Knowledge-Based Systems 123 (2017) 154–162

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Diversity in recommender systems – A survey

Matevž Kunaver[a], Tomaž Požrl[b,*]

[a] University of Ljubljana, Faculty of Electrical Engineering, Tržaška 25, Ljubljana, Slovenia
[b] University of Ljubljana, Faculty of Mechanical Engineering, Aškerčeva 6, Ljubljana, Slovenia

ARTICLE INFO

Article history:
Received 8 September 2016
Revised 6 February 2017
Accepted 7 February 2017
Available online 8 February 2017

Keywords:
Personalization
Recommender system
Diversity
Survey

ABSTRACT

Diversification has become one of the leading topics of recommender system research not only as a way to solve the over-fitting problem but also an approach to increasing the quality of the user's experience with the recommender system. This article aims to provide an overview of research done on this topic from one of the first mentions of diversity in 2001 until now. The articles ,and research, have been divided into three sub-topics for a better overview of the work done in the field of recommendation diversification: the definition and evaluation of diversity; the impact of diversification on the quality of recommendation results and the development of diversification algorithms themselves. In this way, the article aims both to offer a good overview to a researcher looking for the state-of-the-art on this topic and to help a new developer get familiar with the topic.

1. Introduction

Today, users have access to a large number of items through a wide variety of devices and services. Users can access these items anywhere and any time due to increased functionalities offered by mobile platforms. In addition, users are now more involved in the item selection process by having direct control over which items they want to access.

The net effect is that every user gains access to a very large amount of items to choose from. This amount can also quickly become unmanageable and cause the user to have problems finding interesting items in a reasonable amount of time. The item selection process can therefore become cumbersome and complicated.

Recommender systems (RS) were developed to help with this problem by creating a selection of items that would be interesting to the user without requiring a large amount of interaction with him/her. Recommender systems work by tracking the interaction between the user and his/her selected content items. This information is then processed into a user model that is used to filter available content items in order to present the user with a selection of only the most appropriate items.

The development of RSs has started as early as 1980s [1] and has been an expanding research field ever since. Recommender systems are now found in almost any field that requires the user to make a decision - from marketing to shopping, from cinema to library. The research and development of recommender systems have also moved from simply developing new recommendation methods to fine-tuning these methods and finding ways to use additional information about the user in the recommendation process.

One of such improvements is the introduction of diversification into the recommendation process. Diversification is interesting as it is relatively new (first described by Bradley and Smyth [2] in 2001) and therefore offers a lot of potential for new developments. It is also interesting since it does not only try to solve the over-fitting problem but also requires a lot more human-oriented involvement than other RS related problems. That is due to the fact that each user perceives diversity differently (for example, one group of people will say that Star Trek and Star Wars are completely different types of films while another group perceives both film groups as Science-Fiction Action films) and this perception cannot be modelled without asking a lot of users to directly provide their definition of diversity.

This article aims to help new researchers who want to research diversification. As such it presents an overview of the most relevant work done on the subject of diversification. The authors have worked with diversification on several subjects (and presented their findings in several conference papers [3–5]) and felt that such an article would be of great benefit to any researchers who wish to start working in this field but do not know where to start (i.e. which articles to read). This article therefore aims to present most of the relevant literature sources in one place in an organised manner as well as providing some critical thoughts about the findings presented in each of them. The articles are divided into three

* Corresponding author.
E-mail addresses: matevz.kunaver@fe.uni-lj.si (M. Kunaver), tomaz.pozrl@fs.uni-lj.si (T. Požrl).

http://dx.doi.org/10.1016/j.knosys.2017.02.009
0950-7051/

groups according to their focus: definition, evaluation or algorithm development.

The rest of the article is organised as follows: Section 1.1 explains the procedure used to select papers presented in this review. Section 2 contains a brief overview of Recommender systems to acquaint a new reader with the field. This section also includes a description of the uses of RSs as well as some of the most known challenges in this field. Section 3 contains the core of the review and is divided into three sub-sections for better clarity of the article Section 3.1 covers papers that focus on the definition (and evaluation) of diversity. Section 3.2 covers papers that evaluate the impact of the diversification process on recommender systems (and their users). The final Section 3.3 presents actual diversification algorithms. Section 4 then presents the conclusions that the authors have gained from presenting this survey.

1.1. Article selection procedure

The selection process began with the help of Google Scholar using keywords: recommender systems, recommenders, diversity, diversification, algorithm(s), evaluation and impact. We preferred articles from more recent years and took care to remove duplicates, i.e. articles from the same authors that presented the same research from different angles. In such cases we selected the most comprehensive article, usually a full journal contribution.

This resulted in a selection of 67 articles that covered journals as well as conference papers. This selection was then narrowed down to articles that can either be accessed without any additional payment or those that are part of the IEEE Xplore Digital library and Science Direct. This resulted in 39 articles. After analysing their abstracts we created three groups of articles according to their problem statements: those dealing with the definition of diversity (11 articles), those measuring the impact of diversity (10 articles) and those presenting new diversification algorithms (18 articles).

After a detailed full text analysis of the articles in each group we removed several of them (2 from definition, 2 from impact and 8 from algorithms) as they were either duplicates from other articles included in this review or provided no useful information, i.e. did not contain a concrete diversity definition / algorithm.

2. A brief overview of recommender systems

The first mention of a RS was recorded in early 1980s when Salton [1] published an article presenting a word-vector based algorithm for searching amongst textual documents. Further development expanded these algorithms to a broader spectrum of content types from document search [6–9] to e-mail filtering [10] and personalized multimedia item retrieval [11–15].

Recommender systems can be divided into different groups according to several criteria such as the general user model creation approach (collaborative, content-based, hybrid as described by Hand et al. [6]) or the specific prediction generation algorithm (word vectors [16,17], decision trees [15], (naïve) Bayes classifier [15,17], k-nearest neighbours [11], support vector machines [18,19] etc.).

2.1. Scope of recommender systems

As mentioned above, the first recommender systems were used in the field of internet document search [6–9]. With the rapid growth of the internet and multimedia content availability, RS quickly spread to cover a much wider scope of services and content types.

2.1.1. Interactive digital television

With the introduction of digital broadcasting, set-top boxes and advanced TV services (i.e. video on demand, delayed viewing) consumers quickly began to appreciate personalization in this field [20,21]. The first and most basic personalized service introduced was the electronic personal guide (EPG) that alerts the user to the time and channel on which interesting content is being streamed.

