

Adaptive Diversification of Recommendation Results via Latent Factor Portfolio

Yue Shi^a, Xiaoxue Zhao^{b*}, Jun Wang^b, Martha Larson^a, Alan Hanjalic^a
^aDelft University of Technology, Netherlands; ^bUniversity College London, United Kingdom
{x.zhao, j.wang}@cs.ucl.ac.uk, {y.shi, m.a.larson, a.hanjalic}@tudelft.nl

ABSTRACT

This paper studies result diversification in collaborative filtering. We argue that the diversification level in a recommendation list should be adapted to the target users' individual situations and needs. Different users may have different ranges of interests – the preference of a highly focused user might include only few topics, whereas that of the user with broad interests may encompass a wide range of topics. Thus, the recommended items should be diversified according to the interest range of the target user. Such an adaptation is also required due to the fact that the uncertainty of the estimated user preference model may vary significantly between users. To reduce the risk of the recommendation, we should take the difference of the uncertainty into account as well.

In this paper, we study the adaptive diversification problem theoretically. We start with commonly used latent factor models and reformulate them using the mean-variance analysis from the portfolio theory in text retrieval. The resulting Latent Factor Portfolio (LFP) model captures the user's interest range and the uncertainty of the user preference by employing the variance of the learned user latent factors. It is shown that the correlations between items (and thus the item diversity) can be obtained by using the correlations between latent factors (topical diversity), which in return significantly reduce the computation load. Our mathematical derivation also reveals that diversification is necessary, not only for risk-averse system behavior (non-adaptive), but also for the target users' individual situations (adaptive), which are represented by the distribution and the variance of the latent user factors. Our experiments confirm the theoretical insights and show that LFP succeeds in improving latent factor models by adaptively introducing recommendation diversity to fit the individual user's needs.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*Information Filtering*

*The first two authors have equal contribution to this work.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIGIR'12, August 12–16, 2012, Portland, Oregon, USA.
Copyright 2012 ACM 978-1-4503-1472-5/12/08 ...\$10.00.

Keywords

Collaborative filtering, diversity, latent factor model, mean-variance, portfolio theory

1. INTRODUCTION

Collaborative Filtering (CF) is a popular technique to provide users with personalized suggestions of information items, including movies, music, books, news articles to name just a few [1]. An intuition behind CF is that users who had similar preferences in the past are likely to have similar preferences in the future; the more similar they were in the past, the more likely they will agree with each other in the future [26]. Although the underlying mechanism of CF is very different from algorithms used for text retrieval and Web search, recommendation results are often presented to users in the same form as retrieval results, i.e., as a ranked list. The list contains items that have been recommended for a given target user (profile), ranked in order of their predicted preference scores (relevance). Parallel to text retrieval and Web search, diversification of the results list has recently been identified as a critical factor that significantly influences end-user satisfaction with a recommender system [13, 19, 29, 36].

In the past, researchers not only have investigated various means to approach diversification (i.e., to answer the question, “How to diversify?”), but, most importantly, have explored the rationale behind it (i.e., to answer the question, “Why diversify the results?”). In text retrieval, some authors regard diversifying the search results as a way of reducing redundancy and improving information novelty in the results [6, 8], as in work on sub-topic retrieval [35], whereas others consider it as a means of managing uncertainty and risk in the ranked list [7, 32].

However, given the fact that there is a balance between diversification and other criteria such as relevance [6, 8, 32], the third question now emerges, that is, “When to diversify?”. More specifically, do we need to make diversification adaptive to different retrieval/recommendation situations? If yes, what is the appropriate diversification level in each of the situations? A recent study on Web search has found that different queries could benefit from different diversification strategies [23, 24]. In recommendation, it is even more useful to make the diversification adaptive to individual user's tastes. The usefulness of adaptive diversification has two aspects: First, user tastes have different scope and coverage of the underlying topics/factors, indicated by the rated items. Some users' tastes are more specific to a few topical areas, while others are more diversified across various top-

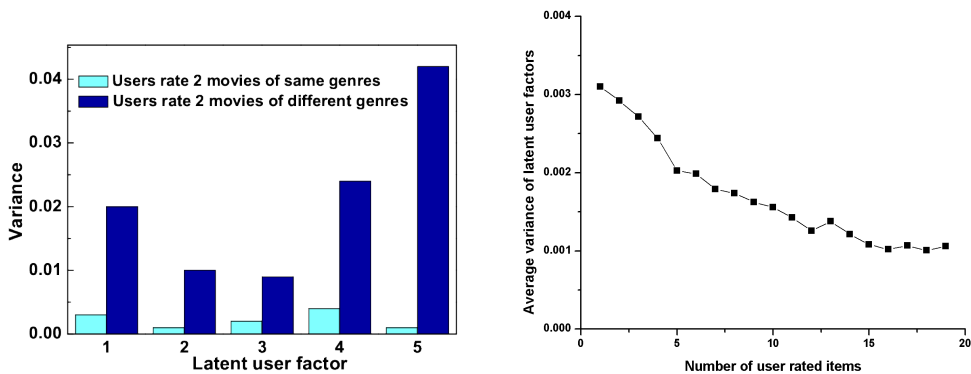


Figure 1: PureSVD (with latent dimensionality 5) [4] on the MovieLens 1M dataset [12]. Left: The variance of each latent user factor for users who rated 2 movies with same or different genres. Right: The relationship between the average variance of latent user factors against the number of movies that the user rated.

ics. To see this, consider the following example in which two movies favored by a user are chosen as a sample that serves as a reflection of the user’s overall movie tastes. Suppose a user favors two movies, “Underworld” and “Twilight”. For that user, we may provide a recommendation list containing less diversified items as it is likely that the user’s taste is more concentrated on a few specific topical areas (likely to prefer Fantasy and Thriller kind movies). By contrast, if the user likes “The Social Network” and “Taken”, then a more diversified recommendation would fit the user’s taste better, implied by the fact that the preferred two movies are in quite different genres. These examples suggest that the diversification level should rely on the underlying topic distribution and the specificity of the individual user’s taste. Second, a target user’s “true” taste is hidden and inferred from the rated items of that user (the user profile). Thus, our understanding of the target user’s taste varies and depends on the ambiguity of the provided user profile. If a user just rated a small number of items, which may not provide enough information to infer the user’s exact taste, a more diversified recommendation would be a safer bid.

Fig. 1 further illustrates our intuition by employing a latent factor model on a movie rating data set. The uncertainty of learned user tastes was measured by their variances (the exact definition can be found later in Section 3.1). First, we can see that the variances of the latent user factors are obviously higher for the users who favored two movies with the same genre, than for the users who favored two movies with different genres. Second, we can also see that the variance of latent user factors decreases as the user rates more items¹.

