

On Achieving Diversity in Recommender Systems

Marialena Kyriakidi
National & Kapodistrian
University of Athens
Athens, Greece
marilou@di.uoa.gr

Kostas Stefanidis
University of Tampere
Tampere, Finland
kostas.stefanidis@uta.fi

Yannis Ioannidis
National & Kapodistrian
University of Athens
and "Athena" Research Center
Athens, Greece
yannis@di.uoa.gr

ABSTRACT

Throughout our digital lives, we are getting recommendations for about almost everything we do, buy or consume. In that way, the field of recommender systems has been evolving vastly to match the increasing user needs accordingly. News, products, ideas and people are only a few of the things that we can be recommended with daily. However, even with the many years of research, several areas still remain unexplored. The focus of this paper revolves around such an area, namely on how to achieve diversity in single-user and group recommendations. Specifically, we decouple diversity from strictly revolving around items, and consider it as an orthogonal dimension that can be incorporated independently at different times in the recommender's workflow. We consider various definitions of diversity, taking into account either data items or users characteristics, and study how to cope with them, depending on whether we opt at diversity-aware single-user or group recommendations.

ACM Reference format:

Marialena Kyriakidi, Kostas Stefanidis, and Yannis Ioannidis. 2017. On Achieving Diversity in Recommender Systems. In *Proceedings of ExploreDB'17, Chicago, IL, USA, May 14-19, 2017*, 6 pages.
<https://doi.org/http://dx.doi.org/10.1145/3077331.3077341>

1 INTRODUCTION

Recommender systems have become indispensable for several Web sites, such as Amazon, Netflix, Yelp and Google News, helping users navigate through the abundance of available data items. Recommender systems facilitate the selection of items by users by issuing recommendations for items they might like. In particular, recommender systems aim at providing suggestions to users or groups of users by estimating their item preferences and recommending those items featuring the maximal predicted preference.

Typically, depending on the type of the input data, i.e., user behavior, contextual information, item/user similarity, recommendation approaches can be classified as content-based, collaborative filtering, knowledge-based and hybrid ones. In content-based approaches, information about the features/content of the items is processed, and the system recommends items with features similar

to items a user likes. For example, if a Yelp user is always eating at sushi restaurants, she most likely likes this kind of food, so we can recommend her restaurants with the same cuisine. In collaborative filtering approaches, we produce interesting suggestions for a user by exploiting the taste of other similar users. For instance, if many users frequently go to Irish pubs after visiting an Italian restaurant, then we can recommend an Irish pub to a user that also shows preference for Italian restaurants. In knowledge-based approaches, users express their requirements, e.g., in terms of recommendation rules, and the system tries to retrieve items that are similar to the specified requirements. Finally, the hybrid recommender systems combine multiple of the aforementioned techniques to identify valuable suggestions.

Nowadays, recommendations have more broad applications, beyond products, like news recommendations, links (friends) recommendations [19], and more innovative ones like query recommendations [7], medicine recommendations [10], diverse venue recommendations [8], and others. There is also a lot of work on specific aspects of recommendations due to challenges beyond accuracy [1], like the cold start problem, the long tail problem and the evaluation of the recommended items in terms of a variety of parameters, like surprise, persistence [3] and serendipity [5]. More recently, many approaches that combine numerical ratings with textual reviews, have been proposed (e.g., [11]). For achieving efficiency, there are approaches that build user models for computing recommendations. For example, [13] applies subspace clustering to organize users into clusters and employs these clusters, instead of a linear scan of the database, for making predictions.

In this work, we focus on how to produce diversity-aware recommendations. Many times, suggestions for users may contain the same or very similar pieces of information. To avoid such redundant information, we opt to provide users with results that exhibit some diversity. Several definitions of diversity can be found in the research literature. Most of them can be classified into [6]: (i) content-based, selecting items that are dissimilar to each other, i.e., they do not contain overlapping information (e.g., [22]), (ii) novelty-based, selecting items that contain new information when compared to what was previously presented to the user (e.g., [4]), and (iii) semantic-based, selecting items that belong to different categories and topics (e.g., [2]).

