ranging from 0 to 1 is determined to indicate how good it is for the target user. *Group score* is a function of user relevance and other optional parameters which vary accord- ing to the domain. These may include some measure of cohesiveness among group members, of user proximity or of their trust. In this case, for simplicity, we determined *group score* as the average of individual values for *user relevance* :

$$\mathrm{GSC}_t(\mathrm{i}) = \frac{\sum_{\mathrm{g} \in \mathcal{G}_t(\mathrm{i})} \mathrm{REL}(\mathrm{g}, \mathrm{t})}{|\mathcal{G}_t(\mathrm{i})|}.$$

Lastly, for each recommended group $G_t(i)$, we compute a value indicating how good the corresponding recommended item is expected to be for the group as a whole, determined simply by averaging the values of item relevance rel(i, g) for each group member $g \in G_t(i)$. Amongst the numerous possible aggregation strategies, we decided to focus on the prediction of the average satisfaction, since it is the one which is most commonly used (together with least misery, see (Sanchez et al., 2013); at the same time, the possibility that there are extremely dissatisfied members is avoided *a priori* by putting only those users who positively evaluated the recommended item into the recommended group.

Recommendation total score

Item and group scores are synthesized into a single value, called *total score* tot, which is calculated as a weighted average of *collective item relevance* $\mathrm{rel}_t(\mathrm{i})$ and *group score* $\mathrm{gsc}_t(\mathrm{i})$. In this definition phase we do not set the weight values as we believe they depend on the application domain. In one of our experiments (see Section 5.3) we actually focus on the event recommendation domain and apply linear regression to calculate such weights as general parameters for all users[6]. For presentation purposes, the DSRs are ordered according to their *total score* and, so as to facilitate the target user in assessing recommendations, the values of collective *item relevance*, *group score* and *average group satisfaction* are explicitly provided, as is the list of *reviewers* with its value for *social confidence*.

5. Evaluation

[6] Different weights might also be applied for each user, depending on preferences, but such a personalized approach is beyond the scope of this paper and may be subject of future research.

A set of four different evaluations were used to assess our approach to DSRs. Firstly, a preliminary test was done, focusing on the goal of supporting users in adopting collaborative buying practices, with the aim of evaluating the usefulness of DSRs. More specifically, the aim was that of answering the following research question:

RQ1: are DSRs generated with the "social" DSR algorithm herein proposed more useful than traditional recommendations that have either a group or an item, but provide equivalent information?

The goal of objectively assessing the performance of the item-recommendation part of the DSR algorithm was then focused on, comparing it with standard collaborative filtering in the context of a large offline study. This evaluation was motivated by two factors: 1) the DSR algorithm as a whole cannot perform well if its main subcomponent (i.e., the item-recommendation part) does not perform well; 2) since there are no public datasets containing user evaluations of items and related groups, concentrating on the (single-sided) item-recommendation part was the only way to compare our approach to standard ones, on large numbers. Our goal can be summarized in the following research question:

RQ2: does the item-recommendation part of the DSR algorithm perform better than standard user-based collaborative filtering?

A third online evaluation was then carried out on the event recommendation domain to 1) assess the performances of our algorithm in terms of standard measures and 2) determine the relative importance of the group and item components as to our focus domain. Namely, we posed the following research questions:

RQ3: does the DSR algorithm, carried out in the context of cultural event recommendation, perform better than a content based algorithm that is known to have good performance with a consolidated user model?

RQ4: to what extent are *collective item relevance* and *group score* able to contribute to the recommendation process in the event recommendation domain?

Lastly, as the verification of the efficacy of DSRs does not depend exclusively on the fact that they provide information obtained from the target users' social network, we devised a follow-up online experiment and posed the following research question:

RQ5: does the full DSR algorithm perform better than its item-recommendation part, given that both provide information as to the users' opinions recommendations are based on?

Preliminary Evaluation

Our preliminary evaluation, carried out in the domain of collaborative buying, compared DSRs with single-sided recommendations, suggesting either products to buy or friends who already participated in buying groups. It also provided comparable information in the form of explanations. The focus was placed on the type of service and information provided by each type of recommendation rather than on the validity of the prediction.

To this purpose, starting from the data in the DUCKBURG world (see Tables 2-3-4), two hypothetical lists were generated for the "target user" (List A and List B). Each one contains three recommendations, where each recommendation is characterized by the recommended product and an explanation including the collective item relevance, the list of reviewers, the social confidence and, only for list B, the recommended group and the group score.

Figure 1 shows List B, while List A is similar but it does not contain information on the suggested group or its evaluation.

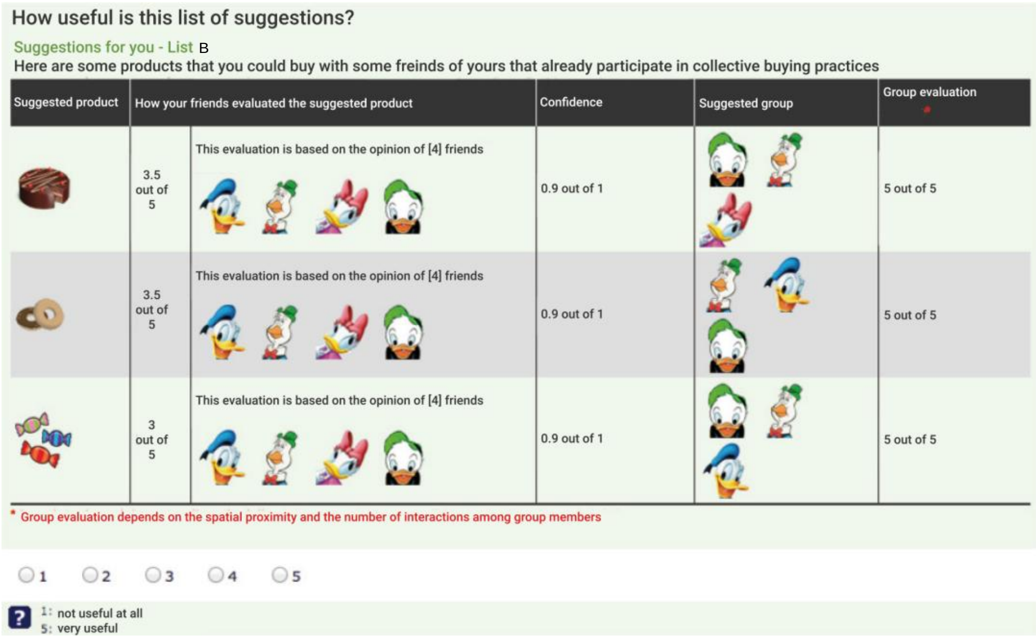

Figure 1: Preliminary evaluation: List B (double-sided recommendations).