The above examples represent the typical aspects that need to be addressed in order to successfully answer the “when” question. Taking them into account, this paper proposes a new recommendation framework, called Latent Factor Portfolio (LFP), by connecting latent factor models [9, 17, 18, 22, 34] with portfolio retrieval [30]. Latent factor models construct latent user factors/topics from user profile data, and use them to predict the unknown ratings. They have been widely used for CF, due to their accuracy and scalability. Our focus in this paper is, however, not on pre-

dicting user ratings, but on capturing the uncertainty of latent factors and subsequently employing it to infer the diversification level. In the proposed framework, the coverage of a user’s preference is modelled by the distribution of latent factors and the uncertainty is represented by using the variances of latent factors. Our derivation then shows that the distribution and the uncertainty of latent factors in a user profile determine the final diversity of the ranked list. If we want to control the final uncertainty at a certain level, the diversification level should reflect the variance (uncertainty) of latent factors in a user profile that has been given. Note that the novelty of our study does not lie only in the combination of latent factor models and portfolio retrieval, but also, importantly, in the insights and understandings of the adaptive diversification, which otherwise would not be derived from either approach independently.

The paper is organized as follows. We present the detail of the proposed LFP in Section 2, after which we discuss the related previous work in Section 3. Our experimental evaluation is reported in Section 4, and Section 5 summarizes and concludes the paper.

2. LATENT FACTOR PORTFOLIO

2.1 Uncertainty in Latent Factors

In latent factor models, a rating from user u for item i is decomposed as the inner product of the latent user factors and the latent item factors, as expressed by:

$$r_{ui} = \sum_{a=1}^A V_{ia} U_{ua} \quad (1)$$

where V_{ia} denotes the extent to which the i th item is associated with the a th latent factor, and U_{ua} denotes the extent that user u is interested in the a th latent factor. A is the number of latent factors that are employed in the model. Assuming that the representation of items is independent of individual users, and the latent item factors can be learned beforehand, in practice, we can regard the latent item factors as constants. In this paper, our focus is on the uncertainty of the user factors given a user profile, however, the uncertainty of the item factors could be analogously derived. The

¹It should be emphasized that more ratings in a profile may not necessarily give us more information about the user preference. It also depends on what items the user has rated, since some ratings may be more informative than others [11].

expected value of a rating r_{ui} is thus expressed as:

$$E(r_{ui}) = \sum_{a=1}^A V_{ia} E(U_{ua}) \quad (2)$$

We also derive the variance of rating r_{ui} and the covariance between rating r_{ui} and r_{uj} as follows:

$$\begin{aligned} \text{Var}(r_{ui}) &= E[r_{ui} - E(r_{ui})]^2 \\ &= E\left[\sum_{a=1}^A V_{ia} U_{ua} - \sum_{a=1}^A V_{ia} E(U_{ua})\right]^2 \\ &= E\left[\sum_{a=1}^A V_{ia} (U_{ua} - E(U_{ua}))\right]^2 \\ &= E\left[\sum_{a=1}^A \sum_{l=1}^A V_{ia} V_{il} (U_{ua} - E(U_{ua})) (U_{ul} - E(U_{ul}))\right] \\ &= \sum_{a=1}^A V_{ia}^2 \sigma_{ua}^2 \end{aligned} \quad (3)$$

$$\begin{aligned} \text{Cov}(r_{ui}, r_{uj}) &= E[(r_{ui} - E(r_{ui})) (r_{uj} - E(r_{uj}))] \\ &= E\left[\sum_{a=1}^A V_{ia} (U_{ua} - E(U_{ua})) \sum_{a=1}^A V_{ja} (U_{ua} - E(U_{ua}))\right] \\ &= E\left[\sum_{a=1}^A \sum_{l=1}^A V_{ia} V_{jl} (U_{ua} - E(U_{ua})) (U_{ul} - E(U_{ul}))\right] \\ &= \sum_{a=1}^A V_{ia} V_{ja} \sigma_{ua}^2 \end{aligned} \quad (4)$$

where σ_{ua}^2 denotes the variance of the a th latent factor of user u , i.e., $\sigma_{ua}^2 = E[U_{ua} - E(U_{ua})]^2$. The variance of each rating in terms of latent factors represents the uncertainty. Note that in the derivation of Eq. (3) and (4), we make use of the property that the user's interest in different latent factors are uncorrelated, as shown:

$$E[(U_{ua} - E(U_{ua})) (U_{ul} - E(U_{ul}))] = 0, a \neq l \quad (5)$$

This property is a common assumption in latent factor models, in which each latent factor represents one aspect independent of all the others. We have two insights from the above formulation in Eqs. (3) and (4). First, as seen in Eq. (3), the variance (uncertainty) of the user preference score (the rating) is associated with the variance in the latent factors, indicating that taking into account the uncertainty in the latent factors could contribute to modelling the user preference. Second, as seen in Eq. (4), the covariance (proportionate to the correlation) between a user's preferences of two items is also associated with the variance of the latent factors, indicating that it is feasible to exploit the uncertainty of latent factors to regulate the recommended items in order to satisfy the user's demand on the diversity and the coverage of recommended items.

Ideally, the variance of a latent user factor, e.g., σ_{ua}^2 , is estimated from a number of observations of U_{ua} , which means we need to sample the rated items from user u multiple times. However, this estimation could be infeasible in practice, since 1) multiple observations of user profiles are typically unavailable, thus, requiring heuristic sampling strategy, 2) it requires training the model multiple times

ALGORITHM 1: Latent Factor Portfolio

Input: a given user u and its latent factors \mathbf{U}_u , latent item factors \mathbf{V} , weighting coefficients $w_n, n = 1, 2, \dots, N$, and parameter b

Output: A local optimal ranked item list for user u .

