The Thirty-First International Florida
Artificial Intelligence Research Society Conference (FLAIRS-31)

Accurate and Diverse Recommendations Using Item-Based SubProfiles*

Mesut Kaya, Derek Bridge
Insight Centre for Data Analytics
University College Cork, Ireland
{mesut.kaya,derek.bridge}@insight-centre.org

Abstract

In many approaches to recommendation diversification, a recommender scores items for relevance and then re-ranks them to balance relevance with diversity. In intent-aware diversification, diversity is formulated in terms of coverage of aspects, where aspects are either explicit such as movie genres or implicit such as the latent factors found during matrix factorization. Typically, the same set of aspects is used across all users. In this paper, we propose a form of intent-aware diversification, which we call SPAD (SubProfile-Aware Diversification), and a variant called RSPAD (Relevance-based SPAD). The aspects we use in SPAD and RSPAD are subprofiles of the user's profile. They are not defined in terms of explicit or implicit features. We compare our methods to other forms of intent-aware diversification. We find that SPAD and RSPAD always improve accuracy (as measured by precision) and diversity (as measured by α-nDCG) even though the diversity metric in our experiments uses explicit features but SPAD and RSPAD make no use of them.

Introduction

It has long been recognized that it is not enough for recommendations to be accurate or relevant (McNee, Riedl, and Konstan 2006). Diversity is one response to uncertainty. A recommender cannot be certain of a user's short-term or longer-term interests, both because some user profiles are small and some, while they may not be so small, will contain preferences over different kinds of items. In the face of uncertainty, a diverse set of recommendations is more likely to contain one or more items that will satisfy the user.

Early work, within both Recommender Systems and Information Retrieval (IR), measures the diversity of a set of items as an aggregate of the all-pairs dissimilarity of the items within the set. Dissimilarity is measured by distance functions over item meta-data (such as movie genres), item ratings or latent factors. Sets of recommendations, or query results in IR, are re-ranked by considering the marginal contribution that would be made by adding an item to the result set (Carbonell and Goldstein 1998). The assumption in this early work is that a set of items that are dissimilar to each other is more likely to contain one or more items that satisfy the user's current needs or interests.

More recently within IR, there has been a body of research into what has been called *intent-aware diversification*. Intent-aware diversification seeks to include items in a query's result set, not on the basis of them being different from other items in the result set, but on the basis of explicitly seeking to cover different interpretations of the query. In this way, one or more of the items should cover the interpretation intended by the user. Intent-aware diversification has been adopted in Recommender Systems: instead of covering the different interpretations of an ambiguous query, the idea analogously is to cover the different tastes or interests revealed by the user's profile. The most common way to characterize a user's tastes is as a probability distribution over so-called aspects of the items. The same set of aspects is used across all users.

In this paper, we propose a new intent-aware diversification framework based on user subprofiles, rather than item features. This is advantageous because item features, such as genres, do not necessarily fully represent a user's tastes or interests and are not available in every recommendation domain. A subprofile is a subset of the items in a user's profile, each such subprofile representing one of the user's distinct tastes. Unlike the aspects used in earlier work, which are global across the set of users, subprofiles differ from user to user, making for a more personalized form of diversification.

Related Work

The dominant approach to diversification is greedy re-ranking.

Greedy re-ranking

The greedy re-ranking approach assumes the existence of a conventional recommender algorithm (which we will refer to as the *baseline recommender*), which, for user u, produces a set of recommended items, RS, and, for each item i in RS, a relevance score, $s(u, i)$ — the predicted relevance of recommended item i to user u. The greedy algorithm re-ranks RS by iteratively inserting into ordered result list RL the item i from RS that maximizes a function, $f_{obj}(i, RL)$; see Algorithm 1. f_{obj} is usually defined as a

*This publication is based upon works supported by the Science Foundation Ireland under Grant No. 12/RC/2289 which is co-funded under the European Regional Development Fund.