We differ from the usual approaches, since we decouple diversity from strictly revolving around items, and we consider it as an orthogonal dimension that can be incorporated independently at different times in the workflow of the recommendation process. Specifically, we distinguish between item and user diversity, and study various definitions of them. Furthermore, we deliberate how to cope with the different notions of diversity, depending on

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ExploreDB'17, May 14-19, 2017, Chicago, IL, USA

© 2017 Association for Computing Machinery.

ACM ISBN 978-1-4503-4674-0/17/05...\$15.00

<https://doi.org/http://dx.doi.org/10.1145/3077331.3077341>

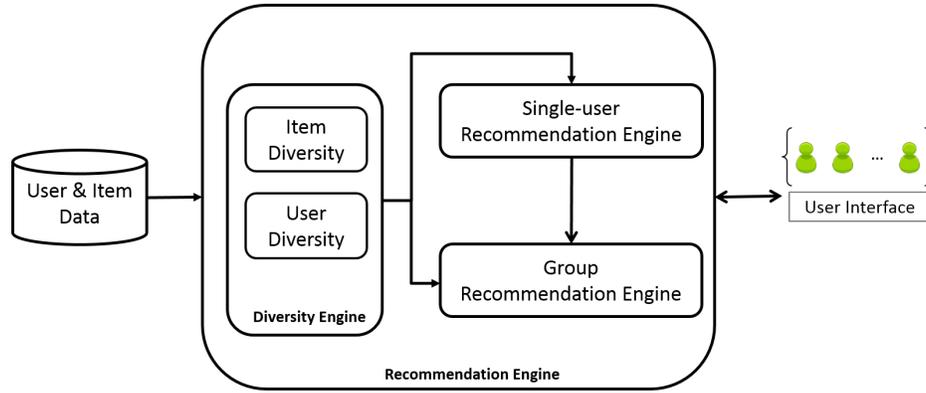


Figure 1: A diversity-aware recommender system.

whether we opt at diverse single-user or group recommendations. Figure 1 shows the general architecture of our diversity-aware recommender system.

The rest of the paper is organized as follows. Section 2 presents the basics of recommendations. Section 3 explores various ways for defining diversity between data items, and users. Section 4 focuses on how to identify diverse single-user recommendations, while Section 5 introduces diverse group recommendations. Section 6 concludes the paper and presents future work.

2 PRELIMINARIES

Assume a recommender system, where I is the set of items to be rated and U is the set of users in the system. A user $u \in U$ might rate an item $i \in I$ with a score $rating(u, i)$, e.g., in $[1, 5]$. Typically, the cardinality of the item set I is high and users rate only a few items. The subset of users that rated an item $i \in I$ is denoted by $U(i)$, while the subset of items rated by a user $u \in U$ is denoted by $I(u)$.

Typically recommender systems try to estimate a utility function in order to predict the user’s preference score for the unrated items, which we will refer to as $relevance(u, i)$, $u \in U, i \in I$.

3 DIVERSITY

In general, the problem of selecting a set of diverse things can be defined as follows. Given a set of n things and a budget k on the number of the diverse things to be selected, locate a subset of k things among the n available ones, such that, the diversity of the k things is maximized.

In this section, we explore various ways for computing the diversity between things. We distinguish things between the main entities that appear in a recommender system, namely, items and users. Figure 2 summarizes the various options in each entity.