2.1.2. Web-based storage of multimedia items

With the rise of online storage capacity and the advent of cloud services a lot of user-generated content has moved from personal computers to online services that offer users an easy way to store and share their items. Well-known examples of such service providers are Youtube [22], Shelfari [23], Facebook [24] and Goodreads [25]. The number of users and items is rising rapidly which has led to the problem of smart content retrieval. Most of these services are therefore offering a way for the user to find interesting items, from Goodreads recommendations to Youtube automatically playing the next video that should be interesting for the user.

2.1.3. Personalized advertisements

With the spread of the internet into most households the notion of personalized advertising has become a very lucrative business. Examples of such advertising range from filtering search results to including sponsored links as well as showing ads for content based on users previous internet activity [26,27]. By introducing personalized advertisements, the service provider can reduce the number of displayed advertisements yet retain a high amount of interest in the displayed products.

2.1.4. Shopping

When a user is presented with a large collection of articles he/she is presented with the dilemma as to what to choose since the sheer volume of items can deter him from trying something new.

Personalized services can be of great help to the user in this case. These services are offered by web-based shops since on-line shopping enables the shops to track the user's interests by simply tracking the user's browsing/shopping history. A good example of such a service is Amazon [12] which creates personalized article selection based on previous transactions and articles purchased by users who have similar taste ("customers who bought this article also bought... ").

2.2. The flow of the recommendation process

Most RSs follow the same steps during the recommendation process [28]. The first step is the analysis of the available information about the user (such as the list of viewed /rated items, history of user interface interaction). The results of such an analysis are then used to create a user model (or user profile) which stores the information required by the recommendation process in order to select the most appropriate item(s) for the current user. Once the selection is complete the items are presented to the user. The last (but also the most important) step is providing the user with a feedback mechanism that enables the RS to track the user's satisfaction with the presented recommendations and adjust the user model accordingly.

2.2.1. User model

The user model can be an independent data structure separate from the algorithm itself [15–17,29], a part of the algorithm itself [14,15] or it can simply be presented as a collection of user's past actions in the form of a vector of item-rating pairs.

Regardless of the implementation, the purpose of the user model is the storage of information required by the recommendation algorithm. This information can include basic (generic) information about the user (such as gender, age, name and surname, etc.), usage history (history of previous interactions with the system, which content items were viewed etc.), user feedback (ratings, comments and reviews), as well as algorithm specific data (SVM vectors, feature values, list of nearest neighbours, etc.). User models therefore differ according to the implemented recommendation algorithm.

These differences severely limit the portability of user models between different systems and content types. One can for instance transfer some information between RSs for books and films since they share some similarities in content description (genre for example), while one cannot transfer any information between a film and a news RS as demonstrated by Mehta and Nejdl [30].

2.2.2. Recommendation generation

Once the system retrieves (or creates) the current user's user model (i.e. the user who is currently accessing the system), it proceeds to use information stored in this model to search for relevant content items. The exact procedure depends on the implemented algorithm. Examples of such algorithms are the content-based (CBR), collaborative filtering (CF) and others as described by Burke [31].

Regardless of the approach the system only considers items that have not yet been viewed by the current user. The system calculates a relevance score for each of these items and stores them in the form of a list, with most relevant items at the top. The size of this list is then reduced to 10–20 of the most appropriate items that are then presented to the user [28,32].

The presentation of the list depends on the platform and content type and can range from a simple textual list to an adapted programme guide (EPG). Finally, the system also tracks whether the user selected any of the items from the list and collects feedback for such items as described by Ricci et al. [28]:

2.2.3. Feedback collection

Feedback collection is an essential part of the recommendation process since the RS works by analysing the user's interactions with the content items and the interface and cannot function without collecting this data. This data can be collected two different ways [28]:

- Explicitly: by directly asking the user to provide feedback, usually by providing a rating mechanism to the user ("rate this film on a scale from 0 to 10") or a questionnaire with which the user can express his/her satisfaction and opinion about the content item.
- Implicitly: by passively tracking the user's activity. This approach seems to be more user-friendly as it requires no active input from the user. It is however also less accurate for the very same reason. This approach tracks user's actions while he/she is viewing the selected content item. Examples of such actions are viewing the item, pausing and stopping at any time. The system then interprets these actions in order to decide whether the user liked or disliked the item.

Explicit feedback collection provides much more accurate information about the user's preferences but can reduce the quality of the user's experience with the system since it requires his/her active participation. On the other hand, the implicit approach is more user-friendly since it works passively and makes it possible for the user to remain ignorant of the fact that feedback is collected. The downside of implicit collection is the fact that the system requires a lot more time and data in order to work efficiently and accurately.

2.3. Research challenges

Research in the field of recommenders has moved from addressing the problem of calculating predicted ratings of relevant items to addressing issues that have surfaced now that RSs have become more widespread [31,33]. The list of most often encountered challenges is presented below.

2.3.1. Data sparsity

Regardless of the content type and/or recommendation all RSs face the problem of data sparsity [34]: working with user-item datasets that are mostly empty. The problem arises from the simple fact that it is impossible for each user to provide feedback for each existing item in the database. Having every possible rating would also defeat the purpose of RSs as it would mean that there is nothing left to recommend. The number of both items and users also constantly grows as new items are added and new users register in the system. RS algorithms must therefore be able to work with data tables that are mostly empty.

2.3.2. Cold start

Each new item / user presents a problem since the system cannot immediately create it's required model. In the case of a new item, the system is unable to decide if the item is relevant for any user until the item is either described by appropriate meta-data and/or has been rated by at least a few users. In the case of a new user the system can in most cases present only generic recommendations ('top rated items' for example) until the user provides some information about himself either in the form of demographic information or in the form of feedback about some of the content items stored in the system. This is known as the cold start problem [35].

2.3.3. Big data problem

Although RSs face the problem of data sparsity they also encounter the problem from the opposite side of the spectrum - the problem of Big Data. As stated above, most values are missing from the user-item tables since most of the users provide feedback on only a few selected content items, but since the numbers of users and content items in most systems, such as Youtube for example [36,37], range in millions if not billions this still means there is an immense data structure that usually cannot be processed without specialized algorithms [38].

2.3.4. Over-fitting

Once the system is able to generate recommendations consistently for each user a new problem arises as the system can begin recommending items from a very narrow spectrum of user's interest such as only football matches for example. This problem can occur when a user is trying to be helpful by providing explicit feedback only about the content he strongly likes. This leads to the creation of a very specific model that knows the exact the user preferences but is therefore unable to detect any other type of interesting items since the user has not shown any interest in it (i.e. has not rated any item of that type) [39]. This is known as Over-fitting or overspecialisation.

3. Diversification

Diversification has been introduced as one of the possible solutions of over-fitting and has in the last few years become a topic discussed by a large number of research groups, as evidenced by the large number of publications addressing this issue. Diversification has also become important enough to be featured at several workshops and challenges in worldwide conferences, such as the

Table 1
Definition and evaluation of diversity in the most relevant papers.