$C = \{1, 2, \dots, N\}$;

$S = \emptyset$;

Select the item with index idx , i.e.,

$idx = \arg \max_{c(n), n=1, 2, \dots, N} \sum_{a=1}^A U_{ua} V_{c(n)a}$ to be ranked in the first

position;

$S = S \cup \{idx\}$;

$k = 2$;

repeat

for each $x \in C \& x \notin S$ **do**

 Compute $\Delta F(x)$ according to Eq. (11);

end

$c(k) = \arg \max_x \Delta F(x)$;

$S = S \cup \{c(k)\}$;

$k = k + 1$;

until $k > N$;

according to different observations of user profiles, thus, inflating the computational cost. For this reason, we propose an approximation for the variance of each latent user factor, based on the latent factors of the items that have been rated by the user, as shown below:

$$\sigma_{ua}^2 = \frac{1}{|N_u|} \sum_{i \in N_u} (U_{ua} - V_{ia})^2 \quad (6)$$

where N_u represents the set of items rated by user u , and $|N_u|$ denotes its cardinality. This approximation satisfies our basic assumptions about the properties of uncertainty as follows. In the case that a user prefers two or more similar items, i.e., the items could be expected to be represented by similar latent factors, the estimated variance with respect to those latent factors could be low. Conversely, if two rated items are quite different, i.e., the items could be expected to be represented by far different latent factors, the corresponding variance of the latent user factors could be high.

2.2 Latent Factor Portfolio Ranking

Taking into account the ranking positions of the items, we can express the overall relevance of a recommendation list based on latent factors as below:

$$R_{uN} = \sum_{n=1}^N w_n r_{uc(n)} = \sum_{n=1}^N w_n \sum_{a=1}^A V_{c(n)a} U_{ua} \quad (7)$$

in which w_n is a weighting coefficient for rank position n and it is a monotonically decreasing function of the ranking position. The most common function for w_n is $w_n = 1/2^{n-1}$ [15], which is also used in this paper. Here, we introduce a rank function $c(n)$ that returns the item index of the n th item in the ranking list. R_{uN} denotes the overall relevance of N recommended items for user u .

2.2.1 From Factor Level to Rank Level

Taking into account Eqs. (2)- (4) and (7), we obtain the expected value of the relevance of the ranked list as:

$$E(R_{uN}) = \sum_{n=1}^N w_n \sum_{a=1}^A V_{c(n)a} E(U_{ua}) \quad (8)$$

and the variance of the ranked list as:

$$\begin{aligned} \text{Var}(R_{uN}) &= \sum_{n=1}^N w_n^2 \sum_{a=1}^A V_{c(n)a}^2 \sigma_{ua}^2 \\ &+ \sum_{n=1}^N \sum_{\substack{m=1 \\ m \neq n}}^N w_n w_m \sum_{a=1}^A V_{c(n)a} V_{c(m)a} \sigma_{ua}^2 \end{aligned} \quad (9)$$

where, for the readability, we skip the detailed derivation from the topic variance to the rank list variance, and leave it to Appendix A. Note that the uncertainty of the recommendation list is represented by the variance in terms of latent factors. We have two insights from the two terms of Eq. (9). The first term reflects that the variance of a recommendation list is also top-biased. In other words, if the variance of a latent user factor, e.g., σ_{ua}^2 , is given, then the latent factor of top-ranked items would have larger influence on the uncertainty of the recommendation list than the low-ranked items. In this sense, in order to reduce the uncertainty of the recommendation list, we need to rank the item relatively higher if its latent factor, e.g., $V_{c(n)a}$, is large and the variance of the corresponding latent user factor, i.e., σ_{ua}^2 , is low. The second term indicates that the relative rank positions of any two items in the recommendation list influence the overall uncertainty. For example, if the variance of a latent user factor is large and the user has shown interest in an item whose corresponding latent factor is also large, then ranking another item with also a large corresponding latent factor at the higher position leads to the larger uncertainty. Conversely, if the variance of a latent user factor is small, then ranking the two items higher would not lead to large increase of the overall uncertainty. Note that the variances of all the latent user factors need to be taken into account for an aggregated impact on the overall uncertainty. Summarizing, it is evident that by exploiting the uncertainty of the latent factors, recommendation diversification can be attained adaptively.

2.2.2 Sequential Ranking

Following the portfolio theory of IR [32], we can attain an optimal recommendation list by taking into account the tradeoff between the mean relevance of the recommended items and the corresponding variance. As a result, the objective function is expressed as:

$$F(R_{uN}) = E(R_{uN}) - b \text{Var}(R_{uN}) \quad (10)$$

in which b is a risk-reward tradeoff parameter. As we shall see later, parameter b is a system level parameter and does not contribute to the user adaptive adjustment of the diversification level. Instead, the diversification level in the ranked list will automatically be adjusted according to the uncertainty of the user factors and their distribution (in other words, rely on how much we understand the target user from the provided ratings). By maximizing this objective function, an optimal ranking can be achieved, which attains an optimal mean-variance balance. In this paper, we adopt the sequential ranking algorithm as used in [30] to solve the optimization problem in Eq. 10. Again, for readability, we leave the exact derivation in Appendix B and give the final

item ranking rule at rank k as:

$$\begin{aligned} c^*(k) &= \arg \max_{c(k)} \{\Delta F(R_{uk})\} = \arg \max_{c(k)} \{F(R_{uk}) - F(R_{uk-1})\} \\ &= \arg \max_{c(k)} \left\{ \sum_{a=1}^A \left(V_{c(k)a} U_{ua} - b w_k \sigma_{ua}^2 V_{c(k)a}^2 \right. \right. \\ &\quad \left. \left. - 2b \sigma_{ua}^2 \sum_{m=1}^{k-1} w_m V_{c(k)a} V_{c(m)a} \right) \right\} \end{aligned} \quad (11)$$

where we have dropped w_k from Appendix B since it is a constant for rank k . We call the above formulation *latent factor portfolio ranking*, since both the mean and the variance are defined on latent factors of users and items, as reflected in the summation over the factor space a . The most important characteristic in LFP is the introduction of the variances of the latent factors σ_{ua}^2 , which introduces the adaptation. Combining σ_{ua}^2 and b , topic diversity in the ranked list is adjusted in two levels. At the *system level*, the need for diversification is due to the risk-averse behaviour of the system and it is adjusted at parameter b . As shown in [33], the risk-averse behaviour is query (user profile)-independent and related to the utility of the system, defined by the used IR metric. At the *user profile level*, the need for diversification is related to the level of absolute certainty about the latent topics that the target user is interested in. The uncertainty is represented by the variances σ_{ua}^2 of the latent factors in the formula. Combined with b , it adaptively balances the mean and reward tradeoff in the user profile level, thus making the topic diversification adapted to each individual users' need.

In the next section, we position the proposed LFP with respect to related work, and discuss its contribution.

3. RELATED WORK AND DISCUSSION

Latent factor models have become a dominating branch among CF approaches, due to their superiority in terms of accuracy and scalability, as shown in Netflix competition [17]. Matrix Factorization (MF) forms a group of the most well-known latent factor models, e.g., Singular Value Decomposition (SVD) [18], SVD++ [16]. Probabilistic Matrix Factorization (PMF) was proposed to carry out the rating factorization from a probabilistic view [22], which leads to the most widely used regularized L2-norm regression model. Additionally, logistic regression was also proposed to learn latent factors [2]. The latest developments in recommender systems have explored different criteria to improve user satisfaction based on latent factor models, such as modelling user choice process [34] and the marginal net utility [31].