3.1 Item Diversity

Many times, the data items that appear in a list of suggestions for users contain very similar pieces of information. To avoid such redundant information, we can provide to users results that exhibit

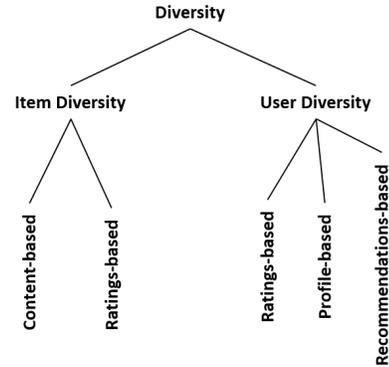


Figure 2: Diversity types.

some diversity. We consider two different approaches for quantifying diversity, namely, a content-based approach and a ratings-based approach.

The main process in the content-based approach, is to select items that are dissimilar to each other, i.e., they do not contain overlapping information (e.g., [18]). For quantifying the overlap between two data items, we can use a simple and relatively fast to compute measure that provides a good indication of the overlapping content of the two recommendations. Specifically, we use a Jaccard-like definition of distance, which measures dissimilarity between the content of the items. That is, given two data items i_x and i_y , consisting of the tokens V and W respectively, the diversity between i_x and i_y is:

$$div(i_x, i_y) = 1 - \frac{V \cap W}{V \cup W}. \tag{1}$$

The content-based approach focuses only on the items content for achieving diversity. Given that many times the description of a data item is poor or not available, we focus as well on a different notion of diversity that exploits the set of users gave ratings for particular items. We call this approach ratings-based approach.

Specifically, the main idea is that, given two data items i_x and i_y , the more overlap they have in their set of users that rated those items, i.e., $U(i_x)$ and $U(i_y)$, the more similar they are with each

other. That is, diversification can be fulfilled by identifying data items with small intersection between $U(i_x)$ and $U(i_y)$.

Then, given two items i_x, i_y , their diversity is defined as follows:

$$\text{div}(i_x, i_y) = 1 - \frac{|U(i_x) \cap U(i_y)|}{|U(i_x) \cup U(i_y)|} \quad (2)$$

To measure the overall diversity of a set of items, we define their set diversity based on their diversity from each other. A number of different definitions for set diversity have been proposed in the context of recommender systems; here we model diversity as the average distance of all pairs of elements in the set [22].

Definition 3.1 (Set Diversity). Given a set B of k items, $B = \{i_1, \dots, i_k\}$, the set diversity of B is:

$$\text{diversity}(B) = \frac{\sum_{x=1}^k \sum_{y>x}^k \text{div}(i_x, i_y)}{(k-1)k/2}.$$

3.2 User Diversity

In this section, we focus on defining diversity between users. Abstractly speaking, different definitions can be employed depending on the available data. Namely, we can measure the diversity between two users based on their ratings in a recommender system, on their profiles' content, or even based on the recommendations received in their previous interactions with the system.

When explicit user interactions with items, i.e., ratings, are available, the diversity between two users $u_x, u_y \in U$ can be computed as:

$$\text{div}(u_x, u_y) = 1 - \text{sim}U(u_x, u_y),$$

where $\text{sim}U(u_x, u_y)$ can be, for example, the Pearson correlation. Pearson correlation is a measure that is widely used for computing similarities between users in recommender systems. It actually measures the linear dependence between two users u_x and u_y : it has a value between +1 and -1, where +1 is total positive linear correlation, 0 is no linear correlation and -1 is total negative linear correlation.

$$\text{sim}U(u_x, u_y) = \frac{\sum_{i \in I'} (r(u_x, i) - \mu_{u_x})(r(u_y, i) - \mu_{u_y})}{\sqrt{\sum_{i \in I'} (r(u_x, i) - \mu_{u_x})^2} \sqrt{\sum_{i \in I'} (r(u_y, i) - \mu_{u_y})^2}},$$

where $r(u_x, i)$ is an abbreviation for $\text{rating}(u_x, i)$, μ_{u_x} is the mean of u_x 's ratings, and $I' = I(u_x) \cap I(u_y)$.

The intuition behind this type of diversity definition, is that when users rate differently the same items, they can be considered diverse.