Ref	Definition Equation	Evaluation	Comparison
[2]	Diversity is the average dissimilarity between all pairs of items in the result set. $D = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(1-Similarity(c_i,c_j))}{n/2*(n-1)}$	Measuring diversity, similarity and relative gain of 7 RSs on a 1000 job advert dataset.	While the evaluations show the clear merit of the proposed metric it is highly dependant of the definition of item similarity (available meta-data).
[41]	Diversity is represented as the Gini coefficient - a measure of distributional inequality. $G = 1 - 2\int_0^1 L(u)du$	The measure was evaluated using a simulated environment that simulated user buying (recommended) items. The Gini coefficient of sales with recommendations was then compared to those without.	While this measure provides an interesting contrast to the intra-list diversity it remains highly domain specific; it is questionable whether it could be applied to environments such as film/music/book recommenders.
[42]	Diversity is part of the calculation of the nDCG measure - it has a direct impact on the calculated probability value. $G[k] = \sum_{i=1}^{m} J(d_k, i)(1-\alpha)^{r_{i,k-1}}$	Relevant document retrieval using the TREC 2006 dataset.	This measure is one of the first to combine ambiguity, diversity, redundancy and novelty into a single measure. The downside of the method is similar to that of the previous methods: it requires extensive data in order to correctly calculate the value of $J(d_k)$.
[43]	Diversity between two items is the product the item's relevance, similarity and places in the ranked list. $ILD(i_k\|u,R) = C'_k \sum_l disc(l\|k)p(rel\|i_l,u)dist(i_k,i_l)$	Measuring the diversity of three different RS using the MovieLens 1M dataset and several re-ranking diversifying algorithms.	This measure is fairly domain independent (authors present several different similarity calculations) and it also includes relevancy into the calculation, which is fairly important. The presented results are also promising.
[45]	User perceived diversity - questionnaire. N/A	20 participants answered the questionnaire about the diversity of recommended items.	It should be noted that this article does not offer a new definition of diversity. Instead it offers a way to collect information about how the user sees the diversity of recommended items, which has been ignored by most of the articles so far.
[46]	Diversity is presented as a nDCG measure (see [42]), with intent replacing the 'nugget' used in the original definition. $IA-nDCG = \sum p(a\|u)NDCG(u\|a), G[k] = \sum_a r(i_k,u)(1-\alpha)^{\sum_{l=0}^{k-1} r(i_l,u;a)}$	Calculation of diversity of diversified and non-diversified recommendation lists created using the Movielens and last.fm dataset.	The authors put a lot of effort into expanding the definition proposed in [42] and introduced intent as a novel way of detecting the user's requirements. Their results are promising but, as with many of the articles presented in this work, they lack a user study that would confirm whether the users can actually detect / appreciate the change in diversity.
[47]	Same as in [2], with the addition of a clearer definition of similarity between two items. $D = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(1-Similarity(c_i,c_j))}{n/2*(n-1)}$	250 volunteers evaluated the diversity of presented recommendations using a 7 point Likert scale.	This article offers a good user study that confirms that the definition used in [2] is viable for use in real-life applications.
[48]	Same as in [2]. $D = \frac{\sum_{i=1}^{n}\sum_{j=1}^{n}(1-Similarity(c_i,c_j))}{n/2*(n-1)}$	Last.fm set.	It should be noted that in this article the authors did not introduce a new diversity measure but rather used an existing one 'in reverse'. They used a change in diversity to detect a change in the user's context.
[50]	Diversity is a combination of genre coverage (how many different genres are present in the ranked list) and non-redundancy (genres do not repeat on the list). $BinomDiv(R) = Coverage(R) * NonRed(R)$	Movielens and Netflix dataset.	The proposed metric is novel and tries to include a lot of 'human behaviour' in it's calculation by using probability functions. It looks promising but has the downside of working only with genres. Should the approach expand to cover additional metadata, the metric might perform better than the current favourite - ILD (see [2]).

ESWC Linked Open Data-enabled Recommender Systems Challenge in 2014 [40]. The importance of diversity lies in the fact that it has a twofold purpose: increasing user satisfaction with the presented recommendations and mitigating the previously mentioned overfitting problem.

Bradley and Smith [2] were one the first to mention diversity by proposing the introduction of diversification in the recommendation procedure and also evaluating a new algorithm designed to diversify recommendations. Most of the advances in this field focus on either the definition of diversity, measuring the impact of diversification or on the development and evaluation of new diversification algorithms.

3.1. Definition and evaluation of diversity

Research that is focused on the definition of an appropriate evaluation measure that can be used to provide information about the diversity of recommendations lists starts with Bradley and Smyth [2] who define diversity as the opposite of similarity. This was followed up in 2007 by Fleder and Hosanagar [41] who performed an experiment that showed that most of RSs reduce the diversity of recommended items by focusing on the accuracy recommendations for each user. Clarke et al. [42] went a step further and tried to combine diversity and novelty into a new measure of retrieved document relevancy, which was based on Normalized Discounted Cumulative Gain measure. In 2011, Vargas [43] defined a diversity evaluation metric in a item browsing scenario that was based on a decreasing discount function. Castells et al. [44] further developed this idea with a metric that also considered the item position and relevance when determining the diversity of the recommendation list. Hu and Pu [45] on the other hand focused on determining how the recommendation diversity is perceived by users by conducting a live-subject study which showed that organization and categorical diversification play an important role in user-perceived diversity. Vargas [46] followed this by suggesting a formalization of diversification methods and evaluation techniques

Table 2
Impact of diversification on the quality of the recommendation process in the most relevant papers.