Algorithm 1 Greedy re-ranking algorithm

Input: RS, set of recommendations for user u, each with relevance score
Output: RL, ranked list containing all items in RS

$RL \leftarrow [\,]$
while $|RS| > 0$ **do**
  $i^* \leftarrow \arg\max_{i \in RS \setminus RL} f_{obj}(i, RL)$
  delete i^* from RS
  append i^* to the end of RL
return RL

linear combination of the item's relevance score and the contribution item i makes to the diversity of RL, $\mathrm{div}(i, RL)$, the trade-off between the two being controlled by a parameter λ ($0 \leq \lambda \leq 1$):

$$f_{obj}(i, RL) = (1 - \lambda)s(u, i) + \lambda\,\mathrm{div}(i, RL) \quad (1)$$

Most commonly, $\mathrm{div}(i, RL)$ is computed as the average (or sum) of the all-pairs intra-list distances (ILD). The distance between items, can be calculated from meta-data such as movie genres or book categories (Smyth and McClave 2001; Ziegler et al. 2005) or from item ratings data (Kelly and Bridge 2006).

The final recommendation comprises the top-N members of the re-ranked list, RL, where $N < |RL|$. Re-ranking using the ILD can result in a top-N that comprises items that are dissimilar to each other. The assumption behind this form of diversification is that dissimilar items will address the different interests of the user, but there is nothing in the operation of the system to explicitly ensure this. Recent approaches, going under the name *intent-aware diversification*, seek to select items that explicitly address different user interests.

Intent-aware diversification

Intent-aware methods for recommendation diversification (Vargas, Castells, and Vallet 2011; 2012; Wasilewski and Hurley 2016) assume a set of aspects $\mathcal{A}$ which describe the items and for which user interests can be estimated. The aspects might be explicit: for example, categories such as politics in a news recommender, or genres such as comedy in a movie recommender. Alternatively, aspects might be implicit, e.g. the latent factors found by a matrix factorization recommender system (Koren and Bell 2011).

User u's interests can be formulated as a probability distribution $p(a|u)$ for aspects $a \in \mathcal{A}$. The probability of choosing an item i from the set of recommendations RS given an aspect a of user u is denoted by $p(i|u, a)$. In Vargas's adaptation to recommender systems (Vargas Sandoval 2015) of Santos et al.'s Query Aspect Diversification framework (xQuAD) (Santos, Macdonald, and Ounis 2010), diversification can be achieved by re-ranking a conventional recommender's recommendation set as per Algorithm 1 and Equation (1) but with $\mathrm{div}(i, RL) = \mathrm{nov}_{\mathrm{xQuAD}}(i, RL)$ defined as:

$$\mathrm{nov}_{\mathrm{xQuAD}}(i, RL) = \sum_{a \in A} [p(a|u)p(i|u, a) \prod_{j \in RL} (1 - p(j|u, a))] \quad (2)$$

Consider the case where the aspects are explicit features $\mathcal{F}$, i.e. $\mathcal{A} = \mathcal{F}$, hence we will write $p(f|u)$ and $p(i|u, f)$ instead of $p(a|u)$ and $p(i|u, a)$. Let $\mathcal{F}_i$ be the subset of $\mathcal{F}$ that describes item i (e.g. the genres of movie i) and let I_u denote the items in user u's profile (i.e. the items she has interacted with). Then $p(f|u)$ can be estimated as:

$$p(f|u) = \frac{|\{i \in I_u : f \in \mathcal{F}_i\}|}{\sum_{f' \in \mathcal{F}} |\{i \in I_u : f' \in \mathcal{F}_i\}|} \quad (3)$$

$p(i|u, f)$, the probability of choosing i from a set of recommendations RS given explicit aspect f of user u, can be estimated as:

$$p(i|u, f) = \frac{\mathrm{ind}(i, f)s(u, i)}{\sum_{j \in RS} \mathrm{ind}(j, f)s(u, j)} \quad (4)$$

where $\mathrm{ind}(i, f) = 1$ if $f \in \mathcal{F}_i$ and 0 otherwise.