Alternatively, in a content-like approach, users interests, or profiles, can be represented as structured, unstructured or semi-structured data. In structured profiles, there is a small number of attributes, each profile is described by the same set of attributes, and there is a known set of values that the attributes may have. Unlike structured profiles, in unstructured profiles, there are no attribute names with well-defined values. In between, in semi-structured profiles, there are some attributes with a set of restricted values and some free-text fields. A common approach to deal with free text (fields) is to convert the text to a structured representation, in which each token may be viewed as an attribute with an integer value indicating the number of times the token appears in the text. In a

more sophisticated approach, each token can be associated with a tf-idf value, $v(t, d)$, that is, for a token t in a text d , a function of: the frequency of t in d , the number of texts containing t , and the total number of texts. The intuition behind tf-idf is that the tokens with the highest values occur more often in that text than in other texts, and therefore are more important. In this scenario, $\text{sim}U(u_x, u_y)$ can be evaluated as the cosine similarity of the vectors representing the profiles of u_x and u_y , respectively.

A different way to measure diversity between users is to look at the outcome of the personalization process. As an example, consider that we can compare the recommendation lists that are produced for each user by the recommender engine. Different measures can be applied depending on whether we care only for the actual suggestions or for cases in which the ranking of the suggestions is important as well. In particular, given the recommendation lists A_{u_x} and A_{u_y} for the users u_x and u_y , their diversity can be measured by:

$$\text{div}(u_x, u_y) = 1 - \frac{|A_{u_x} \cap A_{u_y}|}{|A_{u_x} \cup A_{u_y}|}.$$

When focusing on a more detailed evaluation of the distance between the suggestions computed for two users, we can take also into account the position of each item in the recommendation list, and use the Kendall tau metric. According to it, for each pair of items (i, j) in the lists A_{u_x} and A_{u_y} , if i and j are in the same order in both lists, there is a penalty equal to 0. If i and j are in the opposite order, then let penalty be 1. Finally, there is a penalty p for cases in which one element does not appear in a list. The diversity between the users, in this case, is equal to the distance between their corresponding recommendation lists, i.e., equal to the summation of the penalties of all pairs.

Overall, among the many possible combinations of k users in a group G , the most diverse users are selected as follows.

Definition 3.2 (k-Diverse Users). Given a user group $G = \{u_1, \dots, u_n\}$, the k most diverse users of G is the set C^* for which:

$$C^* = \underset{\substack{C \subseteq G \\ |C|=k}}{\text{argmax}} \text{diversity}(C),$$

where

$$\text{diversity}(C) = \frac{\sum_{x=1}^n \sum_{y>x}^n \text{div}(u_x, u_y)}{(k-1)k/2}.$$

4 DIVERSE RECOMMENDATIONS

Recommender systems aim at giving recommendations to users by estimating their data item preferences and suggesting those items featuring the maximal predicted preference. In this section, we focus on how to identify single-user recommendations and on how to extend our definitions in order to define diversity-aware single-user recommendations.

4.1 Single-user Recommendations

A common way for producing recommendations for a user u_x , is to predict the missing ratings of all items i for u_x , using the previous ratings of i given by users similar to u_x . This method is called user-based collaborative filtering [17]. Similar users are located via a *similarity function* $\text{sim}U(u_x, u_y)$ that evaluates the proximity between $u_x, u_y \in U$ by considering their shared dimensions. We

use F_{u_x} to denote the set of the most similar users to u_x , hereafter, referred to as the *friends* of u_x .

Definition 4.1 (Friends). Let U be a set of users. The friends F_{u_x} , of a user $u_x \in U$ consists of all those users $u_y \in U$, which are similar to u_x w.r.t. a similarity function $simU(u_x, u_y)$ and a threshold δ , i.e., $F_{u_x} = \{u_y \in U : simU(u_x, u_y) \geq \delta\}$.