Ref	Diversification algorithm	Impact type/measure	Impact size
[51]	Re-ranking recommendation list using: item popularity (item ratings), reverse predicted rating value, item average rating, item absolute likeability, item relative likeability	Change in distribution of rec. items in terms of popularity: best sellers/long-tail items; number of long-tail items among the items recommended across all users	With 1% precision loss, percentage of rec. long-tail items increases from 16 to 32, with 5% loss perc. increases to 58
[52]	Maximization of parameterised combined objective function, representing a trade-off between diversity and matching quality, using Greedy and Relaxation and Quantization algorithms	Increased diversity; measured by evaluation of precision and recall against the novelty of the recommendations	
[53]	Diversity determinant is genre difference among films	User satisfaction and perceived diversity; user evaluation, ratings plus additional feedback	Users noticed high-diversity items, found them interesting; especially when placed in blocks of items
[54]	Not directly used; experiment studies how Openness to Experience may affect the diversity of the recommendations given by the participant	Diversity of the recommendations measured by author, genre and themes; binary for author/genre (0/1), 3 level for themes (0/0.3/1)	Participant personality did not affect recommendations diversity
[55]	ClusDiv method, items are clustered, rec. list built by selecting items from different clusters; aims to maximize diversity without decreasing accuracy; added tunable parameter to adjust diversity levels on the rec. lists	Increased diversity with no impact on accuracy; diversity measured by calculating z-diversity, accuracy by calculating recall	Comparable diversity increase (to greedy algorithm) with little recall decrease; much lower computational complexity
[56]	Not directly used; experiment surveys the diversity, novelty, accuracy, satisfaction and degree of personalization of various rec. algorithms	Increased diversity; objective measures: RMSE for accuracy, mean popularity rank for novelty, ILD for diversity; subjective measures: user survey, comparison of rec. lists produced by different rec. algorithms	Diversity positively impacted user satisfaction and thus choice of recommendation list (rec. algorithm)
[57]	SM - probabilistic specification maximizer model	Increased diversity; measured as interlist diversity	Outperforms classic Markov-based models in terms of diversity
[58]	Pareto-efficient multi-objective ranking and rec. list build	Increased diversity; diversity measured by distance based model ([43]), accuracy measured by precision and recall	The approach has the ability to balance each of the objectives according to the desired compromise, or to maximize all 3 objectives simultaneously

in a way that would also consider the rank and relevance as important aspects of the recommendation procedure.

In 2013 Castagos et al. [47] also performed a very interesting live user study that compared the user's acceptance and satisfaction with presented diversified recommendation lists and found that while diversification could reduce the user's acceptance rate it did increase the user's satisfaction with the system. Hullier et al. [48] performed a similar experiment featuring a music recommender that kept track of the user's preferences, context and the diversity of all music items listened to by the user. Jiang et al. [49] addressed the problem of diversity evaluation from a different angle and measured the diversity (and quality) of recommendations based on the choice probability instead of other proposed diversity measures. Their goal was to combine the evaluation of relevancy and diversity of recommendations into a single measure. Vargas et al. [50] focused on genre as one of key attributes of diversity evaluation and proposed a Binomial framework to measure genre diversity of each recommendation list.

Details about the definition (and equation if available) of diversity, implemented evaluation technique and our thoughts about the most relevant papers are given in Table 1.

3.2. Impact of diversification on the quality of the recommendation process for the most relevant papers

Several of research groups have focused on the effect of diversification on the quality of the recommendation procedure. In order to evaluate this they used a combination of proposed diversity measure and existing RS performance measures such as F-measure, MAE and NMAE [28] to determine how diversification impacts the overall performance of the RS.

Adomavicius and Kwon [51] evaluated several item ranking techniques and determined that many of them offer diverse recommendations while maintaining comparable levels of accuracy. Hurley and Zhang [52] also modelled the trade-off between diversity and accuracy as a binary optimization problem. Ge et al. [53] considered a different angle and performed a series of experiments to determine the impact of placement of high-diversity items in the recommendation list.

Tintarev et al. [54] performed an interesting study in which they focused not so much on the impact of diversification on the quality of recommendations but on the impact of diversification on users with different personality traits. They found that users who are more open to new experiences prefer a higher amount of diversity in their recommendations and vice versa. Aytekin and Karakaya [55] went a step further and offered direct control over recommendation diversity to the user in order to measure the user's preferences and satisfaction with the presented options. This was further explored by Ekstrand et al. [56] who performed a live user evaluation in which the user's provided feedback not only on the diversity of presented recommendation lists but also on the novelty, accuracy, satisfaction and the level of personalization. Javari and Jalili [57] devised an experiment in which they used a hybrid RS in which the trade-off between diversity and accuracy could be directly controlled. Ribeiro et al. [58] continued this trend by treating the recommender as a multi-objective recommendation problem that aims to combine several recommendation approaches in a way that tries to maximize accuracy as well as diversity.

Table 2 summarizes the results of the most relevant papers presented in this subsection.

3.3. Diversification algorithms

Several authors have undertaken research focused on developing new diversification algorithms and evaluating them using some of the measures described above. Ziegler et al. [59] were the first to use topic diversification to increase the diversity of recommendations at the cost of a corresponding drop in system accuracy. Slaney and White [60] focused on evaluating the diversity of recommender generated music playlists by projecting each song into a multidimensional feature space created by performing SVD on a combination of basic, statistic and rhythmic features extracted from each music item in the dataset. Adomavicius and Kwon [51] introduced a series of item ranking techniques that can

Table 3
Pseudo-code of the most relevant diversification algorithms.