Procedure. The two lists were evaluated with a set of 40 users[7] (selected among friends and students at the University of Turin by means of an availability sampling strategy) in the context of an online survey. Firstly, the users were asked to imagine that they had just registered for a website supporting collaborative buying practices with the aim of forming buying groups and of purchasing food products they were interested in - the website also supported them by providing lists of recommended products and users. They were then asked to score how useful the two lists were in achieving their goal, using a five-point Likert-like scale, where the first position corresponded to “not useful at all” and the fifth one to “very useful”. Moreover, they were asked decide which list they found to be the most useful and to provide reasons for their choice through a free text comment.

Results. List B, presenting DSRs, was reported to be more useful than List A (average: 2.69; std. dev: 1.30), with an average evaluation of 3.45 (std. dev: 1.19). A one-way ANOVA test, with a $p\!-\!value = 0.01$, showed that these differences in average user evaluations are statistically significant ($F(1; 78) = 10.03$). A total of 30/40 (75%) of users declared List B was the most useful one, 4 users found it clear, 14 users considered it detailed, more informative and complete and 12 people liked the group evaluation and suggestion.

In conclusion, the answer to [RQ1] is: yes, recommendations generated with our DSR algorithm are more useful than traditional, single-sided recommendations as they are clearer, more detailed, informative and complete, thanks to the presence of the group and of its evaluation.

7 Although the number of users involved in the two empirical evaluations presented in this paper is relatively small, it is in line with other studies carried out in the same domain, as in Sanchez et al., 2013

Offline Experiment

In order to evaluate the performance of our DSR algorithm in recommending items, we compared it with a user-based collaborative filtering algorithm, in terms of precision and recall, in an offline evaluation.

Dataset

We adopted the Last FM[8] dataset[9] released in the framework of the 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems (HetRec 2011) (Cantador et al., 2011). To the best of our knowledge, among the datasets available at time of writing on the Web, this is the only one that provides the information we needed to apply the DSR algorithm: actions on items, social relationships among users and ratings on items.

This dataset contains information on 1,892 users listening and tagging 17,632 artists. In particular there are 92,834 “user-listened artist” relations and 186,479 tags have been assigned to the artists by users. Moreover, 12,717 bi-directional relations express friendship links amongst users. The “user-listened artist” relations specify a listening count that we exploited to define which artists can be considered as “selected” by a certain user, i.e. artists that have been listened to by the user a number of times that is greater than the average number of times that user has listened to all the artists.

Recommendation generation

DSR. The selected dataset provided us with data we directly exploited to apply the DSR algorithm: (i) the friendship relations identify target users’ social networks and determine user-user relevance and (ii) the tags that users assigned to artists represent user actions on items. However, the calculation of the similarity between two users necessitated obtaining information that was not explicitly expressed by the available data, in particular we had to build a user model describing users’ preferences for music genres. Having observed that the number of tags users labelled artists with are genres descriptions, we identified 63 tags defining genres and assigned a certain genre to an artist if he/she was labelled with the corresponding tag. The user model was then built by assigning a genre interest value to a user by taking into account how many times the user had listened to an artist belonging to that genre.

User-based collaborative filtering. Recommender-lab (Hahsler, 2011), a widely used R[10] extension package (Chen et al., 2013; Buhl et al., 2016; Beel et al., 2016) that fitted our needs, was used to generate recommendations with a user-based collaborative filtering technique. Indeed, it was created with a completely different goal to the existing software packages, since it provides a general research infrastructure for recommender systems. It offers consistent and efficient data handling, built-in collaborative filtering algorithms, easy incorporation of algorithms, ex- permanent set up and evaluation of the results (Bali and Sarkar, 2016; Gorakala and Usuelli, 2016). We adopted a Recommender-Lab evaluation scheme with a split method that assigns 90% of users to a training set and 10% of users to a test set. As for the parameter settings, similarity between users was measured using the cosine similarity metric, whilst user rating bias was reduced using centered normalization (i.e., ratings by user u are represented as

---

[8] http://www.lastfm.com
[9] http://grouplens.org/datasets/hetrec-2011/
[10] R is a programming language and software environment for statistical computing and graphics (see https://www.r-project.org/).

variations compared to the average of all available ratings expressed by user u) (Table 7). Lastly, all missing ratings (zeros) were treated as negative examples. For all the aforementioned parameters, we followed the standard approach described by Hahsler (2011).

Table 6
Precision and recall for DSR and the user-based collaborative filtering algorithms in the offline evaluation.

	Precision		Recall	
	DSR	UBCF	DSR	UBCF
TOP 1	0.13	0.0	0.01	0.0
TOP 3	0.09	0.0	0.02	0.0
TOP 5	0.09	0.01	0.03	0.0
TOP 10	0.07	0.0	0.05	0.0
TOP 15	0.06	0.01	0.06	0.0
TOP 20	0.06	0.0	0.14	0.0

Results

We compared precision and recall obtained in recommending the TOP-n items (with $N \in 1, 3, 5, 10, 15, 20$) with the DSR and the user-based collaborative filtering algorithm. Table 6 shows the results of the experiment: both precision and recall are higher for the DSR algorithm, under all conditions. This result provides an affirmative answer to [RQ2]: i.e, yes, the item-recommendation part of the DSR algorithm performs better than a user-based collaborative filtering one.