From Eq. (11), we observe that: on one hand, compared to the latent factor models (e.g., [17] and [22]), LFP ranks an item at position k based on not only its rating predictions, i.e., the first term in Eq. (11), but also its uncertainty in terms of the latent item factors, i.e., the second term, and the correlation between this item and the items ranked before it, i.e., the third term. Therefore, we can regard LFP as a general extension for latent factor models that introduces the tradeoff with respect to the uncertainty of recommended items. Note that in our derivation we only consider the uncertainty from user latent factors. One can also consider the randomness of both the user factors and items factor simultaneously, but the study is worth a full attention and is

beyond the scope of this paper. We thus leave the detailed discussion and research for future work.

Diversity is realized as one of the most important aspects for the recommendation quality [13, 19, 29, 36]. It is the key factor to help users to explore new interests that they might not discover by themselves and thus enhance user experience. Ziegler et al. proposed a re-ranking algorithm to bring topic diversification, which balances the ranking list according to user’s complete spectrum of interests [36]. According to their studies, diversity of a recommendation list may hamper precision to some degree, but will improve user satisfaction as a whole. Then, Zhang and Hunley formalized the intra-list topic diversification problem by addressing a multi-objective optimization problem on diversity and preference similarity [13]. On the other hand, Lathia et al. considered the recommender system as a temporally evolving system that gives diversified recommendations over time [19]. They provided a hybrid algorithm that can offer dynamic recommendations, and they also discovered the negative correlation between user profile length and the degree of recommendation diversity. The issue of evaluating the novelty and the diversity of recommendations has also been raised [29]. In addition, we note that diversifying search results has been extensively studied in the IR community [6, 8, 10, 20, 23, 24, 25], resulting in a fruitful set of diversification methods.

It is also of interest to specifically compare the proposed LFP as in Eq. (11) with other adaptive diversification methods recently proposed in text retrieval [23, 24]. In [23], the diversification trade-off of an unseen query was obtained by mapping it to the known queries whose optimal diversification level is known a priori. By contrast, our method is fully unsupervised and the diversification level is naturally adapted to the latent topics that the target user is interested in and also how many of them we have already obtained in the ranked list. As shown by the first term of Eq. (11), an item is promoted if it has the same topic as the user’s. However its rank score will be penalized if the same topic has already appeared in the lower ranks (see the product $V_{c(k)a}V_{c(m)a}$ in the third term). In [24], an intent-aware search result diversification method was proposed. This study was focused on the first term in our formula. In their approach, query aspect intents are classified into two categories (factors): informational and navigational, and a machine learning algorithm is used to rank documents with respect to the categories.

The other branch of research related to our work exploits portfolio theory for various information retrieval and recommendation tasks. The importance of such approaches has recently been underlined in a talk by Resnick [21], who projects the usefulness and the necessity of portfolio theory in personalized systems. The application of portfolio theory in information retrieval and recommendation was first proposed by Wang et al. [30, 32]. Recent increasing attention to exploiting principles in economics for IR [3] may also fall under the same direction.

In the original portfolio retrieval formulation, the uncertainty about the overall relevance of a ranked list is linked to the co-variances between individual documents (relevancies) [30, 32]. However, as they are conditioned on a given query or user profile, exactly, how to obtain such a co-variance matrix remains an open question. In practice, the covariance between two relevance scores is approximated by the covariance with respect to their document term occurrences

or user ratings. Computationally, this approach is expensive because every document or item pair needs to be considered. In this paper, we solve this issue by providing a better explanation of the correlation: document or items are correlated because of their underlying topics and latent factors. As shown in Eq. (11), LFP ranks items by taking into account item correlations based on their latent factors/topics, i.e., the products between item factors, which are not exploited in the original model. For example, when ranking a movie in position k , LFP (in the case of $b > 0$) would prefer a movie with a genre (assumed to be represented by a latent factor) that was not contained by the movies ranked before position k , in order to maximize Eq. (11). In this sense, we can regard LFP as a general extension for the original retrieval model, where when $A = 1$, LFP returns to the original retrieval model in [32].

4. EXPERIMENTAL EVALUATION

In this section we present a series of experiments to evaluate the proposed adaptive diversification method. We specifically focus on the following aspects: 1) As discussed, user tastes have different scope and coverage, reflected by their rated items: some are more specific, while others have wider interests. The question is whether our method is able to adapt the diversification level to the taste of each user. 2) The number of rated items provided by users varies. As a result, we have different accuracy and uncertainty about the users’ “true” taste. We intend to investigate whether our method could adjust the diversification level of the ranked list to the uncertainty. 3) If we consider the overall recommendation quality is an aggregated effect from both the relevance and the diversity, whether LFP could benefit for improving the overall recommendation quality?

4.1 Experimental Framework

4.1.1 Dataset

The publicly available dataset MovieLens-1M is used in our experimental evaluation. The dataset contains 1M ratings (scale 1-5, 5 for the best, and 1 for the worst) from about 6K users on about 3.7K movies/items. The data sparseness is 95.5%. Each user in the dataset has at least 20 ratings. In addition, the genre information of movies is provided. There are in total 18 genres, and each movie can be associated with multiple genres. The average number of genres per movies is 1.62. Note that our focus in this paper is not on the performance comparison against the state-of-the-art baselines, but on investigating how the proposed method could diversify recommendation results under different conditions of user profiles (tastes). The choice of a moderate size dataset enables an efficient exploration of experimental results under various settings.

In order to create the training set (for estimating the latent factor models) and the test set (for evaluation), we split the data into a large set containing 80% of the users and a small set the remaining 20%. For each user profile length (UPL) that we investigate in the experiments, we create a test set containing users with that UPL, by randomly selecting the desired number of items (i.e., UPL=2,3,5 and 10) from each user in the small set. The remaining items from each profile in the small set are added to the complete large set to form the training set.

4.1.2 Evaluation Metrics

In our experiments, we adopt a common metric in text retrieval, Mean Average Precision (MAP), to measure the effectiveness of the ranked recommendation list. In order to calculate MAP, we set the relevance threshold as rating 4. In other words, we regard items with ratings equal to or larger than 4 as relevant.