Ref	Pseudo-code of the algorithm
[59]	1. Generate predictions (at least 5N for a final top-N recommendation list). 2. For each N+1 position item calculate the ILS (diversity) if this item was part of the top-N list. 3. Sort the remaining items in reverse (according to ILS rank) to get their dissimilarity rank. 4. Calculate new rank for each item as $r = a * P + b * P_d$, with P being the original rank, P_d being the dissimilarity rank and a, b being constants in range 0, 1]. 5. Select the top-N items according to the newly calculated rank.
[60]	1. Perform s MARSYAS analysis of all audio files to get basic features (Spectral centroid, rolloff and flux, Zero Crossings). 2. Calculate statistic characteristics of each feature (Mean of mean, Mean of standard deviation, Standard deviation of mean, Standard deviation of standard deviation). 3. Calculate 8 rhythmic features (High peak amplitude and beats-per-minute, Low peak amplitude and beats-per-minute, Peak ratio, 3 energy measures). 4. Perform statistical transformations (normalization for example) on all features. 5. Perform SVD followed by multi-class LDA to transform all features into a 2-D SVD space. 6. Fit a Gaussian probability model on each of the playlists (one song being one point in the SVD space). 7. Calculate the diversity of each playlist by calculating the volume of the fitted (ellipsoid) model.
[62]	1. Calculate predicted ratings for the current user using adjusted weight sum average. 2. Generate a list of Top N+S recommendations (N between 3 and 10; S between 1 and 10). 3. Calculate the TDE of each item as the sum of distances to all other (N+S-1) items on the list. 4. Remove S items with the lowest TDE score and so generate the Top N recommendations for the current user.
[63]	1. Collect genre information for each film in the dataset. 2. Calculate genre correlation for all film in the dataset by counting the number of occurrences of each possible pair of genres. 3. Normalize genre correlation values. 4. Collect user genre preferences explicitly. 5. Generate recommendations by calculating each predicted ratings as: 5.1 $\Sigma R_M{}^* r_{ij}$ with R_M being the films average ratings and r_{ij} being the genre correlations of all genres liked by user i and being attributed to film j. 5.2 Normalize the sum according to the number of user's genre preferences. 6. Select the top N items with the highest predicted rating and present them to the user.
[64]	1. Present the document (D) and topic (T) space as a bipartite graph, with edge weights representing the relevance of document D to topic T. 2. Sort the relevant documents (i.e. those considered for recommendation) according to their weighted coverage (sum of weights for all relevant topics). 3. For each document: 3.1 If the size of the recommendation list is less then desired, add document to list. 3.2 If the list is full check if replacing any of the documents on the list with the current one increases the overall diversity of the list. If so make the replacement that results in the largest increase in diversity.
[65]	1. Generate recommendations using a list of items ranked according to the average ratings of nearest neighbours. 2. Offer these items to the user and check if the user wants further recommendations. 3. If so, ask the user to select the item he/she thinks is currently most relevant. 4. Create a list of bN possible recommendations using Bounded Greedy algorithm and select the best N of them and add them to final recommendation list R. 4.1 Calculate the quality of each item as a combination of its predicted rating and ILD if included in R (ILD = 1 in the first iteration as R is empty). 4.2 Select the item with the highest quality and add it to R. 4.3 Repeat until there are N items on the list. 5. Present R to the user.
[66]	**1. Generate recommendations using the Matrix Factorization, kNN or mean average rating approach.** **2. Track if the user selected/rated any of the recommended items.** **3. Generate a new recommendations using updated information (all new ratings provided by the user).** **4. Calculate the diversity of the newly generated list of recommendations, by comparing it with the previous iteration (the list of recommendations presented to the user during his/her previous interaction with the system).** **Note: this approach measures how the diversity of the user's recommendation changes over time, not how diverse the items are that are presented to the user.**
[68]	1. Create a set of I recommendations for a given user using one of the existing CF recommenders. 2. Cluster these recommendations into priority-medoids according to comparison of ratings given to these items by other users. 3. Determine the cluster representative as the item with the highest predicted rating, not the one with the smallest distance to other cluster members. 4. Construct a cover tree using these cluster representatives. 5. Select a tree level that contains k items. 6. Present these items to the user as his/her recommendations.
[71]	**1. Create recommendations for all existing user using any CF approach.** **2. For each item that was recommended at least once, calculate a 5D score as a combination of: Accuracy, Balance, Coverage, Quality and Quantity of long-tail.** **3. For each user create a list of possible recommendations and:** **3.1 Order the list by predicted rating in order to assign a rank r to each item.** **3.2 Reverse order the list by each items 5D score in order to assign a rank r_{5D} to each item.** **3.3 Create a combined rank $r_n = r * r_{5D}$ and order the items by this rank.** **4. Present the top N items to the user.**
[72]	1. Create an MxM (item-by-item) adjacency matrix, with x_{mn} = number of times item m and n were both rated with a rating above users average. 2. For each user: remove all rows (items) that were not yet rated by the user. 3. For each remaining node (item): calculate Shanon entropy. 4. Sort nodes according to their weights (number of times occurred in matrix), then remove items with entropy value below threshold.

be used to substantially increase the diversity of recommendation without losing a large decrease of accuracy. They also proposed a graph-theoretic approach [61] that aimed to increase recommendation diversity based on maximum flow algorithms. Using a similar philosophy, Premchaiswadi et al. [62] proposed a new, hybrid ranking method called Total Diversity Effect Ranking that improves the overall recommendation diversity by considering the diversity effect of each item on the final recommendation list.

Choi and Han [63] focused on web queries and implemented an algorithm that calculates category correlations in order to provide the user with more diverse search results. Similarly, Abbassi et al. [64] increased the diversity of retrieved documents on aggregation websites by avoiding showing several documents that have the same category. They achieved this by using a (partition) matroid constraint algorithm. Bridge and Kelly [65] increased recommendation diversification using only collaborative data, which they

Table 4
Advantages and disadvantages of the most relevant diversification algorithms.

Ref	Advantages	Disadvantages
[59]	Flexible - can be used in any system that can define the distance (similarity) between 2 items	Diversification is applied after prediction generation, meaning that if the predicted items are not very diverse there will not be a noticeable increase in final diversity
[60]	Extremely detailed and precise definition of diversity, each item (song) must only be processed once	Music domain specific, and as authors themselves state - highly dependant on genre definitions, which could be too commercial
[62]	Flexible - can be used in any system that defines distance between 2 items, can be reversed	As with most diversity oriented systems - dependant on the definition of similarity / distance between two items
[63]	Does not require a lot of computing; needs only genre description of each item	Needs to recalculate values for each added item, only works if items include a meta-data description, requires explicit user information to function
[64]	Simple implementation, graphical representation	Requires topic, i.e. MD description of the content
[65]	Flexible - can be used in any system that defines distance between 2 items, can be reversed	Dependant on the definition of distance / similarity between two items, only considers diversity after the recommendation process is completed
[66]	**Interesting since it does not require tampering with the RS process itself but rather introduces diversity through multiple RS systems working in parallel**	**Requires several RS working at the same time**
[68]	Plug-in for any existing CF recommender, diversifies built recommendation list; declaratively balances the rating and the diversity of the RS	Does not use item semantic data
[71]	**Generic algorithm that can be applied to any existing system - it is an additional module fitted between the calculation of recommended ratings and presenting the user with the top-N list**	**Works when we are able to generate a large amount of recommendations, requires a lot ratings in order for the 5D score to be relevant**
[72]	Graphical representation, relative simplicity of algorithm, innovative diversity calculation	Still vulnerable to cold start - items that were never rated will never occur

achieved by using Hamming Distance for item comparison. Lathia et al. [66] studied the temporal aspect of diversity in collaborative RS. They analysed how collaborative RS recommendations change over time and the impact of these changes on the diversity of resulting recommendations. Mourao et al. [67] were dealing with a similar problem by introducing the Oblivion problem, which tries to exploit items that were relevant to user in the past but lost their relevancy through time. Boim et al. [68] estimated item diversity by comparing all the ratings given to the items by the users. Using this approach they were able to create item clusters and create recommendation lists with higher diversity. Vaishnavi et al. [69] tackled the problem of diversity in E-marketing by proposing an approach based on LCM version 2 and I-Tree.

Basille et al. [70] participated in the ESWC 2014 diversity challenge [40] and proposed an approach that combined several existing algorithms in order to achieve high diversity by using a very diverse set of semantic features. Ho et al. [71] used diversification as a way to address the problem of recommending seldom rated items in collaborative RS (i.e. the Long Tail problem) and found that they were able to recommend better items and improve the user's experience. Lee and Lee [72] introduced a graph theory based recommendation algorithm that used only the user's positively rated items to create an undirected graph and then used entropy to find novel and relevant recommendations.