Online Experiment

With our online experiment, we aim at comparing the performances of our "social" DSR algorithm to those of a traditional, content-based algorithm which is known to have a satisfying performance in a scenario where consolidated user models are available.

It was decided to focus on the recommendation of social and cultural events for this experiment on the basis for various factors:

- people usually participate in such events with other people, so this is an appropriate domain to test DSRs and, particularly, our social approach which leverages the preferences of the target user's contacts;
- in the past, we contributed to the development of iCITY DSA (Carmagnola et al., 2008), a social recommender system which suggests events taking place in the municipality of Turin:
  - iCITY uses a content-based recommendation algorithm which proved to provide good performances in a consolidated context of use;
  - we were able to access information on user preferences for cultural events and friendship relationships collected during one of the past evaluations of iCITY, when users interacted with the system for two weeks.

Noteworthy is the fact that an event is a peculiar item to recommend: according to Cornelis et al. (2007) it is a "one-and-only item", since it usually happens at a specific time and place, has a limited availability and is often experienced in groups. Moreover, it needs to be recommended before anyone has been able to attend it (excluding recurrent events) and feedback is available only for past events, which are not those being recommended.

Participants

We recruited participants by sending an email to all the people who had participated in the past, two-week evaluation of iCITY and received a positive answers from about 40% of them, thus obtaining a group of 25 participants, 8 males and 17 females, age range 25-50 years, all residing in the Turin area.

Content-based algorithm

The content-based algorithm, which was designed for the iCITY recommender system, assigns a score to candidate events based on the following features:

- user interests. They are represented in a user model in the form of a probability distribution as to domain-related categories, e.g., events, sport, rock music, classical music or inaugurations. Thus, each value ranges from 0 to 1 and the sum of all values in the user model is equal to 1. The specific method used to initialize and update user models is beyond the scope of this paper, but any readers wishing to have more information may find it in the paper by Carmagnola et al. (2008). Each event is associated to a domain category and the users' interest for a certain event is inferred from their interest in the corresponding category.
- Position. Position is intended as the spatial proximity between the target user and a candidate event. It is calculated as $1 - d$, where d is the Euclidean distance between the user's current position and the event location, normalized compared to a value representing the maximum distance users are willing to travel to participate in a social or cultural event[11].
- Recentness. Similarly to position, this feature refers to the temporal proximity of an event. It is calculated as $1 - (td)$, where td is the temporal distance, expressed in minutes, between the current time and the time an event finishes, normalized compared to a value indicating the maximum temporal distance for forthcoming events (this was set at 14 days).

- *Rating*. It is the average of the user ratings.

More specifically, system scores for each event are determined according to the following formula:

$$eventScore = interests * w_1 + proximity * w_2 + recentness * w_3 + rating * w_4$$

where w_n are weights assigned to the four features. In our case, weights were set at 0.5, 0.3, 0.15 and 0.05, respectively, a combination which favours user interests and has been successfully tested in past evaluations of this algorithm.

Data gathering and recommendation generation

Most of the data on users and events we needed to generate recommendations were obtained from the past evaluation done in iCITY.

[11] This value was determined based on (Cena et al., In press) and set to 47Km.

Event-related data. Each event is characterized by the following information: title, detailed description, start and end date, location and corresponding domain category. Moreover, it was possible to access a record of the actions users had performed for each event: rating, bookmarking, tagging and commenting.

Ratings were used by the content-based algorithm, together with information on user interests for the corresponding event category, event location and event end date (see Section 5.3.2). All user actions were taken into account by the DSR algorithm to assess event relevance (see Section 4.1).

User-related data. All users have a user model, storing their preferences as to domain categories. Moreover, it was possible to access information on those actions users performed in the past which have an effect on other users, like befriending and sending a message. However, differently from what had happened during other, longer evaluations, however, no users had sent messages to other participants in the two- week evaluation of iCITY. That, and since we suspected friendship relationships from our system might not reflect actual user habits in taking part to events, prompted us to email all participants, asking them to choose the people they normally participated (or might like to participate) in events with from a list of all the participants in our current evaluation. User choices were treated as if they were "sending message" actions, i.e., they were assigned the same parameters for the maximum value, threshold and weight (see Table 5).

Information as to user interests is exploited by the content-based algorithm in the recommendation generation process (see Section 5.3.2) and by the DSR algorithm to assess similarity between two users (see Section 4.2.1). Information on friendship relationships is used by the DSR algorithm to assess user relevance (see Section 4.1).

Recommendation generation. Two different lists of recommendations were generated for each participant, according to the DSR and the content-based algorithms, respectively. Each recommendation list had 10 items , i.e. the 5 items with the highest predicted system score, which were considered "recommended items" and 5 items chosen at random among those with the lowest system scores ("non-recommended items"). DSRs were presented as explained in Section 4.4 and employed a value of 0.5 for both weights, as a neutral combination. As aforementioned, such weights will be tuned in the second part of this experiment so as to answer question RQ4. The scores for collective item relevance and group score were displayed by means of a 5-star and a 5-heart widget, respectively, whilst social confidence was simply presented by means of a number (see Figure 3).

All the available information (e.g., title, description, location, etc.) was visualized for each event in content-based recommendations. A representation of the average user rating and the system prediction for the current target user were provided by two 5-star widgets (see Figure 2).

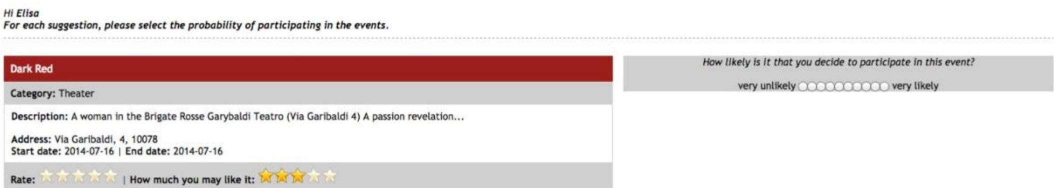

Figure 2: Evaluation: List A (content-based recommendations).