A possible weakness of xQuAD is that its formulation implies selection of a single item from the recommended set RS. In RxQuAD, Vargas et al. formulate a model based on maximizing relevance, rather than the probability of choosing a single item:

$$\mathrm{nov}_{\mathrm{RxQuAD}}(i, RL) = \sum_{a \in A} [p(a|u)p(rel|i, u, a) \prod_{j \in RL} (1 - p(rel|j, u, a)p(stop|rel))] \quad (5)$$

$p(rel|i, u, a)$ is the probability that user u finds recommended item i relevant when interested in aspect a. In the case of explicit features, this probability is obtained by mapping from relevance scores $s(u, i)$ using an exponential function (Vargas, Castells, and Vallet 2012). $p(stop|rel)$ is the probability that a user stops exploring a recommendation list conditional on finding a relevant item. Vargas observes that, to maximize α-nDCG, the best value for $p(stop|rel)$ is approximately equal to the value of α (Vargas Sandoval 2015).

The advantage in intent-aware approaches such as xQuAD and RxQuAD of using explicit aspects, such as movie genres, is their interpretability. A disadvantage is that they may be less accurate. The advantage, by contrast, of using implicit aspects, such as latent factors, is that they have been chosen for their predictive performance; their disadvantage is that they may be less interpretable (Wasilewski and Hurley 2016). Wasilewski & Hurley propose an intent-aware diversification method that is based on explicit aspects (and is hence interpretable) but in which the probabilities are learned (and hence are optimized for predictive performance) (Wasilewski and Hurley 2016). The learning is done by a constrained pLSA model (Hofmann 2004). They call their approach c-pLSA. More recently, the same authors presented an intent-aware framework that uses a minimum variance criterion based on portfolio theory from finance (Wasilewski and Hurley 2017).

What characterizes the work on intent-aware diversification in recommender systems that we have described so far is the use of a global set of aspects. The probabilities differ between users since they are computed from each user's profile, but the aspects are the same for all users. In our work,

we infer the aspects from the user's profile, making even the aspects personalized: the aspects for one user need not be the same for another.

SubProfile Aware Diversity

In this section, we explain our new approach to diversification in recommender systems, which we call SubProfile Aware Diversification (SPAD). It is a greedy re-ranking approach; it is intent-aware; but it is also personalized, based on subprofiles within the user's profile.

Let I be the set of all items. Subprofile detection works on positively-rated items in the user's profile. In the case of positive-only feedback, user u's profile, $I_u \subseteq I$, is the set of items she has interacted with (liked, clicked on, purchased, etc.). In the case of explicit ratings r_{ui} (e.g. 1-5 stars), then I_u must be defined in terms of items the user liked, which will usually involve thresholding the ratings, e.g. in our experiments, we use $I_u = \{i | r_{ui} \geq 4\}$. A user's subprofiles are subsets of I_u.

Subprofile extraction

Our approach to extracting user subprofiles comprises two steps. Firstly, we compute a set of top-n recommendations for this user each with what we call an explanation. We stress that these recommendations are not shown to the user. They are only being used to help us to extract subprofiles: their explanations are candidate subprofiles. Then, secondly, we prune the explanations to exclude any that are wholly contained in any of the others. The remaining explanations are the user's subprofiles. We will now explain the steps in more detail.

In the first step, we use an item-based nearest-neighbours recommender for implicit ratings (i.e. positive only ratings), which we designate IB+, and which is based on (Deshpande and Karypis 2004). For each candidate item $i \notin I_u$, IB+ finds items in the user's profile that have the candidate as one of their k-nearest-neighbours: $S_i^* = \{j \in I_u | i \in \mathrm{KNN}(j)\}$. IB+ scores each candidate by taking the sum of the similarities of the candidate to the items in S_i^*: $s(u,i) = \sum_{j \in S_i^*} \mathrm{sim}(i,j)$, using cosine similarity. The set S_i^* is the explanation for why i should be recommended.