Ren et al. [73] used an interdisciplinary approach and proposed a directed weighted conduction diversification algorithm that is based on economics (Gini index) and physics (heat conduction process) in order to improve the novelty and diversity of the recommendations. Bedi et al. [74] developed a clustering approach that is based on the SPRS metric in order to introduce a reasonable amount of diversity into news recommendations. It should be noted that their approach also offered explanations for 'unexpected' recommendations, which further served to increase the quality of the user experience. Di Noia et al. [75] turned the problem of diversity around by first modelling the user's tendency to select diverse items and then using this model to re-rank the user's Top-N recommendations.

The pseudo-code of the most relevant algorithms described in this subsection are given in Table 3.

We can see that all of the algorithms that create a diversified list of recommendations work by reordering recommendations after they have been generated using one of the existing approaches. This means that the results of such algorithms assume that recommendations are already diverse and just need to be reordered in order to achieve the maximum possible effect. Most of these algorithms are either collaborative [65,68,71,72] or content-based [59,62]. Noticeable exceptions are [63,64], which try to diversify items during the recommendation process. The remaining two algorithms ([60,66]) do not diversify recommendations but rather focus on measuring the overall diversity of each presented set of items without changing any of the items. Table 4 gives a comparison of the advantages and disadvantages of these algorithms, with the two of the most promising algorithms being marked with bold text.

4. Conclusions

Diversification in recommender systems has become one of the major strategies for solving the problem of over-fitting. Diversity has also been featured at several prominent conferences as either a special section or as a research challenge, which serves to prove its relevance.

We found that while all research groups agree that diversity is important and should be measured, few groups agree on the metric that should be used. While some metrics (such as the intra-list diversity) appear more often, the community still has not accepted one (or several) of them as the preferred diversity measurement. Once this is accomplished it will be a lot easier to quantify and compare results from different research groups.

Another notable result is that according to many research groups increasing diversity does not necessarily mean sacrificing accuracy as the use of the correct approach can actually lead to an increase in both. This is important as it offers greater flexibility in new algorithm designs and also means there is room for improvement.

Quite a few diversification challenges remain. The most outstanding ones, in our opinion, would be:

- Even though quite a few of the presented articles feature live user studies, we believe that they are not conclusive enough. While they did question the users about the perceived diversity of recommendations, they did not question how the users actually define diversity and use this knowledge to develop an appropriate diversity measure. Further users studies with expanded questions could therefore provide some benefit when developing new definitions of diversity.

- We also believe that since at least some aspect of the diversity is highly subjective (for example, most users see no difference between Star Wars and Star Trek, while others claim that they are completely different movie types), researches would benefit from including expert knowledge from the field of psychology in the development of new diversity measures.
- Most of the current diversity measures compare items according to their meta-data description. There are some exceptions to this (as presented in the algorithm section). The problem that arises in such case is how to handle systems that work with many different item types (collaborative RS for example) or with items that do not have all the meta-data available (user created content for example). A sort of generalization should therefore be proposed with such measures in order to simplify their use in different systems.
- Diversity should be considered during the recommendation process instead of being applied during the post-recommendation process, as is the case with most of the algorithms presented in this review. If a system suffers from overfitting we can come across a case where all of the recommendation items will be more or less the same, which will cause the diversification process to fail. Diversification should therefore be present from the start of the recommendation procedure and should be included in the process of ranking / calculation of predicted ratings.

Future research on diversification of recommender systems should therefore focus not only on developing new algorithms but also on finding an evaluation measure that would reflect the average user's perception of diversity.