Overall, given a user u_x , his friends F_{u_x} and a similarity measure $simU$ for computing similarities between users, if u_x has expressed no preference for an item i , the relevance of i for u_x is estimated as:

$$relevance(u_x, i) = \frac{\sum_{u_y \in (F_{u_x} \cap U(i))} simU(u_x, u_y) rating(u_y, i)}{\sum_{u_y \in (F_{u_x} \cap U(i))} simU(u_x, u_y)}.$$

Item-based collaborative filtering resembles the general process of the user-based collaborative filtering approach. However, the basic idea here is to compute similarities between data items, instead of users, based on a function $simI(i_x, i_y)$, in order to make predictions. Specifically, to compute the relevance of an item i_x for a user u , we first locate the most similar items to i_x , say S_{i_x} , with respect to the ratings given to them by the users in U .

Overall, given an item i_x and its most similar items S_{i_x} , the relevance of i_x for u is estimated as:

$$relevance(u, i_x) = \frac{\sum_{i_y \in (S_{i_x} \cap I(u))} simI(i_x, i_y) rating(u, i_y)}{\sum_{i_y \in (S_{i_x} \cap I(u))} simI(i_x, i_y)},$$

where $simI(i_x, i_y) = 1 - div(i_x, i_y)$.

In both cases, after estimating the relevance scores of all unrated user items for a user u_x , the items A_{u_x} with the top- m relevance scores are recommended to u_x .

Differently, we can totally ignore the user ratings and produce recommendations by analyzing the descriptions of the items, so as to identify those items that are of particular interest to the users [15]. Similar to user profiles, data items can be represented as structured data, unstructured data or semi-structured data. The goal of the recommender here is to identify the items that are the most similar to the user's profile, via, for example, a cosine similarity function between vectors that represent data descriptions and user profiles. Therefore, as in the case of recommendations computed based on user ratings, the list A_{u_x} with the top- m most closely related items to the profile of u_x are recommended to him.

4.2 Diversity-aware Single-user Recommendations

Usually, suggestions for users contain very similar pieces of information. To avoid such redundant information, we opt to provide users with results that exhibit some diversity. The main way for doing so, is to select items that are dissimilar to each other, i.e., they do not contain overlapping information, e.g., by exploiting Equation 1.

Given that many times the items descriptions are not available, we use also a different definition of diversity that exploits the set of users that contribute towards producing the relevance scores of particular items. Specifically, given two different recommended items, the more overlap they have in their set of users employed for computing the relevance scores of the items, referred to as the *contributors* of the items, the more similar they are with each other.

This way, diversification can be fulfilled by identifying data items ranked highly in a list A_{u_x} for a user u_x , but their intersection in their contributors is small. A similar notion of diversity, used for providing recommendations explanations, is given in [20].

Formally, the contributors of an item i for a user u are defined as follows:

Definition 4.2 (Contributors). Let u be a user in U with friends F_u . The contributors of an item $i \in A_u$ are defined as the set of users in F_u that have rated i .

$$contributors(u, i) = F_u \cap U(i).$$

Interestingly, for the collaborative filtering scenario, we do not need any detailed descriptions about data items for computing diverse recommendations, which is very useful, since many times, especially in social networks, we have only partial descriptions about data items.

Next, based on the notion of *contributors*, we update Equation 1 so as to take into account the contributors of two data items, in order to compute their diversity.

Overall, for a user u_x , among the many possible combinations of k data items in his list of recommendations A_{u_x} , we choose the one with the most diverse items. That is, the k recommendations for u_x are selected according to the following definition:

Definition 4.3 (k-Diverse Recommendations). Given a user u_x and a set of n recommendations A_{u_x} , the k -diverse recommendations for u_x is the set B^* for which:

$$B^* = \underset{\substack{B \subseteq A_{u_x} \\ |B|=k}}{\operatorname{argmax}} diversity(B).$$

5 DIVERSE GROUP RECOMMENDATIONS

Recently, group recommendations that make recommendations to groups of users instead of single users, have received considerable attention. In this section, we broaden ideas used for computing group recommendations in order to consider the notion of diversity as well.