Let E_u be the explanations for the n candidates whose scores, $s(u,i)$, are highest. We define the set of subprofiles for user u, $\mathcal{S}_u$, to be those members of E_u that do not contain any other members of E_u, i.e. $\mathcal{S}_u = \{S_i^* \in E_u | \neg \exists S \in E_u, S_i^* \subset S\}$. We obtain $\mathcal{S}_u$ from E_u by sorting the explanations in descending order of size and greedily retaining those that are not subsets of any already chosen. Note that, since $|E_u| = n$, there can be no more than n subprofiles.

Subprofiles as aspects

In the work on intent-aware diversification that we described earlier, the same set of aspects $\mathcal{A}$ was used for all users. In SPAD, aspects are user-specific: user u has set of aspects $\mathcal{A}_u$. And, in the earlier work, aspects were often based on explicit features $\mathcal{F}$, i.e. $\mathcal{A} = \mathcal{F}$. In SPAD, aspects are user subprofiles, i.e. $\mathcal{A}_u = \mathcal{S}_u$. Each subprofile $S \in \mathcal{S}_u$ contains a set of items from I_u. Different subprofiles can be of different lengths; the number of subprofiles can differ across users.

We produce a set of recommendations RS using some recommender. This can be any recommender that produces scores, $s(u,i)$, for the items that it recommends. (In particular, just because we detect subprofiles using IB+, we are not obliged to use IB+ to produce RS.) The set RS is greedily re-ranked (Algorithm 1) using the objective function given as Equation (1) with $\mathrm{div}(i, RL) = \mathrm{nov}_{\mathrm{xQuAD}}(i, RL)$ (Equation (2)). What differs is the computation of the probabilities used in Equation (2). Given that aspects are now subprofiles, we will write $p(S|u)$ and $p(i|u,S)$ instead of $p(a|u)$ and $p(i|u,a)$ for $S \in \mathcal{S}_u$.

Analogously to Equation (3), $p(S|u)$ can be estimated as:

$$p(S|u) = \frac{|S|}{\sum_{S' \in \mathcal{S}_u} |S'|} \tag{6}$$

$p(i|u,S)$, the probability of choosing i from a set of recommendations RS given subprofile S of user u, can be estimated as:

$$p(i|u,S) = \frac{\mathrm{ind}(i,S) s(u,i)}{\sum_{j \in RS} \mathrm{ind}(j,S) s(u,j)} \tag{7}$$

But here there is a problem. We want $\mathrm{ind}(i,S)$ to be 1 when item i is 'related to' subprofile S, and 0 otherwise. We cannot just use membership ($i \in S$), because i is a candidate recommendation and therefore will not in general already be a member of the user's profile or its subprofiles. Accordingly, in SPAD we define $\mathrm{ind}(i,S)$ as follows:

$$\mathrm{ind}(i,S) = \begin{cases} 1 & \text{if } i \in \bigcup_{j \in S} \mathrm{KNN}(j) \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

where $\mathrm{KNN}(j)$ is the set of j's k-nearest-neighbours in I. In other words, i must be a neighbour of a member of S.

Analogously to the relationship between xQuAD and RxQuAD (Vargas, Castells, and Vallet 2012), we can define RSPAD, by replacing aspects a by subprofiles S in Equation (5).

Experiments

Datasets

The datasets we use are the MovieLens 1M dataset and the LastFM dataset[1]. But we use the modified versions of these datasets that were used in (Kaminskas and Bridge 2016). For example, the listening event frequencies in the LastFM dataset are converted into ratings on the scale 1-5 and the dataset is augmented with additional meta-data (user-generated tags). Table 1 summarizes their characteristics.

[1] http://grouplens.org/datasets/movielens,http://www.dtic.upf.edu/ocelma/MusicRecommendationDataset/lastfm-1K.html

MovieLens	6040 users	3706 items	∼1M ratings
	18 genres in total; avg. 1.65 per movie		
LastFM	992 users	7280 items	∼500k ratings
	71833 tags in total; avg. 8 per artist		

Table 1: Datasets

Recommender systems

We compare SPAD and RSPAD with the other diversification techniques available in the RankSys library[2]: MMR (Carbonell and Goldstein 1998), xQuAD (Vargas, Castells, and Vallet 2011), RxQuAD (Vargas, Castells, and Vallet 2012) and c-pLSA (Wasilewski and Hurley 2016).