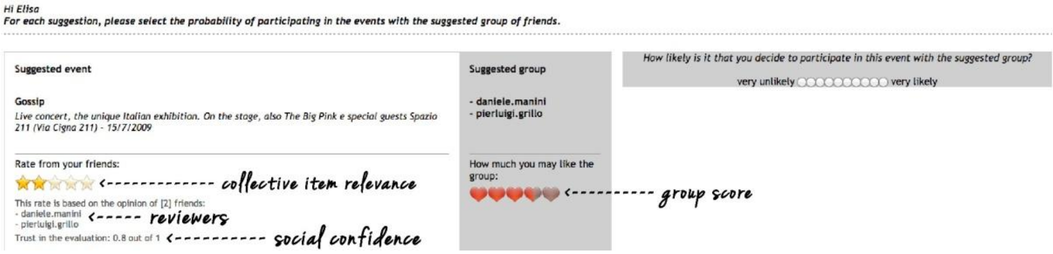

Figure 3: Evaluation: List B (DSR recommendations).

Empirical evaluation

Experimental Design. We chose a within-subjects, single factor design, where the independent factor is the *recommendation algorithm* and has two possible levels: content-based algorithm from iCITY or a DSR algorithm. We counterbalanced to control for order effects by randomizing the order of presentation of recommendation lists generated with the two algorithms, respectively.

There is also a single dependent variable, the *user score*, i.e., a score, on a 10-point rating scale, users may adopt to express how likely they are to participate in the recommended event (with the recommended group, if present).

Material. Instructions, as well as the two recommendation lists, were presented on separate dedicated web pages.

Procedure. Participants were invited to take part in the experimental evaluation via email. They accessed a web page by following a dedicated link where an introduction to our evaluation was available. Going ahead, they were provided with two different lists of recommendations, one at a time, generated by the two algorithms under evaluation. Participants were asked to assess each recommendation (either an event, or an event/group package), indicating how likely it was that they would actually participate in the recommended event, using a 10-point scale ranging from 1 (very unlikely) to 10 (very likely). A small script was exploited to check whether the participants had actually evaluated all recommendations before they could leave a page and proceed. Once they had completed all the evaluations, the participants were thanked for their help and could simply quit the survey.

Results and discussion

Let us present the results of our evaluation and discuss how they provide us with answers to questions [RQ3] and [RQ4].

To answer [RQ3], we compared the DSR algorithm with the iCITY algorithm, that had been given a positive performance rating in a real-world evaluation that lasted a few months (Gena et al., 2013). We compared them adopting precision, recall, accuracy and F_1, that are well-known metrics for recommender systems (Shani and Gunawardana, 2011). They consider a recommender as a classification system and evaluate its ability to predict the class of a candidate item correctly, whether "selected by the target user" or "not selected by the target user". Since we asked users to rate the proposed recommendations on a 1-10 scale, we considered an item with a rating of at least 6, which corresponds to sufficiency in the commonsense view, as a selected item.

Table 7
Summary of parameter setting in Recommenderlab.

Method	10-fold cross validation
Similarity metric	Cosine
Rating normalization	Yes (centering)
Missing values	Treated as zeroes

Table 8 reports the results of the DSR and iCITY algorithms in terms of precision, recall, accuracy and F_1. All the scores are quite low, which can be explained by the fact that user models cannot be considered “consolidated”, since they are based on only two weeks of interaction with the iCITY system. Conversely, DSR allows us to obtain improvements on all the four measurements: 8.9% for precision, 25.6% for recall, 15.4% for accuracy and 15.2% for F1. The particularly good results obtained for recall shows that using information drawn from the target users’ social network allows for an improvement in the capability of a recommender system to identify candidate recommendations that might be appreciated by the target user, even if no precise representations of the target users’ preferences are available.

Therefore, referring back to question [RQ3], we can conclude that the “social” DSR algorithm proposed in this paper performs better than a traditional, content-based one. Moreover, it provides more complete and informative suggestions than do single-sided recommendations, as it couples the recommended item (in this case, an event) with a group of people it can be attended with: as confirmed by our preliminary evaluation, users do appreciate this two-fold information, a result which opens up the possibility of assessment and maybe the adoption of double-sided approaches in different contexts.

With question [RQ4], we aimed at investigating how much the two factors *collective item relevance* and *group score* should be weighted in calculating the total score for the event recommendation domain. Moreover, we wanted to understand which function of *collective item relevance* and *group score* better models our system. To date, we have simply adopted the average,(we have used average everywhere else) that is, we adopted a linear combination with each factor weighted 0.5, but is a linear combination the best solution?

In order to answer this question, we tried different combinations with the aim of lowering the RMSE (Root Mean Square Error) between “system scores”, i.e., the predictions given by our algorithm and “user scores”, i.e., the values explicitly provided by users. RMSE[12] is a well-accepted statistical accuracy metric for recommender systems (Adomavicius and Tuzhilin, 2005; Victor et al., 2011; Herlocker et al., 2004; Arazy et al., 2010). The initial RMSE value for our DSR algorithm (namely, considering a linear combination of *collective item relevance* and *group score*, with a 0.5 weight for both factors) was 0.2820, considering only recommended items. Note that the RMSE obtained by the content-based algorithm was slightly higher, i.e. 0.2861.