5.1 Group Recommendations

Since recommendations are typically personalized, different users are presented with different suggestions. However, there are cases where a group of people participates in a single activity. For instance, visiting a restaurant or a tourist attraction, watching a movie or a TV program and selecting a holiday destination are examples of recommendations well suited for groups of people. For this reason, recently, there are methods for group recommendations, trying to satisfy the preferences of all the group members. These methods can be classified into two approaches [9]. The first approach creates a joint profile for all users in the group and provides the group with recommendations computed with respect to this joint profile (e.g., [21]). The second approach aggregates the recommendations of all users in the group into a single recommendation list (e.g., [12, 16]).

Our work on group recommendations follows the second approach, since it is more flexible [9, 14] and, typically, offers opportunities for improvements in terms of efficiency. This way, our goal is to first estimate the relevance scores of the unrated items for each

user in the group, and then, aggregate these predictions to compute the suggestions for the group.

Definition 5.1. Let U be a set of users and I be a set of items. Given a group of users G , $G \subseteq U$, the group relevance of an item $i \in I$ for G , such that, $\forall u \in G$, $\exists \text{rating}(u, i)$, is:

$$\text{relevance}(G, i) = \text{Aggr}_{u \in G}(\text{relevance}(u, i))$$

As in [12], we consider three different designs regarding the aggregation method *Aggr*: (i) the *least misery design*, capturing cases where strong user preferences act as a veto (e.g., do not recommend steakhouses to a group when a member is vegetarian), (ii) the *fair design*, capturing more democratic cases where the majority of the group members is satisfied, and (iii) the *most optimistic design*, capturing cases where the more satisfied member of the group acts as the most influential one (e.g., recommend a movie to a group when a member is highly interested in it and the rest have reasonable satisfaction). In the least misery (resp., most optimistic) design, the predicted relevance score of an item for the group is equal to the minimum (resp., maximum) relevance score of the item scores of the members of the group, while the fair design, that assumes equal importance among all group members, returns the average score.

5.2 Diversity-aware Group Recommendations

Given a restriction k on the number of suggestions to be presented to the group, our goal here is to focus on the diversity-aware aggregation of the recommendations lists that were computed separately, for each user in the group. In particular, we would like to provide the group with p data items that as a whole cover many of the group members preferences and exhibit low redundancy. To achieve this, we resort to the following procedure that offers us the flexibility of fine-tuning the importance between the coverage of the users preferences and the diversity in selecting the top- p items.

For a group $G = \{u_1, \dots, u_n\}$, we use A_{u_1}, \dots, A_{u_n} to denote the sets of single-user recommendations computed for the users u_1, \dots, u_n . We want all users to contribute to the top- p group recommendations. The number of suggestions offered by a user u_i is captured by a function $f(i)$. Among the many possible combinations of the p data items that satisfy the constraints imposed by f , we choose the one with the most diverse items. Next, we define the top- k diverse group recommendations.

Definition 5.2. (**P-DIVERSE GROUP RECOMMENDATIONS**). Given a group $G = \{u_1, \dots, u_n\}$ and the set $A_G = \{A_{u_1}, \dots, A_{u_n}\}$, where A_{u_i} is the set of recommendations for $u_i \in G$, the p -diverse group recommendations for G is the set S^* for which:

$$S^* = \underset{\substack{S \subseteq \cup_{i=1}^n A_{u_i} \\ |S|=p}}{\text{argmax}} \text{diversity}(S),$$

such that, A_{u_i} contributes $f(i)$ data items to S^* , $1 \leq i \leq n$, with $\sum_{i=1}^n f(i) = p$.