For the investigation of the trade-off (and combination) of the relevance and the diversity in Section 4.2.4, we utilize another evaluation metric from text retrieval, α NDCG [8]. We use movie genres as “nuggets” in calculating α NDCG. The exact definition of α NDCG is expressed below:

$$\alpha NDCG@K = \frac{\alpha DCG@K}{\alpha IDCG@K} \quad (12)$$

in which,

$$\alpha DCG@K = \sum_{k=1}^K \frac{\sum_{l=1}^L J_{kl}^u (1 - \alpha)^{q_{l,k-1}^u}}{\log_2(1 + k)} \quad (13)$$

J_{kl}^u is an indicator function that is equal to $r_{uc(k)}$ (i.e., the rating of the k th movie in the list for user u), if the k th movie in the recommendation list of user u contains genre l , otherwise 0. $q_{l,k-1}^u$ denotes the number of movies ranked up to position $k-1$ that contain genre l in the recommendation list for user u . α is a constant set to control the magnitude of penalty for the redundancy of the recommended items. The value of α can be within the range $[0, 1]$, in which the higher value indicates the larger penalty. In our experiments, we use $\alpha = 0.5$ as a moderate choice for measuring diversity. $\alpha IDCG@K$ denotes the highest possible value of $\alpha DCG@K$ in the case that the top K recommendation list contains “ideally” diversified relevant items. Thus, $\alpha NDCG$ is normalized to be $[0, 1]$. Note that $\alpha NDCG$ depends on both the movie ratings and genres, representing a suitable metric for our purpose of evaluating the trade-off between the relevance and the diversity. Since we particularly focus on the top-ranked items in recommender systems, we use $K = 5$ in the experiments.

To solely measure the recommendation diversity in a ranked list, we also introduce a simple diversity measure called DNG@K. It measures the number of genres in the top-K ranked list. The number is discounted according to the position of the corresponding movie in order to consider the rank bias. Specifically, we define DNG@K as:

$$DNG@K = \sum_{k=1}^K w_k G(k) \quad (14)$$

in which $G(k)$ denotes the number of genres that the k th movie have and that are not included in the top $k-1$ movies. w_k is a discount factor that is set as $w_k = 1/2^{k-1}$. Similar to $\alpha NDCG$, we focus our evaluation with $K = 5$. The reported DNG is an average across all the test users. Note that DNG is different from other proposed diversity measures that take into account the relevance of recommended items, such as the work in [29].

4.1.3 Latent Factor Models

In collaborative filtering, there are various ways of obtaining latent factors in Eq. (1), either by non-probabilistic approaches [16, 17, 18] or from a probabilistic viewpoint [22].

In our experiment, we choose one representative from each of the two categories, briefly described as follows:

PureSVD: It is the basic form of singular value decomposition [4], as show in Eq. (15). Supposing the user-item rating matrix R consists of M users and K items, the rating matrix R can be decomposed into three low-rank matrices, $P^{M \times A}$, $S^{A \times A}$, and $Q^{K \times A}$. Then, the latent factors of users can be denoted as \mathbf{U} , and the latent factors of items can be denoted as \mathbf{V} , as shown in Eq. (16). Each column vector in \mathbf{U} (e.g., \mathbf{U}_u for user u) or \mathbf{V} (e.g., \mathbf{V}_i for item i) represents the corresponding user or item. Although PureSVD features the most basic latent factor model, the recommendation performance for top-N tasks is competitive according to a recent empirical study [9]. For this reason, we choose it as a representative of non-probabilistic latent factor models.

$$\mathbf{R} = \mathbf{P}\mathbf{S}\mathbf{Q}^T \quad (15)$$

$$\mathbf{U} = (\mathbf{P}\mathbf{S}^{1/2})^T, \mathbf{V} = \mathbf{S}^{1/2}\mathbf{Q}^T \quad (16)$$

PMF: We use probabilistic matrix factorization [22] (PMF) as one of the state-of-the-art latent factor models in CF. PMF estimates the latent factors of users and items by maximizing the posterior, which is the conditional probability of the latent factors given the observed ratings R and hyper-parameters Θ , as shown below:

$$\mathbf{U}, \mathbf{V} = \arg \max_{\mathbf{U}, \mathbf{V}} p(\mathbf{U}, \mathbf{V} | R, \Theta) \quad (17)$$

The resulting objective function of PMF is expressed as:

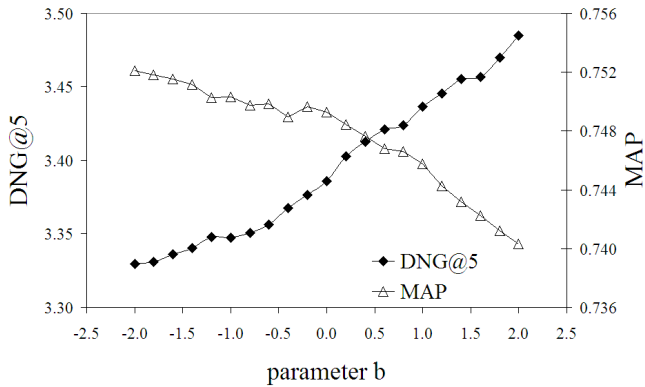
$$\mathbf{U}, \mathbf{V} = \arg \min_{\mathbf{U}, \mathbf{V}} \frac{1}{2} \sum_{u=1}^M \sum_{j=1}^K I_{uj} \left(r_{uj} - g(\mathbf{U}_u^T \mathbf{V}_j) \right)^2 + \frac{\lambda_U}{2} \|\mathbf{U}\|_F^2 + \frac{\lambda_V}{2} \|\mathbf{V}\|_F^2 \quad (18)$$

The latent factors of users and items are learned from the user-item ratings (normalized to $[0, 1]$), and the magnitudes of latent factors are penalized in order to alleviate overfitting. $g(x)$ is a logistic function, i.e., $g(x) = 1/(1 + \exp(-x))$, that serves to bound the range of the inner product of latent factors. A simplification is usually made to set $\lambda = \lambda_U = \lambda_V$, which is also used in this paper.