We exploited the Weka Environment for Knowledge Analysis[13] (Witten et al., 2011) to run several tests with different classifier functions. The input attributes were *collective item relevance* and *group score*, the predicted value was the “user score” and the mode was 10-fold cross-validation, for all the tests. Table 9 summarizes the results obtained

[12] Note that a negative variation in the RMSE is regarded as an improvement since a lower RMSE denotes a higher degree of accuracy. As evidenced by Arazy et al. (2010) even small RMSE improvements are considered valuable in the context of recommender systems: for example the Netflix prize competition offered a one million dollar reward for an RMSE reduction of 10 percent.
[13] http://www.cs.waikato.ac.nz/ml/weka/

showing the RMSE for each function and the improvement compared to the iCITY content-based algorithm, on the one hand and to the original linear combination we used for the DSR algorithm on the other, considering only recommended items/packages in both cases.

The best result was obtained by Simple Linear Regression with the following model, which completely excludes the *collective item relevance* :

totalScore = 0.67 · gsc + 0.02

Assuming that the scores we determined for the item and group components do reflect users' opinions, our analysis does not only prove useful to fine-tune our algorithm, but also sheds light on the relative importance users assign to the two components of DSRs when they are evaluating a suggestion. Surprisingly, it appears that the recommended event is not taken into account at all by users, who seem to base their preference only on the group. A possible explanation for this result is that events are peculiar items, where other people's company bears a predominant weight over other factors.

On the basis of this idea, we performed a new analysis to compare the group-score-based DSR (GSB-DSR) algorithm compared to the iCity content-based algorithm and the basic DSR algorithm and obtained a better performance, as shown in Table 10. More specifically, comparing GSB-DSR to the former, there was an improvement in all the measurements: 20.6% for precision, 23.1% for recall, 25.4% for accuracy and 21.6% for F1. On the other hand, the comparison with the basic DSR algorithm shows an improvement in precision (10.7%), accuracy (8.7%) and F1 (5.6%) at the expense of a worse recall (−2.0%), meaning that the GSB-DSR algorithm discards more events that the user would select than basic DSR algorithm does. A possible interpretation of this phenomenon is that some (very few) users choose events, not only by taking into consideration groups, but also collective item relevance, something that the GSB-DSR algorithm ignores and the basic DSR algorithm takes into account.

We surmise that recommenders in different application domains might need different "default" weight values if no detailed information on user preferences and/or interests are available and that personalized weights might need to be defined for each user once a recommender is functioning at full speed. We can now answer question [RQ4] as follows: in the domain of events, users ground their choices only on the *group score* and seem to neglect *collective item relevance*.

Table 8
Precision, recall, accuracy and F1 for DSR and iCITY content based recommender systems.

	CB	DSR	Δ (%)
Precision	0.35	0.38	8.9%
Recall	0.5	0.63	25.6%
Accuracy	0.5	0.58	15.4%
F1	0.41	0.47	15.2%

Table 9
Classifier functions test results.

Classifier	RMSE	Δ (%) w.r.t. CB alg.	Δ (%) w.r.t. "social" DSR alg.
GaussianProcesses	0.2706	−6.45%	−4.05%
IsotonicRegression	0.2844	−1.67%	0.84%
LeastMedSq	0.2845	−1.64%	0.88%
LinearRegression	0.2739	−5.30%	−2.88%
MultilayerPerceptron	0.3514	21.49%	24.60%
Pace Regression	0.2716	−6.10%	−3.70%
RBF network	0.2688	−7.07%	−4.69%
SimpleLinearRegression	0.2676	−7.48%	−5.11%
SM oreg	0.2782	−3.82%	−1.36%

Table 10
Precision, recall, accuracy and F1 for group-score-based DSR compared to DSR and iCITY content based recommender systems.

	CB	DSR	GSB-DSR	Δ (%) w.r.t. CB	Δ (%) w.r.t. DSR
Precision	0.35	0.38	0.42	20.6%	10.7%
Recall	0.50	0.63	0.62	23.1%	−2.0%
Accuracy	0.50	0.58	0.63	25.4%	8.7%
F1	0.41	0.47	0.50	21.6%	5.6%

Table 11
Precision, recall, accuracy and F1 for the full DSR algorithm and its item-recommendation part.

	DSR - item-recommendation part only	DSR	Δ (%)
Precision	0.46	0.44	−4.5%
Recall	0.59	0.66	10,6%
Accuracy	0.57	0.61	6,6%
F1	0.52	0.53	1,9%

Follow-up Online Experiment

A follow-up online experiment was devised to investigate whether the performances of our DSR algorithm may largely depend on the fact that it exploits information extracted from the target user's social network. Indeed, Guy et al. (2009) reported that providing information on the users whose opinion recommendations are based on improved the performances of their algorithm and Bourke et al. (2011) stated that "revealing the neigbourhood to the participant does have an impact".

To this purpose, we compared the recommendations generated with the full DSR algorithm that we assessed in our online evaluation with a set of recommendations generated only with its item-recommendation part. Both kinds of recommendations contain the list of reviewers who expressed an opinion about the suggested item and the only aspect which differentiates full DSRs is the presence of the suggested group, which is the hallmark of this kind of recommendation.

All the people who participated in our online experiment (see Section 5.3) were invited to participate in the follow-up and a positive answer was received from 19/25 of them.

A list containing suggestions generated with the item-recommendation part of the DSR algorithm was prepared for each participant, according to the same criteria described in Section 5.3.3. The follow-up evaluation was also carried out according to the same evaluation format, described in the same section.

Results

Table 11 reports the results of the two algorithms in terms of precision, recall, accuracy and F_1. For reasons of comparability, we only considered data from the participants that took part in both studies, as far as the full DSR algorithm is concerned. The full DSR algorithm performs better than its item-recommendation part for recall, accuracy and F1. Therefore, we can state that the performance improvements shown by the DSR algorithm in the online experiment (see Section 5.3) are not simply produced by the inclusion of information drawn from the target user's social network: instead, users place particular value on recommendations that include a group of people who the suggested item can be consumed with.

However, noteworthy is the fact that the full DSR algorithm has a little lower precision than does its item-recommendation part. This result might be explained by the fact that good DSRs are intrinsically more complicated to generate, making them more prone to error, as they require good predictions on three aspects (rather than only one, i.e. the suggested item, the suggested group and the item/group match. Given that the item-recommendation part already has better precision than does the full DSR algorithm, we surmise that we should concentrate our efforts on improving the group recommendation part, which is quite simplistic at the moment.