Alternatively, we can work with the diversity-aware single-user recommendation lists of the users in the group (as defined in Section 4.2). Such approach can potentially increase the diversity of the produced suggestions, leading at the same time to a resulting set with lower overall relevance.

We use the function f as a parameter for tuning the numbers of suggestions each user contributes to the recommendations list of the group. For scenarios in which *fairness* is of high importance, group members equally contributes data items to the top- k set of recommendations. However, many times, in real life scenarios, there are cases in which some of the group members act as the most *influential* ones. For instance, we can recommend a particular brewery to 5 friends, where 2 of them are highly interested in it and the rest have a reasonable satisfaction. To capture such cases, we consider a monotonically decreasing function f that distinguishes between different clusters of users, from more influential ones to less influential ones. A high decrease rate of f leads to users from fewer clusters contributing to the top- k recommendations, i.e., mainly the most influential users will be used. A low decrease rate of f means that less results will be contributed by more users, i.e., the less influential users will contribute to the results as well. Diversity is calibrated through the number that expresses the size of the union of suggestions computed separately for all users in the group, out of which to select the p most diverse suggestions for the group.

Note that the definition of the p -diverse group recommendations can be easily updated, so as to take into consideration the diversity of the users that participate in the group. Intuitively, the idea here is to locate the k -diverse users, according to Definition 3.2, keep only the recommendations lists of the k users, and compute the p -diverse group recommendations based on their lists. This variation offers the improvement of reducing the search space of available recommendation lists to consider.

6 CONCLUSION AND FUTURE WORK

Several recent studies motivate the benefits diversity provides. In our case, the challenge is to introduce algorithms resulting in sets of suggestions that as a whole cover all the different needs of the users in question and not only focus on particular question aspects. We opt to extend our definitions of diversity in order to take into account issues related to novelty, referring to not seen things, and coverage, referring to things from different categories.

Overall, we envision a general processing model that puts humans in the core, in order to produce recommendations that take into consideration additional perspectives like transparency, fairness and privacy. Transparency facilitates the understanding of data. Exploration and explanation are two techniques that can be used for achieving transparency to assist users identify the what, where, when, how and who of a data item. For example, given the huge amounts of data available nowadays, exploration can support users by offering sophisticated discovery capabilities. Differently, explanations target at telling the story that the data has to say, by providing the reasons behind particular recommendations.

Fairness in data processing can be expressed as the lack of bias, where bias can come from data processing methods that reflect the preferences of the data scientists designing them. Regarding fairness in group recommendations, the goal is to locate, when possible or helpful, suggestions that include data items fair to the members of the group. That is, we should be able to recommend items that are both strongly related and fair to the majority of the group members.

Finally, keeping track of the users history in multiple Web applications, allows for studying changes in users tastes and identifying periodicity in their habits. Change is natural over time; in a recommender system, this means that the recommender should adapt to the changes in order to provide up-to-date recommendations to the users. However, we should also take into account that a long term users monitoring implies an extensive knowledge about their tastes and preferences, which might result in privacy risks for them.

ACKNOWLEDGMENTS

The work of Marialena Kyriakidi and Yannis Ioannidis has been partially supported by the Horizon 2020 Programme (H2020) of the European Commission under contracts #688865 (WholoDance) and #727188 (EMOTIVE) as well as by the European Social Fund (ESF) and Greek national funds through the Operational Program "Education and Lifelong Learning" of the National Strategic Reference Framework (NSRF) - Research Funding Programme "Thales" under contract #376908 (EICOS).