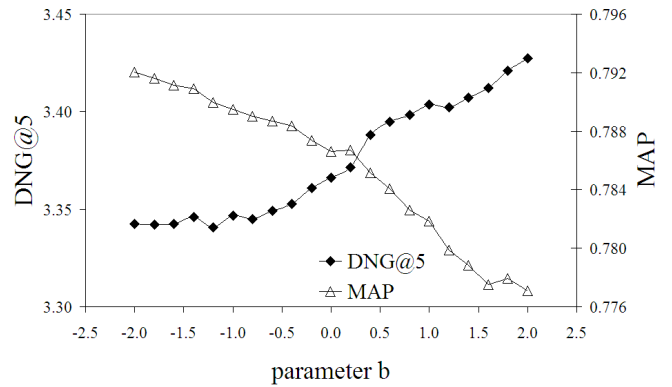
4.2 Results and Analysis

4.2.1 System-level Diversity

As discussed, our LFP model implies that the need for diversification in a ranked list comes from two levels. Our first experiment is to investigate the system level diversity, which is controlled by the parameter b in Eq. (11). In our experiment, we use the training set to train the latent factor models, and for each user in the test set, we randomly select 2 rated items, i.e., User Profile Length (UPL=2), as user profiles, and use the remaining rated items as ground truth. By varying the value of parameter b in LFP, we evaluate its influence on the recommendation performance of different latent factor models, which is shown in Fig. 2. As can be seen, for both PureSVD and PMF, the diversity measure DNG@5 generally increases as the value of b in LFP increases, while MAP decreases. Note the baseline latent factor models are equivalent to the case that we set $b = 0$. The figures show that a positive value of b could contribute to diversifying the recommendation results, and the magnitude of diversification is controlled by its value. However, a

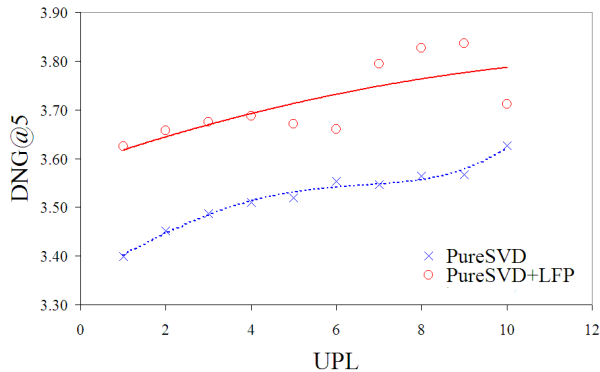


(a) LFP for PureSVD

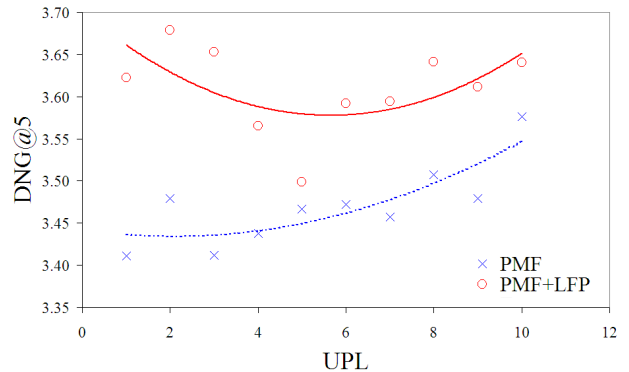


(b) LFP for PMF

Figure 2: The system level diversity: the impact of parameter b on DNG@5 and MAP.



(a) LFP for PureSVD



(b) LFP for PMF

Figure 3: The diversity that depends on the target user profiles: the number of rated items.

Table 1: The diversity DNG@5 adapted to the target user profiles: the range of interests.

	UPL=2		UPL=3	
	Focused	Broad	Focused	Broad
PureSVD	3.148	3.419	3.174	3.386
PureSVD+LFP	3.293	3.641	3.304	3.608
p-value	0.096	0.000*	0.026	0.000*
PMF	3.353	3.400	3.373	3.432
PMF+LFP	3.415	3.494	3.449	3.570
p-value	0.335	0.006*	0.411	0.002*

positive value of b could reduce the MAP, indicating that it may degrade the end-user satisfaction when the results are over-diversified. The observation is consistent to the study in text retrieval in [33]. Because the parameter is a constant across target users, it serves to adjust the diversity of recommendation at the system level, and its optimal value depends on the evaluation metrics used (in other words, the utility of the recommendation system).

4.2.2 Adaptive diversity: the num. of rated items

We have discussed in Section 1 that the observed numbers of rated items are different across users. As illustrated in Fig. 1, the number of user rated items influences the uncertainty of the learned latent user factors—the more information we have about the user, the less uncertain our model is about the user’s hidden tastes. In the following experiment, we evaluate the impact of the model uncertainty on

the diversity of recommended items, where the model uncertainty is indicated by the number of rated items provided in the user profile. We generate the user profile length (UPL) from 1 to 10, and randomly select the rated items as user profile items. As in our dataset, each user has at least rated 20 items. Setting UPL up to 10, we ensure that there are at least 10 rated items per user used for testing.

From Fig. 3, we observe that the LFP models succeed in consistently increasing the diversity of recommendation results for both PureSVD and PMF. Note that the increases are all statistically significant, according to Wilcoxon signed rank significance test with $p < 0.01$. This indicates that LFP could effectively capture the uncertainty of latent factors and use it to diversify recommendation results.

In addition, the diversity achieved by both our LFP and each of the basic latent factor models, PureSVD and PMF, generally increases as the users rated more items. At a first glance, the result seems to contradict with the idea that adding ratings in the user profile would reduce the uncertainty of the user model and thus the need for diversifying the results. A closer look, however, suggests that this is intuitively correct because when there are few rated items known from the users, the recommended items could be strongly biased toward the few known items, and thus less diversified, whereas when more rated items are known from the users, the recommended items could be more likely to cover different aspects of user interest, and are thus more diversified.

Table 2: Example recommendation results for five movies using the two different types of user profiles. We refer “Ac” to Action, “Ad” to Adventure, “C” to Comedy, “D” to Drama, “H” to Horror, “R” to Romance, “SF” to Sci-Fi, “T” to Thriller, and “W” to War. LFP with PureSVD is used.

Type	User with <i>Focused</i> interest		User with <i>Broad</i> interests	
Profile	Chariots of Fire (D) Erin Brockovich (D)		American Pie (C) The Blair Witch Project (H)	
Rank	PureSVD	PureSVD+LFP	PureSVD	PureSVD+LFP
1	Second Best(D)	Second Best(D)	Big Daddy(C)	Big Daddy(C)
2	Saving Private Ryan(Ac,D,W)	North by Northwest(D,T)	Bowfinger(C)	The Mask of Zorro(Ac,Ad,R)
3	North by Northwest(D,T)	The Truman Show(D)	Parasite(H,SF)	Baby Geniuses(C)
4	The Truman Show(D)	Saving Private Ryan(Ac,D,W)	Baby Geniuses(C)	Parasite(H,SF)
5	Jakob the Liar(D)	Jakob the Liar(D)	The Mask of Zorro(Ac,Ad,R)	Bowfinger(C)
DNG@5	2.25	1.75	1.69	2.75

Finally and most importantly, we also observe that the increase of diversity introduced by LFP generally decreases as the number of user rated items increases. As shown, the diversity increase brought by LFP with PureSVD tends to be constant as the UPL increases, and the diversity achieved by using LFP with PMF tends to be closer to that of using only PMF as the UPL increases. When the UPL is small, LFP automatically provides relatively larger increase of diversity against the basic latent factor models. In other words, when the users rated only a few items, the basic latent factor models tend to recommend items based on highly uncertain latent factors. LFP addresses the risk of the basic latent factor models by providing more diversified results.