6. Conclusions and Future Research

In this paper we presented an algorithm based on the Social Comparison Theory, which generates double-sided recommendations starting from the preferences extrapolated from the social network of the target user. Our empirical evaluations assessed this algorithm from various points of view (usefulness, performance, parameter tuning) and in different domains, trying to answer five research questions (RQ1-5, see section 5).

Firstly, our data showed that our double-sided recommendations were considered more useful than traditional (single-sided) recommendations that provide equivalent information by users considered who wish to adopt collaborative buying practices (RQ1). This result suggests that double-sided recommendations can better support decision making in domains where target activities are usually performed with other people, but where a pre-defined group is not always available.

Secondly, we observed that the item-recommendation part of our algorithm performs better than a standard, user-based collaborative-filtering algorithm in terms of precision and recall (RQ2). This means that our algorithm has a strong foundation and that further research in the double-sided recommendation field could be concentrated on other aspects, such as group recommendation.

Indeed, a weak point in our approach is the way groups are recommended, which is currently quite simplistic. Our future research will be aimed at improving on this point by taking into consideration also such parameters as group members' overall similarity in preferences and interests (to build homogeneous groups), or the relationships tying group members themselves, aiming at identifying more cohesive groups. As a first step, we are planning to further investigate into the idea suggested in (Vernero, 2011), i.e. to use structural subcomponents in the target user's social network as a basis for group generation. Furthermore, we are considering integrating information on users' duties and constraints, with the aim of generating more realistic and useful group suggestions. For example, we could add information from social calendar applications to understand when users are actually available (e.g., in case the system suggests participating in a social event), or take into account users' geographical location in cases where it is appropriate that group members all live near to one other (e.g., for collective buying groups).

Thirdly, we demonstrated that, in the event recommendation domain, our "social" DSR algorithm performs better than a content-based one which is already known to have a good performance, considering precision, recall, accuracy and $F1$ (RQ3). This result implies that our algorithm does not produce many "spam-like" recommendations, even if the preferences of their friends are favoured over those of the target users. Coupling this result with our finding on double-sided recommendation usefulness, we can conclude that this type of suggestion is better suited to user needs in contexts where the "social" part is important. However, we are considering mediating the preferences expressed by the target users' social network with their personal ones in future research. A similar approach was successfully carried out by Carmagnola et al. (2014), where the authors showed that mixing information on personal and "social" preferences enhanced the precision of recommendations on the standard SoNARS algorithm (which uses only information drawn from the target user's social network).

Fourthly, our analyses showed that users' scores for recommended item- group packages can be better predicted by considering only the system scores for the recommended groups, at least in the domain of social and cultural events (RQ4). An understanding of how people assess the recommended items and groups in other domains is an open point which we are planning to deal with in the future. Therefore, we may sum up by saying that the different application domains are likely to call for different weights for items and groups. These will necessarily be uniform for all users in the cold-start phase, but might well be personalized according to user's preferences and decision-making styles once detailed information on these aspects becomes available.

Lastly, we verified that the good performance of the DSR algorithm does not simply depend on the inclusion of information drawn from the target user's social network (RQ5). This is an important result, since it confirms that the value of DSRs depends on their particular format, i.e., the fact that they include a suggested group, which is the hallmark of this kind of recommendation.

With the current spread of social networks and the possibility of using social login to access specific services, we can imagine that the application of DSRs will become more and more widespread. We expect that the success of this kind of recommendation will largely depend on the capability to generate realistic group suggestions, which depends partly on the refinements that we can make to the group generation algorithm and partly on the availability of high-quality data. To this end, we envisage the need to carry out a large scale, real-world evaluation of our algorithm, where users can access some valuable service (e.g., a website providing information on social events) through their social login, browse among its pages at leisure and receive DSRs based on the behaviour of their social connections.

References

Adomavicius, G., Tuzhilin, A., Jun. 2005. Toward the next generation of recommender systems: a survey of the state-of-the-art and possible ex- tensions. IEEE Transactions on Knowledge and Data Engineering 17 (6), 734–749.

Amer-Yahia, S., Roy, S. B., Chawla, A., Das, G., Yu, C., 2009. Group recommendation: Semantics and efficiency. PVLDB 2 (1), 754–765.

Arazy, O., Kumar, N., Shapira, B., 2010. A theory-driven design framework for social recommender systems. Journal of the Association for Information Systems, AIS 11 (9).

Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torasso, P., 2003. Intrigue: Personalized recommendation of tourist attractions for desktop and hand held devices. Applied Artificial Intelligence 17 (8-9), 687–714.

Baatarjav, E.-A., Phithakkitnukoon, S., Dantu, R., 2008. Group recommen- dation system for facebook. In: OTM ’08: Proceedings of the OTM Con- federated International Workshops and Posters on On the Move to Mean- ingful Internet Systems. Springer-Verlag, Berlin, Heidelberg, pp. 211–219.

Bali, R., Sarkar, D., 2016. R Machine Learning By Example. Community experience distilled. Packt Publishing.
URL https://books.google.it/books?id=fLPjCwAAQBAJ

Baltrunas, L., Makcinskas, T., Ricci, F., 2010. Group recommendations with rank aggregation and collaborative filtering. In: Proceedings of the 2010 ACM Conference on Recommender Systems, RecSys 2010, Barcelona, Spain, September 26-30, 2010. pp. 119–126.
URL http://doi.acm.org/10.1145/1864708.1864733

Basu, C., Hirsh, H., Cohen, W., 1998. Recommendation as classification: Using social and content-based information in recommendation. In: Proc. of the 15th National Conference on Artificial Intelligence. AAAI Press, pp. 714–720.