All of these approaches to diversification use greedy re-ranking, therefore they need a baseline recommender, whose recommendation sets are re-ranked. We use the following baseline recommenders (again using their RankSys implementations): probabilistic latent semantic analysis (pLSA) (Hofmann 2004), a fast alternative least-squares matrix factorization recommender (MF) (Pilászy, Zibriczky, and Tikk 2010), and a factorization machine that uses Bayesian pairwise loss for ranking (FMBPR) (Bayer 2015).

Three baselines paired with seven re-ranking approaches (the six above but also none at all) gives 21 systems to compare on each dataset. However, we were unable to obtain results for c-pLSA on the LastFM dataset because the implementation is based on the maximum possible features (71833 user-generated tags), whereas the other re-ranking approaches only depend on the number of distinct features that describe items in the user's profile.

All of the baseline algorithms have hyperparameters. In our cross-validation methodology (later section), as in (Vargas, Castells, and Vallet 2012), we select hyperparameter values that maximize precision for top-10 recommendations. For pLSA, MF and FMBPR, we choose the number of latent factors (d) from $V = \{10, 30, 50, 70, 90, 110, 130, 150, 170, 190, 210\}$. FMBPR's learning rate (lr) and regularization parameters ($regW$ and $regM$) are chosen from $\{0.01, 0.001\}$, and MF's confidence level (α) is chosen from $\{1, 2, \ldots, 10\}$. This resulted in the following values:

- pLSA: $d = 50$ for MovieLens; $d = 30$ for LastFM.
- MF: $d = 30, \alpha = 1.0$ for MovieLens; $d = 30, \alpha = 1.0$ for LastFM.
- FMBPR: $d = 190$, $lr = 0.01$, $regM = 0.01$, $regW = 0.001$ for MovieLens; $d = 10$, $lr = 0.01$, $regW = 0.01$, $regM = 0.001$ for LastFM.

We use the baselines to produce recommendation sets RS and greedily re-rank each RS to give a ranked list RL from which we recommend the top-N. MMR uses ILD with distance defined as the complement of Jaccard similarity on the item meta-data (genres, keywords, tags). xQuAD, RxQuAD and c-pLSA all use the meta-data as explicit aspects.

As explained earlier, SPAD and RSPAD use an item-based recommender, IB+, when extracting subprofiles. IB+ has its own hyperparameters: the number of neighbours (k, referred to below as k_{IB}) and the number of recommendations to make (n), both of whose values we select from V. Equation (8) also has hyperparameter k (referred to below as k_{DC}) and its value is also selected from set V.

The values selected for the MovieLens dataset are:

- pLSA: $n = 50$ for SPAD, and $n = 70$ for RSPAD; $k_{IB} = 10$, $k_{DC} = 30$ for both.
- MF: $n = 50$ for SPAD, $n = 90$ for RSPAD; $k_{IB} = 10$ and $k_{DC} = 50$ for both.
- FMBPR: $n = 30$, $k_{IB} = 10$ and $k_{DC} = 10$ for both SPAD and RSPAD.

The values selected for the LastFM dataset are:

- pLSA: $k_{IB} = 30$ for SPAD, $k_{IB} = 50$ for RSPAD and $n = 110$, $k_{DC} = 10$ for both.
- MF: $k_{IB} = 10$, $k_{DC} = 30$ for SPAD, $k_{IB} = 50$ and $k_{DC} = 10$ for RSPAD; $n = 110$ for both.
- FMBPR: $k_{IB} = 30$, $k_{DC} = 10$ and $n = 110$ for both SPAD and RSPAD.

We emphasize that all of the hyperparameter values are found using cross-validation (later section).