References

[1] G. Salton, M. McGill, An Introduction to Modern Information Retrieval, McGraw-Hill, New York, USA, 1986.
[2] K. Bradley, B. Smyth, Improving recommendation diversity, in: Proc. 12th Nat. Conf. Artif. Intell. Cogn. Sci. (AICS-01), Maynooth, Ireland, 2001, pp. 75–84.
[3] M. Kunaver, Š. Dobravec, T. Požrl, A. Košir, Increasing top-20 search results diversity through recommendation post-processing, in: Posters, Demos, Late-breaking Results and Workshop Proceedings of UMAP, 2014a.
[4] M. Kunaver, T. Požrl, Š. Dobravec, U. Droftina, A. Košir, Increasing top-20 diversity through recommendation post-processing, in: Semantic Web Evaluation Challenge, Springer International Publishing, 2014b, pp. 188–192.
[5] M. Kunaver, Š. Dobravec, A. Košir, Using latent features to measure the diversity of recommendation lists, in: Proceedings of 38th International Convention MIPRO 2015, Croatian Society for Information and Communication Technology, Electronics, 2015, pp. 1477–1481.
[6] D. Hand, H. Mannila, P. Smyth, Principles of Data Mining, MIT Press, Cambridge, USA, 2001.
[7] B. Crabtree, J. Soltysiak, Identifying and tracking changing interests, Int. J. Digit. Libr. 2 (1) (1998) 38–53.
[8] J. Mirković, D. Cvetković, N. Tomča, Genetic algorithms for intelligent internet search,, IEEE TCCA Newsl. (1999).
[9] D. Mladenič, Text-learning and related intelligent agents: a survey, IEEE Intell. Syst. 14 (4) (1999) 44–54.
[10] T. Malone, K. Grant, F. Turbak, S. Brobst, M. Cohen, Intelligent information sharing systems, Commun. ACM 30 (5) (1987) 390–402.
[11] B. Bezzera, F. de Carvalho, G. Ramalho, J. Zucker, Speeding up recommender systems with meta-prototypes, in: Advances Artificial Intelligence,, in: Springer Lecture Notes Computer Science, 2002, pp. 227–236.
[12] Amazon webpage, (http://www.amazon.com). Accessed: July 2015.
[13] A. Buczak, J. Zimmerman, K. Kurapati, Personalization: Improving ease-of-use, trust and accuracy of a tv show recommender, in: Proc. AH'2002 Workshop Person. Future TV, Madrid, Spain, 2002.
[14] A. Difino, B. Negro, A. Chiarotto, A multi agent system for a personalized electronic program guide, in: Proc. AH'2002 Workshop Personal. Future TV, Madrid, Spain, 2002.
[15] K. Kurapati, S. Gutta, D. Schaffer, J. Martino, J. Zimmerman, A multi agent recommender, in: Proc. UM 2001 Workshop Person. Future TV, Sonthofen, Germany, 2001.
[16] G. Anhanger, T. Little, Data semantics for improving retrieval performance of digital news video systems, IEEE Trans. Knowl. Data Eng. 13 (3) (2001) 352–360.
[17] G. Uchygit, K. Clark, An agent based electronic program guide, in: Proc. AH'2002 Workshop Person. Future TV, Madrid, Spain, 2002.
[18] J. Yuan, Y. Yu, X. Xiao, X. Li, Svm based classification mapping for user navigation, IEEE Trans. Knowl. Data Eng. 13 (3) (2001) 352–360.
[19] M. Pogačnik, Uporabniku Prilagojeno Iskanje Multimedijskih Vsebin, Univ. of Ljubljana, Ljubljana, Slovenia, 2004 Ph.D. thesis.
[20] , Personalized Digital Television: Targeting Programs to Individual Viewers, in: L. Ardissono, A. Kobsa, M. Maybury (Eds.), 6, Kluwer Academic Publishers, Netherlands, 2004.
[21] A. Hanjalic, Extracting mood from pictures and sounds: towards truly personalized tv, IEEE Signal Process. Mag. 23 (2) (2006) 90–100.
[22] Youtube, (http://www.youtube.com). Accessed: July 2015.
[23] Shelfari, (http://www.shelfari.com). Accessed: July 2015.
[24] Facebook, (http://www.facebook.com). Accessed: July 2015.
[25] Goodreads, (http://www.goodreads.com). Accessed: July 2015.
[26] Google ads, (http://www.google.com/ads). Accessed: July 2015.
[27] J. Kim, B. Lee, M. Shaw, H.-L. Chang, M. Nelson, Application of decision-tree induction techniques to personalized advertisements on internet storefronts, Int. J. Electron. Commerce 5 (3) (2001) 45–62.
[28] , Recommender Systems Handbook, in: F. Ricci, L. Rokach, B. Shapira, P. Kantor (Eds.), 1, Springer, 2011.
[29] B. Manjunath, P. Salembier, T. Sikora, Introduction to MPEG-7 Multimedia Content Description Interface, John Wiley & Sons, Chichester, England,, 2002.
[30] B. Mehta, W. Nejdl, Intelligent distributed user modelling: From semantics to learning, in: Proc. UM 2007 Workshop Ubiq. Decent. User Model., Corfu, Greece, 2007, pp. 18–28.
[31] R. Burke, Hybrid recommender systems: survey and experiments, User Model. User-Adap. Interact. 12 (4) (2002) 331–370.
[32] G. Miller, The magical number seven, plus or minus two: some limits in our capacity for processing information, APA Psychol. Rev. 63 (2) (1956) 81–97.
[33] N. Good, J. Schafer, J. Konstan, A. Borchers, B. Sarwar, J. Herlocker, J. Riedl, Combining collaborative filtering with personal agents for better recommendations, in: Proc. 16th Nat. Conf. Artif. Intell., 11th Innov. Appl. of Artif. Intell. Conf. (AAAI '99/IAAI '99), Orlando, USA, 1999, pp. 439–446.
[34] Z. Huang, H. Chen, D. Zeng, Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering, ACM Trans. Inf. Syst. (TOIS) 22 (1) (2004) 116–142.
[35] A. Schein, A. Popescul, L. Ungar, D. Pennock, Methods and metrics for cold-start recommendations, in: Proc. 25th Ann. Int. ACM SIGIR Conf. Res. Dev. Inf. retrieval, 2002, pp. 253–260.
[36] Dmr - youtube usage statistics, (http://expandedramblings.com/index.php/youtube-statistics/). Accessed: July 2015.
[37] Youtube usage statistics, (https://www.youtube.com/yt/press/statistics.html). Accessed: July 2015.
[38] B. Sarwar, G. Karypis, J. Konstan, J. Riedl, Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering, in: Proc. 5th Int. Conf. Comput. Inf. Tech. (CIT '05), 1, 2002.
[39] Z. Abbassi, S. Amer-Yahia, L. Lakshmanan, S. Vassilvitskii, C. Yu, Getting recommender systems to think outside the box, in: Proc. RecSys 2009, 3rd ACM Conf. Recomm. Syst., 2009, pp. 285–288.
[40] , ESWC 2014 Semantic Web Eval. Challenge (Linked-Data Enabled Recomm. Syst.), in: V. Presutti, M. Stanković, E. Cambria, I. Contador, A.D. Iorio, T.D. Noia, C. Lange, D. Recupero, A. Tordai (Eds.), Springer, Crete, Greece, 2014.
[41] D. Fleder, K. Hosanagar, Recommender systems and their impact on sales diversity, in: Proc. EC '07 8th ACM Conf. Electr. Commerce, San Diego, CA, USA, 2007, pp. 192–199.
[42] C. Clarke, M. Kolla, G. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, I. MacKinnon, Novelty and diversity in information retrieval evaluation, in: Proc. 31st Int. ACM SIGIR Conf. Res. Dev. Infor. Retrieval (SIGIR '08), Singapore, Singapore, 2008, pp. 659–666.
[43] S. Vargas, New approaches to diversity and novelty in recommender systems, in: Proc. 4th BCS-IRSG Symp. Future Directions Inf. Access (FDIA 2011), Koblenz, Germany, 2011, pp. 8–13.
[44] P. Castells, S. Vargas, J. Wang, Novelty and diversity metrics for recommender systems: Choice, discovery and relevance, in: Int. Workshop Diversity Doc. Retrieval (DDR 2011) 33rd Eur. Conf. Inf. Retrieval (ECIR 2011), Dublin, Ireland, 2011, pp. 29–36.
[45] R. Hu, P. Pu, Helping users perceive recommendation diversity, in: Proc. ACM RecSys 2011, 1st Int. Workshop Novelty Diversity in Recomm. Syst. (DiveRS 2011), Chicago, IL, USA, 2011, pp. 43–50.
[46] S. Vargas, Novelty and Diversity Enhancement and Evaluation in Recommender Systems, Autonomous University of Madrid, Madrid, Spain, 2012 Master's thesis.
[47] S. Castagnos, A. Brun, A. Boyer, When diversity is needed...but not expected!, in: Proc. 3rd Int. Conf. Advances Inf. Mining Manage. (IMMM '13), Lisbon, Portugal, 2013, pp. 44–50.
[48] A. L'Huillier, S. Castagnos, A. Boyer, Understanding usages by modeling diversity over time, in: Proc. UMAP 2014, 22nd Conf. User Model., Adapt. and Personal. (late-breaking results), Aalborg, Denmark, 2014, pp. 81–86.
[49] H. Jiang, X. Qi, H. Sun, Choice-based recommender systems: a unified approach to achieving relevancy and diversity, Oper. Res. 62 (5) (2014) 973–993.
[50] S. Vargas, L. Baltrunas, A. Karatzoglou, P. Castells, Coverage, redundancy and size-awareness in genre diversity for recommender systems, in: Proc. RecSys 2014, 8th ACM Conf. Recomm. Syst., Foster City, Silicon Valley, USA, 2014, pp. 209–216.
[51] G. Adomavicius, Y. Kwon, Improving aggregate recommendation diversity using ranking-based techniques, IEEE Trans. Knowl. Data Eng. 24 (5) (2012) 896–911.