REFERENCES

- [1] Gediminas Adomavicius, Nikos Manouselis, and YoungOk Kwon. 2011. Multi-Criteria Recommender Systems. In *Recommender Systems Handbook*.
- [2] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. 2009. Diversifying search results. In *WSDM*.
- [3] Jöran Beel, Stefan Langer, Marcel Genzmehr, and Andreas Nürnberger. 2013. Persistence in Recommender Systems: Giving the Same Recommendations to the Same Users Multiple Times. In *TPDL*.
- [4] Charles L. A. Clarke, Maheedhar Kolla, Gordon V. Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. 2008. Novelty and diversity in information retrieval evaluation. In *SIGIR*.
- [5] Christian Desrosiers and George Karypis. 2011. A Comprehensive Survey of Neighborhood-based Recommendation Methods. In *Recommender Systems Handbook*. 107–144.
- [6] Marina Drosou and Evaggelia Pitoura. 2010. Search result diversification. *SIGMOD Record* 39, 1 (2010), 41–47.
- [7] Magdalini Eirinaki, Suju Abraham, Neoklis Polyzotis, and Naushin Shaikh. 2014. Query: Collaborative database exploration. *IEEE TKDE* 26, 7 (2014), 1778–1790.
- [8] Xiaoyu Ge, Panos K. Chrysanthis, and Konstantinos Pelechrinis. 2016. MPG: Not So Random Exploration of a City. In *MDM*.
- [9] Anthony Jameson and Barry Smyth. 2007. Recommendation to Groups. In *The Adaptive Web, Methods and Strategies of Web Personalization*.
- [10] Haridimos Kondylakis, Lefteris Koumakis, Eleni Kazantzaki, Maria Chatzimina, Maria Psaraki, Kostas Marias, and Manolis Tsiknakis. 2015. Patient Empowerment through Personal Medical Recommendations. In *MEDINFO*.
- [11] Julian McAuley and Jure Leskovec. 2013. Hidden Factors and Hidden Topics: Understanding Rating Dimensions with Review Text. In *RecSys*.
- [12] Eirini Ntoutsi, Kostas Stefanidis, Kjetil Nørnvåg, and Hans-Peter Kriegel. 2012. Fast Group Recommendations by Applying User Clustering. In *ER*.
- [13] Eirini Ntoutsi, Kostas Stefanidis, Katharina Rausch, and Hans-Peter Kriegel. 2014. Strength Lies in Differences: Diversifying Friends for Recommendations through Subspace Clustering. In *CIKM*.
- [14] Mark O'Connor, Dan Cosley, Joseph A. Konstan, and John Riedl. 2001. PolyLens: A recommender system for groups of user. In *ECSCW*. 199–218.
- [15] Michael J. Pazzani and Daniel Billsus. 2007. Content-Based Recommendation Systems. In *The Adaptive Web, Methods and Strategies of Web Personalization*. 325–341.
- [16] Senjuti Basu Roy, Sihem Amer-Yahia, Ashish Chawla, Gautam Das, and Cong Yu. 2010. Space efficiency in group recommendation. *VLDB J.* 19, 6 (2010), 877–900.
- [17] Jeff J. Sandvig, Bamshad Mobasher, and Robin D. Burke. 2008. A Survey of Collaborative Recommendation and the Robustness of Model-Based Algorithms. *IEEE Data Eng. Bull.* 31, 2 (2008), 3–13.
- [18] Kostas Stefanidis, Marina Drosou, and Evaggelia Pitoura. 2010. PerK: personalized keyword search in relational databases through preferences. In *EDBT*.
- [19] Zhijun Yin, Manish Gupta, Tim Wening, and Jiawei Han. 2010. LINKREC: a unified framework for link recommendation with user attributes and graph structure. In *WWW*.
- [20] Cong Yu, Laks V. S. Lakshmanan, and Sihem Amer-Yahia. 2009. It takes variety to make a world: diversification in recommender systems. In *EDBT*.
- [21] Zhiwen Yu, Xingshe Zhou, Yanbin Hao, and Jianhua Gu. 2006. TV Program Recommendation for Multiple Viewers Based on user Profile Merging. *User*

Model. User-Adapt. Interact. 16, 1 (2006), 63–82.

- [22] Mi Zhang and Neil Hurley. 2008. Avoiding monotony: improving the diversity of recommendation lists. In *RecSys*.