Evaluation measures

For accuracy we measure precision and for diversity we measure α-nDCG (Clarke et al. 2008), which is a redundancy-aware version of nDCG. (Again, we use RankSys implementations.) Diversity is measured with respect to the explicit features $\mathcal{F}$ (the meta-data).

We evaluate top-N for $N = 10$ recommendations and, we treat test set items with a rating of 4 or 5 as being relevant. In α-nDCG, we use $\alpha = 0.5$. (Hence, following the argument from (Vargas Sandoval 2015) given earlier, we use $p(stop|rel) = 0.5$ in RxQuAD and RSPAD too.)

Methodology

In experiments, we randomly partition the ratings into training, validation and test sets such that 60% of each user's ratings are in the training set, 20% of them are in the validation set and 20% are in the test set. Results are averaged over five runs with different splits.

We select hyperparameter values for each baseline recommender that optimize precision on the validation sets (Vargas, Castells, and Vallet 2012). Then, we train the baselines using the selected hyperparameter values on the training+validation sets and, for each user, generate a recommendation set RS, where $|RS| = 100$. Then, we re-rank each RS to produce ranked lists RL using each of the re-ranking algorithms. In doing this, we try different values for λ, which controls the balance between relevance and diversity (Equation 1), from $[0.1, 0.2, \ldots, 1.0]$. Then, from each RL, we select the top-N recommendations, $N = 10$.

Results

The results for the experiments on the MovieLens dataset are in Table 2. In each block of the table, results for the baseline are presented first, and then results for each of the

[2] https://github.com/RankSys

Table 2: MovieLens results. The best result for each metric is highlighted in bold for each block. The value of λ that optimizes α-nDCG for each baseline and re-ranking strategy is given. All of the results are statistically significant with respect to their baseline (Wilcoxon signed rank with $p < 0.05$) except those shown in italics.

		Metrics		% change over baseline	
	λ	Precision	α-nDCG	Precision	α-nDCG
pLSA		0.2639	0.2842		
MMR	0.3	*0.2635*	0.2913	-0.17%	+2.47%
xQuAD	0.7	0.2456	**0.3428**	-6.93%	+20.61%
RxQuAD	1.0	0.2452	0.3341	-7.1%	+17.53%
c-pLSA	0.5	0.2763	0.3075	+4.69%	+8.19%
SPAD	1.0	0.2783	0.3233	+5.44%	+13.74%
RSPAD	1.0	**0.2797**	0.3213	+5.98%	+13.05%
MF		0.2916	0.3197		
MMR	0.2	0.2906	0.3243	-0.34%	+1.43%
xQuAD	0.5	0.2739	**0.3668**	-6.08%	+14.72%
RxQuAD	0.7	0.2629	0.3586	-9.85%	+12.15%
c-pLSA	0.3	**0.2978**	0.3292	+2.11%	+2.96%
SPAD	0.6	0.2947	0.337	+1.04%	+5.39%
RSPAD	0.7	0.2945	0.3368	+1.00%	+5.32%
FMBPR		0.2655	0.3025		
MMR	0.2	0.2649	0.3068	-0.22%	+1.42%
xQuAD	0.4	0.2534	**0.3376**	-4.56%	+11.61%
RxQuAD	0.5	0.2429	0.3272	-8.48%	+8.16%
c-pLSA	0.3	0.2754	0.3157	+3.75%	+4.38%
SPAD	0.4	**0.2765**	0.321	+4.16%	+6.13%
RSPAD	0.5	0.2736	0.3178	+3.07%	+5.06%

re-ranking methods are given. For each method, we report the results using the value of λ that gives highest α-nDCG on the validation set (Wasilewski and Hurley 2016).

Consider precision first. For each baseline (i.e. each block in the table), with only one exception, SPAD or RSPAD has highest precision. The exception is where MF is the baseline, when c-pLSA is slightly more accurate. These results show that both SPAD and RSPAD give high precision for the MovieLens dataset.