[52] N. Hurley, M. Zhang, Novelty and diversity in top-n recommendation analysis and evaluation, ACM Tran. Internet Tech. (TOIT) 10 (4) (2011) 14.
[53] M. Ge, F. Gedikli, D. Jannach, Placing high-diversity items in top-n recommendation lists, in: Proc. 9th Workshop Intell. Tech. Web Personal. & Recomm. Syst. (ITWP 2011), Barcelona, Spain, 2011.
[54] N. Tintarev, M. Dennis, J. Masthoff, Adapting recommendation diversity to openness to experience: A study of human behaviour, in: Proc. UMAP 2013 21st Conf. User Model., Adapt. Personal., Rome, Italy, 2013, pp. 190–202.
[55] T. Aytekin, M. Karakaya, Clustering-based diversity improvement in top-n recommendation, J. Int. Inf. Syst. 42 (1) (2014) 1–18.
[56] M. Ekstrand, F. Harper, M. Willemsen, J. Konstan, User perception of differences in recommender algorithms, in: Proc. RecSys 2014, 8th ACM Conf. Recomm. Syst., Foster City, Silicon Valley, USA, 2014, pp. 161–168.
[57] A. Javari, M. Jalili, A probabilistic model to resolve diversity accuracy challenge of recommendation systems, Int. J. Knowl. Inf. Syst. (2014).
[58] M. Ribeiro, A. Lacerda, E. de Moura, I. Hata, A. Veloso, N. Ziviani, Multi-objective pareto-efficient approaches for recommender systems, ACM Trans. Intell. Syst. Tech. 5 (4) (2015).
[59] C. Ziegler, S. McNee, J. Konstan, G. Lausen, Improving recommendation lists through topic diversification, in: Proc 14th Int. Conf. World Wide Web, Chiba, Japan, 2005, pp. 22–32.
[60] M. Slaney, W. White, Measuring playlist diversity for recommendation systems, in: Proc. MM '06 1st ACM Workshop Audio Music Comput. Multimedia, Santa Barbara, CA, USA, 2006, pp. 77–82.
[61] G. Adomavicius, Y. Kwon, Maximizing aggregate recommendation diversity: A graph-theoretic approach, in: Proc. ACM RecSys 2011, 1st Int. Workshop Novelty Diversity in Recomm. Syst. (DiveRS 2011), Chicago, IL, USA, 2011, pp. 3–10.
[62] W. Premchaiswadi, P. Poompuang, N. Jongswat, N. Premchaiswadi, Enhancing diversity-accuracy technique on user-based top-n recommendation algorithms, in: Proc. 2013 IEEE 37th Ann. Comput. Softw. Appl. Conf. Workshops (COMPSACW), Kyoto, Japan, 2013, pp. 403–408.
[63] S. Choi, Y. Han, A content recommendation system based on category correlations, in: Proc. 2010 5th Int. Multi-Conf. Compu. Global Inf. Tech. (ICCGI), Valencia, Spain, 2010, pp. 66–70.
[64] Z. Abbassi, V. Mirrokni, M. Thakur, Diversity maximization under matroid constraints, in: Proc. 19th ACM SIGKDD Int. Conf. Knowl. Disc. Data Mining, Chicago, IL, USA, 2013, pp. 32–40.
[65] D. Bridge, J. Kelly, Ways of computing diverse collaborative recommendations, in: Adaptive Hypermedia and Adaptive Web-Based Systems, in: Lecture Notes Computer Science, 4018, 2006, pp. 41–50.
[66] N. Lathia, S. Hailes, L. Capra, X. Amatriain, Temporal diversity in recommender systems, in: Proc. 33rd Int. ACM SIGIR Conf. Res. Dev. Inf. Retrieval, Geneva, Switzerland, 2010, pp. 210–217.
[67] F. Mourao, C. Fonseca, C. Araujo, W. Meira-Jr, The oblivion problem: Exploiting forgotten items to improve recommendation diversity, in: Proc. ACM RecSys 2011, 1st Int. Workshop Novelty Diversity in Recomm. Syst. (DiveRS 2011), Chicago, IL, USA, 2011, pp. 27–34.
[68] R. Boim, T. Milo, S. Novgorodov, Diversification and refinement in collaborative filtering recommender, in: Proc. 20th ACM Int. Conf. Inf. Knowl. Manage., Glasgow, UK, 2011, pp. 739–744.
[69] S. Vaishnavi, A. Jayanthi, S. Karthik, Ranking technique to improve diversity in recommender systems, Int. J. Comp. Appl. (IJCA) 68 (2) (2011) 20–24.
[70] P. Basile, C. Musto, M. de Gemmis, P. Lops, F. Narducci, G. Semeraro, Aggregation strategies for linked open data-enabled recommender systems, in: Proc. 2014 Europ. Semantic Web Conf. (Linked Open Data-enabled Recomm. Syst. Challenge), Rome, Italy, 2014.
[71] Y.-C. Ho, Y.-T. Chiang, Y.-J. Hsu, Who likes it more? mining worth-recommending items from long tails by modeling relative preference, in: Pro. WSDM 2014, 7th ACM Int. Conf. Web Search. Data Mining, New York, NY, USA, 2014, pp. 253–262.
[72] K. Lee, K. Lee, Escaping your comfort zone: a graph-based recommender system for finding novel recommendations among relevant items, Expert Sys. Appl. 42 (10) (2015) 4851–4858.
[73] X. Ren, L. Lu, R. Liu, J. Zhang, Avoiding congestion in recommender systems, New J. Phys. 16 (6) (2014).
[74] P. Bedi, S. Agarwa, A. Singhal, E. Jain, G. Gupta, A novel semantic clustering approach for reasonable diversity in news recommendations, in: Proc. Int. Conf. Comput. Intell. Data Mining (CIDM), Odisha, India, 2015, pp. 437–445.
[75] T. di Noia, V. Ostuni, J. Rosati, P. Tomeo, E. di Sciascio, An analysis of users' propensity toward diversity in recommendations, in: Proc. RecSys 2014, 8th ACM Conf. Recomm. Syst., Foster City, Silicon Valley, USA, 2014, pp. 285–288.