4.2.3 Adaptive diversity: the range of interests

We now focus on the evaluation by considering the users with different ranges of interests. To make our study focused and controlled, we are particularly interested in two types of users, i.e., the users who rated movies with the same genre (denoted as the “Focused” type), and the users who rated movies with the non-overlapped genres (denoted as the “Broad” type). The first type of user profiles represents a typical situation in which the target user has a specific range of interests and as a result, the diversification is less required, while the second type represents the opposite situation in which diversification is more desired. Also as demonstrated in the previous section, LFP could be most beneficial for diversifying recommendation results for the users who only rated a limited number of items. For this reason, we use UPL 2 and 3 in this investigation. For each UPL, we classify a user into the “Focused” type if all his or her rated items contain the same genre, and into the “Broad” type if all rated items are associated with genres different from each other.

The results are shown in Table 1, from which we have two observations. First, for both PureSVD and PMF, the diversity of the “Focused” type is lower than the “Broad” type for most of the cases. This result is in accordance with our understanding, as discussed in Section 1, that the commonality of the items in the user profile has a significant impact on the user need for diversification. In the case of the “Focused” type of user profiles, the latent user factors are learned from the items that have the same or similar topics (in this case genres), and the latent factors of those movies could be similar. As a result, the uncertainty and variance of the latent user factors could be low, and those latent user factors would promote to recommend movies with the same or similar genres as the movies that the user has already watched, i.e., a less diversified recommendation. By contrast, a more diversified recommendation would be promoted to the “Broad” type of user profiles.

Second, we observe that for both PureSVD and PMF, and

Table 3: Relevance vs. diversity (with PureSVD).

	MAP	DNG@5	α NDCG@5
UPL=2			
PureSVD	0.749	3.483	0.845
PureSVD+LFP ($b = -1$)	0.751	3.341	0.833
PureSVD+LFP ($b = 1$)	0.738	3.645	0.858
UPL=5			
PureSVD	0.764	3.533	0.854
PureSVD+LFP ($b = -1$)	0.772	3.417	0.841
PureSVD+LFP ($b = 1$)	0.741	3.683	0.866
UPL=10			
PureSVD	0.769	3.555	0.857
PureSVD+LFP ($b = -1$)	0.774	3.456	0.845
PureSVD+LFP ($b = 1$)	0.745	3.706	0.870

Table 4: Relevance vs. diversity (with PMF).

	MAP	DNG@5	α NDCG@5
UPL=2			
PMF	0.787	3.412	0.846
PMF+LFP ($b = -1$)	0.791	3.361	0.839
PMF+LFP ($b = 1$)	0.758	3.500	0.864
UPL=5			
PMF	0.807	3.515	0.863
PMF+LFP ($b = -1$)	0.814	3.415	0.851
PMF+LFP ($b = 1$)	0.763	3.611	0.874
UPL=10			
PMF	0.818	3.538	0.865
PMF+LFP ($b = -1$)	0.826	3.462	0.855
PMF+LFP ($b = 1$)	0.774	3.610	0.872

for both UPL=2 and UPL=3, LFP has a significant increase of diversity for the “Broad” type of user profiles, while introducing a slight change of diversity for the “Focused” type. The result indicates that LFP could effectively exploit the distribution of latent user factors to adaptively determine the level of diversification. This is further illustrated by the example in Table 2. We clearly see that LFP automatically adjusts the diversity of recommendations according to the different range of interests learned from the user profiles.

4.2.4 Combining Relevance and Diversity

Our final experiment investigates how LFP can benefit the end-user satisfaction by considering both the relevance and diversity of recommended items. We test the recommendation performance in terms of relevance, as measured by MAP, and the performance in terms of diversity, as measured by DNG@5, under two different settings of parameter b in LFP, i.e., $b = -1$ and 1. Note that as shown in Section 4.2.1, a positive value of b tends to increase the recommendation diversity, while decreasing the recommendation relevance. The opposite results can be observed in the case of a negative value of b . The results are shown in Table 3 and 4. We first observe that LFP could improve the relevance of recommendations for the users who have broad interests. When $b = -1$, LFP improves MAP

for both PureSVD and PMF. This result indicates that in the case that LFP increases the similarity among the recommended items based on latent item factors, it could contribute to improving the relevance of the recommendation. The empirical result is also consistent with the statistical mean-variance analysis of MAP conducted in [33]. Second, LFP could achieve a trade-off between the recommendation relevance and the diversity. As can be seen, when $b = 1$, LFP diversifies the recommendation results and results in an improved DNG@5 for both PureSVD and PMF, while degrading the relevance performance as measured by MAP. However, on the whole, α NDCG is substantially improved, indicating that the degraded relevance is compensated by a payoff in the overall quality of recommendation when both relevance and diversity are taken into account. Although here α NDCG could only serve as an approximation of the end-user satisfaction for the recommended items, the results are evident that we can use LFP to adjust the trade-off between the relevance and the diversity of recommendations from latent factor models, allowing us to give a positive answer to our final research question.

5. CONCLUSION AND FUTURE WORK

We proposed a new recommendation framework, called Latent Factor Portfolio (LFP), specifically for adaptively diversifying recommendation results for individual users. We exploited the variance of the latent user factors to capture the range of user interests and uncertainty of the user profiles and use them as the basis for indicating users' needs for diversity. Through our experiments, we demonstrated the effectiveness of LFP for adapting result diversification to the users' needs without accessing to explicit item properties. In addition, we also showed that LFP is capable of effectively adjusting the trade-off between the relevance and the diversity of recommended items, and thus could further contribute to the overall recommendation quality.

Our future work involves several possible directions. First, we are interested in alternative optimization algorithms to improve the portfolio selection process. Second, we will extend the basic latent factor models and explore various ways of estimating the variances of latent factors of both users and items, possible through Bayesian approaches [5]. Third, we are also interested in investigating the possibility to develop dynamic LFP that could instantly modify recommendations through the interactions with users [14, 27]. Finally, we would like to extend our work to adaptive search result diversification in text retrieval [28] and compare our work with other diversification approaches.