Next, consider the diversity metric, α-nDCG. SPAD and RSPAD are at a disadvantage since they make no use of the explicit features. Indeed, α-nDCG is a metric that is very similar to what is used for re-ranking in xQuAD and RxQuAD. Even so, SPAD and RSPAD have higher diversity than all baselines and have higher diversity than MMR and c-pLSA re-ranking. For diversity, xQuAD is the best re-ranking method for all three baselines, and both xQUAD and RxQUAD achieve higher diversity than both SPAD and RSPAD. However, when we look at percentage changes, we see that xQUAD and RxQUAD achieve their diversity performance at the expense of the largest decreases in precision. Surprisingly, c-pLSA, SPAD and RSPAD increase accuracy along with diversity.

The results for the LastFM dataset are in Table 3. Recall that c-pLSA is missing from these results because we were unable to run it to completion on a dataset with so many explicit features (tags). Here, SPAD has the highest precision in each block, and RSPAD the second highest.

Table 3: LastFM results. The best result for each metric is highlighted in bold for each block. The value of λ that optimizes α-nDCG for each baseline and re-ranking strategy is given. All of the results are statistically significant with respect to their baseline (Wilcoxon signed rank with $p < 0.05$) except those shown in italics.

		Metrics		% change over baseline	
	λ	Precision	α-nDCG	Precision	α-nDCG
pLSA		0.3804	0.3426		
MMR	0.3	0.3773	0.3499	-0.84%	+2.13%
xQuAD	0.5	0.41	**0.3847**	+7.78%	+12.28%
RxQuAD	0.8	0.3993	0.3631	+4.96%	+5.96%
SPAD	0.9	**0.4199**	0.3823	+10.37%	+11.59%
RSPAD	0.6	0.4162	0.3773	+9.39%	+10.13%
MF		0.4654	0.4244		
MMR	0.3	0.4545	0.4312	-2.35%	+1.62%
xQuAD	0.3	0.4701	**0.4354**	+1.01%	+2.61%
RxQuAD	0.3	*0.4654*	0.4253	0.0%	+0.22%
SPAD	0.3	**0.4733**	0.429	+1.69%	+1.11%
RSPAD	0.2	0.4716	0.4277	+1.33%	+0.8%
FMBPR		0.3737	0.3409		
MMR	0.1	*0.3727*	0.3432	-0.27%	+0.67%
xQuAD	0.4	0.3972	0.3758	+6.28%	+10.23%
RxQuAD	0.5	0.3856	0.3529	+3.17%	+3.52%
SPAD	0.5	**0.4126**	**0.3793**	+10.41%	+11.26%
RSPAD	0.5	0.4088	0.3724	+9.38%	+9.25%

Again, despite making no use of explicit features, SPAD and RSPAD increase diversity. In fact, SPAD gives the highest α-nDCG where FMBPR is the baseline algorithm and is competitive with xQuAD (which performs best) where pLSA is the baseline. Interestingly, all re-ranking methods except MMR increase precision (or leave it unchanged in one case for RxQUAD) as well as increasing diversity. None increase precision as much as SPAD and RSPAD, which arguably achieve the best balance between increased precision and increased diversity.

Conclusions and Future Work

We have presented a personalized form of intent-aware diversification, in which the aspects to be covered by the re-ranked recommendations are subprofiles of the user's profile, each representing a distinct user taste. We extract subprofiles using an approach that extends an item-based recommender (IB+), and estimate interest in these personalized aspects in the manner of the xQuAD and RxQuAD systems (Vargas, Castells, and Vallet 2011; Vargas Sandoval 2015). We compare our approaches, SPAD and RSPAD, to other re-ranking approaches on two datasets. We find that they produce recommendations that are both among the most accurate in the experiments and are diverse. The latter is noteworthy since the diversity metric that we use in the evaluation is defined in terms of explicit features. It may favour competitor algorithms, which re-rank in ways that are sensitive to those features. SPAD and RSPAD, by contrast, make no use of meta-data.