Beel, J., Breitinger, C., Langer, S., Lommatzsch, A., Gipp, B., 2016. Towards reproducibility in recommender-systems research. User Modeling and User- Adapted Interaction 26 (1), 69–101.
URL http://dx.doi.org/10.1007/s11257-016-9174-x

Berkovsky, S., Freyne, J., 2010. Group-based recipe recommendations: anal- ysis of data aggregation strategies. In: RecSys ’10: Proceedings of the fourth ACM conference on Recommender systems. ACM, New York, NY, USA, pp. 111–118.

Berkovsky, S., Freyne, J., Smith, G., 2012. Personalized network updates: In- creasing social interactions and contributions in social networks. In: Mas- thoff, J., Mobasher, B., Desmarais, M. C., Nkambou, R. (Eds.), User Mod- eling, Adaptation, and Personalization - 20th International Conference, UMAP 2012, Montreal, Canada, July 16-20, 2012. Proceedings. Vol. 7379 of Lecture Notes in Computer Science. Springer, pp. 1–13. http://dx.doi.org/10.1007/978-3-642-31454-4_1.

Boratto, L., Carta, S., 2011. State-of-the-art in group recommendation and new approaches for automatic identification of groups. In: Soro, A., Vargiu, E., Armano, G., Paddeu, G. (Eds.), Information Retrieval and Mining in Distributed Environments. Springer, pp. 1–20.

Boratto, L., Carta, S., Chessa, A., Agelli, M., Clemente, M. L., 2009. Group recommendation with automatic identification of users communities. In: Web Intelligence/IAT Workshops. IEEE Computer Society, pp. 547–550.

Bourke, S., McCarthy, K., Smyth, B., 2011. Power to the people: Exploring neighbourhood formations in social recommender system. In: Proceedings of the Fifth ACM Conference on Recommender Systems. RecSys ’11. ACM, New York, NY, USA, pp. 337–340.
URL http://doi.acm.org/10.1145/2043932.2043997

Buhl, M., Famulare, J., Glazier, C., Harris, J., McDowell, A., Waldrip, G., Barnes, L. E., Gerber, M. S., April 2016. Optimizing multi-channel health information delivery for behavioral change. In: 2016 IEEE Systems and Information Engineering Design Symposium (SIEDS). pp. 130–135.

Cantador, I., Brusilovsky, P., Kuflik, T., 2011. 2nd workshop on informa- tion heterogeneity and fusion in recommender systems (hetrec 2011). In: Proceedings of the 5th ACM conference on Recommender systems. RecSys 2011. ACM, New York, NY, USA.

Carmagnola, F., Cena, F., Console, L., Cortassa, O., Gena, C., Goy, A., Torre, I., Toso, A., Vernero, F., Jul. 2008. Tag-based user modeling for social multi-device adaptive guides. User Modeling and User-Adapted In- teraction 18 (5), 497–538.

Carmagnola, F., Vernero, F., Grillo, P., 2009. Sonars: A social networks- based algorithm for social recommender systems. In: Houben, G.-J., Mc- Calla, G., Pianesi, F., Zancanaro, M. (Eds.), User Modeling, Adaptation, and Personalization. Vol. 5535 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 223–234.

Carmagnola, F., Vernero, F., Grillo, P., 2014. Advanced social recommendations with sonars++. Interacting with Computers 26 (1), 75–88.
URL http://dx.doi.org/10.1093/iwc/iwt028

Cena, F., Likavec, S., Lombardi, I., Picardi, C., In press. Should I Stay or Should I Go? Improving Event Recommendation in the Social Web. Interacting with Computers.

Chao, D. L., Balthrop, J., Forrest, S., 2005. Adaptive radio: achieving con- sensus using negative preferences. In: GROUP ’05: Proceedings of the 2005 international ACM SIGGROUP conference on Supporting group work. ACM, New York, NY, USA, pp. 120–123.

Chen, J. H., Chao, K. M., Shah, N., 2013. Hybrid recommendation system for tourism. In: Proceedings of the 2013 IEEE 10th International Con- ference on e-Business Engineering. ICEBE ’13. IEEE Computer Society, Washington, DC, USA, pp. 156–161.
URL http://dx.doi.org/10.1109/ICEBE.2013.24

Christensen, I. A., Schiaffino, S. N., 2014. Social influence in group recom- mender systems. Online Information Review 38 (4), 524–542.
URL http://dx.doi.org/10.1108/OIR-08-2013-0187

Cornelis, C., Lu, J., Guo, X., Zhang, G., 2007. One-and-only item recommen- dation with fuzzy logic techniques. Information Sciences 177 (22), 4906– 4921.

Festinger, L., 1954. A theory of social comparison process. Human Relations 7, 117–140.

Gartrell, M., Xing, X., Lv, Q., Beach, A., Han, R., Mishra, S., Seada, K., 2010.

Enhancing group recommendation by incorporating social relation- ship interactions. Proceedings of the 16th ACM international conference on Supporting group work - GROUP ’10, 97–106.

Gena, C., Cena, F., Vernero, F., Grillo, P., 2013. The evaluation of a social adaptive website for cultural events. User Model. User-Adapt. Interact. 23 (2-3), 89–137.

Gorakala, S. K., Usuelli, M., 2016. Building a Recommendation System with R. Packt Publishing.

Guy, I., Levin, R., Daniel, T., Bolshinsky, E., 2015. Islands in the stream: A study of item recommendation within an enterprise social stream. In: Baeza-Yates, R. A., Lalmas, M., Moffat, A., Ribeiro-Neto, B. A. (Eds.), Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, August 9-13, 2015. ACM, pp. 665–674.
URL http://doi.acm.org/10.1145/2766462.2767746

Guy, I., Zwerdling, N., Carmel, D., Ronen, I., Uziel, E., Yogev, S., Ofek- Koifman, S., 2009. Personalized recommendation of social software items based on social relations. In: Bergman, L. D., Tuzhilin, A., Burke, R. D., Felfernig, A., Schmidt-Thieme, L. (Eds.), Proceedings of the 2009 ACM Conference on Recommender Systems, RecSys 2009, New York, NY, USA, October 23-25, 2009. ACM, pp. 53–60.
URL http://doi.acm.org/10.1145/1639714.1639725

Hahsler, M., 2011. recommenderlab: A framework for developing and testing recommendation algorithms. Tech. rep.
URL http://scholar.google.de/scholar.bib?q=info:ULHTdsUD_
n4J:scholar.google.com/&output=citation&hl=de&as_sdt=0,5&ct= citation&cd=0

Herlocker, J. L., Konstan, J. A., Terveen, L. G., Riedl, J., 2004. Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. 22 (1), 5–53.