In the future, we will compare all the systems with datasets with different amounts of meta-data. We will also

explore the interpretability of SPAD's recommendations: we will look to see how we can explain SPAD's recommendations in terms of subprofiles. Since subprofiles are just sets of items, we can take inspiration from the work on item-based explanations (as used, for example, in amazon.com), which has been proven to produce effective explanations (Bilgic and Mooney 2005).

References

Bayer, I. 2015. Fastfm: a library for factorization machines. *arXiv preprint arXiv:1505.00641*.

Bilgic, M., and Mooney, R. J. 2005. Explaining recommendations: Satisfaction vs. promotion. In *Beyond Personalization Workshop, IUI*, volume 5, 153.

Carbonell, J., and Goldstein, J. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Procs. of the 21st ACM SIGIR International Conference on Research and Development in Information Retrieval*, 335–336.

Clarke, C. L. A.; Kolla, M.; Cormack, G. V.; Vechtomova, O.; Ashkan, A.; Büttcher, S.; and MacKinnon, I. 2008. Novelty and diversity in information retrieval evaluation. In *Procs. of the 31st ACM SIGIR International Conference on Research and Development in Information Retrieval*, 659–666.

Deshpande, M., and Karypis, G. 2004. Item-based top-N Recommendation Algorithms. *ACM Trans. Inf. Syst.* 22(1):143–177.

Hofmann, T. 2004. Latent semantic models for collaborative filtering. *ACM Transactions on Information Systems* 22(1):89–115.

Kaminskas, M., and Bridge, D. 2016. Diversity, serendipity, novelty, and coverage: A survey and empirical analysis of beyond-accuracy objectives in recommender systems. *ACM Transactions on Interactive Intelligent Systems* 7(1):2.

Kelly, J. P., and Bridge, D. 2006. Enhancing the diversity of conversational collaborative recommendations: a comparison. *Artificial Intelligence Review* 25(1-2):79–95.

Koren, Y., and Bell, R. 2011. Advances in collaborative filtering. In Ricci, F., et al., eds., *Recommender Systems Handbook*. Springer. 145–186.

McNee, S. M.; Riedl, J.; and Konstan, J. A. 2006. Being accurate is not enough: how accuracy metrics have hurt recommender systems. In *Procs. of the CHI'06 extended abstracts on Human Factors in Computing Systems*, 1097–1101.

Pilászy, I.; Zibriczky, D.; and Tikk, D. 2010. Fast als-based matrix factorization for explicit and implicit feedback datasets. In *Procs. of the 4th ACM conference on Recommender Systems*, 71–78.

Santos, R. L.; Macdonald, C.; and Ounis, I. 2010. Exploiting query reformulations for web search result diversification. In *Procs. of the 19th International Conference on World Wide Web*, 881–890.

Smyth, B., and McClave, P. 2001. Similarity vs. diversity. In *Procs. of the International Conference on Case-Based Reasoning*, 347–361. Springer.

Vargas, S.; Castells, P.; and Vallet, D. 2011. Intent-oriented diversity in recommender systems. In *Procs. of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1211–1212.

Vargas, S.; Castells, P.; and Vallet, D. 2012. Explicit relevance models in intent-oriented information retrieval diversification. In *Procs. of the 35th ACM SIGIR International Conference on Research and Development in Information Retrieval*, 75–84.

Vargas Sandoval, S. 2015. *Novelty and Diversity Evaluation and Enhancement in Recommender Systems*. Ph.D. Dissertation, Universidad Autónoma de Madrid, Spain.

Wasilewski, J., and Hurley, N. 2016. Intent-Aware Diversification Using a Constrained PLSA. In *Procs. of the 10th ACM Conference on Recommender Systems*, 39–42.

Wasilewski, J., and Hurley, N. 2017. Personalised diversification using intent-aware portfolio. In *Adjunct Publication of the 25th ACM Conference on User Modeling, Adaptation and Personalization*, 71–76.

Ziegler, C.-N.; McNee, S. M.; Konstan, J. A.; and Lausen, G. 2005. Improving recommendation lists through topic diversification. In *Procs. of the 14th International Conference on World Wide Web*, 22–32.