Hsu, W. H., King, A. L., Paradesi, M. S. R., Pydimarri, T., Weninger, T., 2006. Collaborative and structural recommendation of friends using weblog-based social network analysis. In: in AAAI Spring Symposia 2006 on Computational Approaches to Analysing Weblogs.

Jameson, A., Berendt, B., Gabrielli, S., Gena, C., Cena, F., Vernero, F., Reinecke, K., 2014. Choice architecture for human-computer interaction. Foundations and Trends in Human-Computer Interaction 1-2 (7), 1–235.

Jameson, A., Smyth, B., 2007. Recommendation to groups. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (Eds.), The Adaptive Web. Vol. 4321 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 596–627.

Kim, J. K., Kim, H. K., Oh, H. Y., Ryu, Y. U., Jun. 2010. A group rec- ommendation system for online communities. Int. J. Inf. Manag. 30 (3), 212–219.

Kobsa, A., Koenemann, J., Pohl, W., Dec. 2001. Personalised hypermedia presentation techniques for improving online customer relationships. The Knowledge Engineering Review 16 (02), 111.

Mart´ınez-Cruz, C., Porcel, C., Bernab´e-Moreno, J., Herrera-Viedma, E., 2015. A model to represent users trust in recommender systems using on- tologies and fuzzy

linguistic modeling. Inf. Sci. 311, 102–118.
URL http://dx.doi.org/10.1016/j.ins.2015.03.013

Ntoutsi, I., Stefanidis, K., Nørv˚ag, K., Kriegel, H., 2012. grecs: A group recommendation system based on user clustering. In: Lee, S., Peng, Z., Zhou, X., Moon, Y., Unland, R., Yoo, J. (Eds.), Database Systems for Advanced Applications - 17th International Conference, DASFAA 2012, Busan, South Korea, April 15-19, 2012, Proceedings, Part II. Vol. 7239 of Lecture Notes in Computer Science. Springer, pp. 299–303.
URL http://dx.doi.org/10.1007/978-3-642-29035-0_25

Popescu, G., Pu, P., 2012. What's the best music you have?: designing music recommendation for group enjoyment in groupfun. In: Konstan, J. A., Chi,
E. H., H¨oo¨k, K. (Eds.), CHI Conference on Human Factors in Computing Systems, CHI '12, Extended Abstracts Volume, Austin, TX, USA, May 5-10, 2012. ACM, pp. 1673–1678.
URL http://doi.acm.org/10.1145/2212776.2223691

Recio-Garcia, J. A., Jimenez-Diaz, G., Sanchez-Ruiz, A. A., Diaz-Agudo, B., 2009. Personality aware recommendations to groups. In: RecSys '09: Proceedings of the third ACM conference on Recommender systems. ACM, New York, NY, USA, pp. 325–328.

Sa´nchez, L. Q., Recio-Garc´ıa, J. A., D´ıaz-Agudo, B., Jim´enez-D´ıaz, G., 2013. Social factors in group recommender systems. ACM TIST 4 (1), 8. URL http://doi.acm.org/10.1145/2414425.2414433

Schafer, J. B., Frankowski, D., Herlocker, J. L., Sen, S., 2007. Collaborative filtering recommender systems. In: The Adaptive Web. pp. 291–324.

Shani, G., Gunawardana, A., 2011. Evaluating recommendation systems. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P. B. (Eds.), Recommender Systems Handbook. Springer US, pp. 257–297.

Sinha, R., Swearingen, K., 2001. Comparing recommendations made by on- line systems and friends. In: In Proceedings of the DELOS-NSF Workshop on Personalization and Recommender Systems in Digital Libraries.

Stefanidis, K., Pitoura, E., 2013. Finding the Right Set of Users : Generalized Constraints for Group Recommendations. In: CoRR 2013.

Stefanidis, K., Shabib, N., Nørv˚ag, K., Krogstie, J., 2012. Contextual rec- ommendations for groups. In: Castano, S., Vassiliadis, P., Lakshmanan,
L. V. S., Lee, M. (Eds.), Advances in Conceptual Modeling - ER 2012 Workshops CMS, ECDM-NoCoDA, MoDIC, MORE-BI, RIGiM, SeCoGIS, WISM, Florence, Italy, October 15-18, 2012. Proceedings. Vol. 7518 of Lec- ture Notes in Computer Science. Springer, pp. 89–97.
URL http://dx.doi.org/10.1007/978-3-642-33999-8_11

Suls, J., Martin, R., Wheeler, W., March 2002. Responding to the social world: Explicit and implicit processes in social judgments and decisions,. In: Forgas, J. P., Williams, K. D., Von Hippel, W. (Eds.), Social judg- ments: implicit and explicit processes. Cambridge University Press, pp. 1–22.

Vernero, F., 2011. Double-sided recommendations: a novel framework for recommender systems. In: Pirrone, R., Sorbello, F. (Eds.), AI* IA 2011: Artificial Intelligence Around Man and Beyond. Vol. 6934. pp. 262–273.

Victor, P., Cock, M. D., Cornelis, C., 2011. Trust and recommendations. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P. B. (Eds.), Recommender Systems Handbook. Springer, pp. 645–675.

Witten, I. H., Frank, E., Hall, M. A., 2011. Data Mining: Practical Machine Learning Tools and Techniques, 3rd Edition. Morgan Kaufmann.