6. REFERENCES

- [1] G. Adomavicius and A. Tuzhilin. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749, 2005.
- [2] D. Agarwal and B.-C. Chen. Regression-based latent factor models. KDD '09, pages 19–28, 2009.
- [3] L. Azzopardi. The economics in interactive information retrieval. SIGIR '11, pages 15–24, 2011.
- [4] R. Bambini, P. Cremonesi, and R. Turrin. *Recommender Systems Handbook*, chapter Recommender Systems for a IPTV Service Provider: A Real Production Environment. Springer, 2010.
- [5] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 2003.
- [6] J. Carbonell and J. Goldstein. The use of MMR diversity-based reranking for reordering documents and producing summaries. SIGIR '98, pages 335–336, 1998.
- [7] H. Chen and D. R. Karger. Less is more: probabilistic models for retrieving fewer relevant documents. SIGIR '06, 2006.
- [8] C. L. Clarke, M. Kolla, G. V. Cormack, O. Vechtomova, A. Ashkan, S. Büttcher, and I. MacKinnon. Novelty and diversity in information retrieval evaluation. SIGIR '08, pages 659–666, 2008.
- [9] P. Cremonesi, Y. Koren, and R. Turrin. Performance of recommender algorithms on top-N recommendation tasks. RecSys '10, pages 39–46, 2010.
- [10] S. Guo and S. Sanner. Probabilistic latent maximal marginal relevance. SIGIR '10, pages 833–834, 2010.
- [11] A. S. Harpale and Y. Yang. Personalized active learning for collaborative filtering. SIGIR '08, pages 91–98, 2008.
- [12] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. SIGIR '99, pages 230–237, 1999.
- [13] N. Hurley and M. Zhang. Novelty and diversity in top-N recommendation – analysis and evaluation. *ACM Trans. Internet Technol.*, 10:14:1–14:30, March 2011.
- [14] T. Jambor, J. Wang, and N. Lathia. Using control theory for stable and efficient recommender systems. WWW '12, pages 11–20, 2012.
- [15] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.*, 20:422–446, October 2002.
- [16] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. KDD '08, pages 426–434, 2008.
- [17] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42:30–37, August 2009.
- [18] M. Kurucz, A. A. Benczúr, and K. Csalogány. Methods for large scale svd with missing values. In *Proceedings of KDD Cup and Workshop*, 2007.
- [19] N. Lathia, S. Hailes, L. Capra, and X. Amatriain. Temporal diversity in recommender systems. SIGIR '10, pages 210–217, 2010.
- [20] F. Radlinski and S. Dumais. Improving personalized web search using result diversification. SIGIR '06, pages 691–692, 2006.
- [21] P. Resnick. Personalized filters yes; bubbles no. <http://presnick.livejournal.com/21239.html>, July 2011.
- [22] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. NIPS '08, 2008.
- [23] R. L. Santos, C. Macdonald, and I. Ounis. Selectively diversifying web search results. CIKM '10, pages 1179–1188, 2010.
- [24] R. L. Santos, C. Macdonald, and I. Ounis. Intent-aware search result diversification. SIGIR '11, pages 595–604, 2011.
- [25] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl. Item-based collaborative filtering recommendation algorithms. WWW '01, pages 285–295, 2001.
- [26] U. Shardanand and P. Maes. Social information filtering: algorithms for automating 'word of mouth'. CHI '95, pages 210–217, 1995.
- [27] M. Sloan and J. Wang. Dynamical information retrieval modelling: a portfolio-armed bandit machine approach. WWW '12, pages 603–604, 2012.
- [28] J. Teevan, S. T. Dumais, and E. Horvitz. Potential for personalization. *ACM Trans. Comput.-Hum. Interact.*, 17(1):4:1–4:31, Apr. 2010.
- [29] S. Vargas and P. Castells. Rank and relevance in novelty and diversity metrics for recommender systems. RecSys '11, pages 109–116, 2011.
- [30] J. Wang. Mean-variance analysis: A new document ranking theory in information retrieval. ECIR '09, pages 4–16, 2009.

- [31] J. Wang and Y. Zhang. Utilizing marginal net utility for recommendation in e-commerce. SIGIR '11, pages 1003–1012, 2011.
- [32] J. Wang and J. Zhu. Portfolio theory of information retrieval. SIGIR '09, pages 115–122, 2009.
- [33] J. Wang and J. Zhu. On statistical analysis and optimization of information retrieval effectiveness metrics. SIGIR '10, pages 226–233, 2010.
- [34] S.-H. Yang, B. Long, A. J. Smola, H. Zha, and Z. Zheng. Collaborative competitive filtering: learning recommender using context of user choice. SIGIR '11, pages 295–304, 2011.
- [35] C. X. Zhai, W. W. Cohen, and J. Lafferty. Beyond independent relevance: methods and evaluation metrics for subtopic retrieval. SIGIR '03, pages 10–17, 2003.
- [36] C.-N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen. Improving recommendation lists through topic diversification. WWW '05, pages 22–32, 2005.

APPENDIX

A. TOPICAL VS. RANK VARIANCES

We present the detailed derivation of $Var(R_{uN})$ in Eq. (9) below. Let us start with

$$Var(R_{uN}) = E[R_{uN} - E(R_{uN})]^2.$$

Taking into account Eq. (7), we have:

$$\begin{aligned} & Var(R_{uN}) \\ = & E\left[\sum_{n=1}^N w_n \sum_{a=1}^A V_{c(n)a} U_{ua} - \sum_{n=1}^N w_n \sum_{a=1}^A V_{c(n)a} E(U_{ua})\right]^2 \\ = & E\left[\sum_{n=1}^N w_n \sum_{a=1}^A V_{c(n)a} (U_{ua} - E(U_{ua}))\right]^2 \\ = & E\left[\sum_{n=1}^N w_n^2 \sum_{a=1}^A \sum_{l=1}^A V_{c(n)a} V_{c(n)l} (U_{ua} - E(U_{ua}))(U_{ul} - E(U_{ul}))\right. \\ & + \sum_{n=1}^N \sum_{\substack{m=1 \\ m \neq n}}^N w_n w_m \\ & \left. \times \sum_{a=1}^A \sum_{l=1}^A V_{c(n)a} V_{c(m)l} (U_{ua} - E(U_{ua}))(U_{ul} - E(U_{ul}))\right]. \end{aligned}$$

Using the property as in Eq. (5), we obtain:

$$\begin{aligned} & Var(R_{uN}) \\ = & \sum_{n=1}^N w_n^2 \sum_{a=1}^A V_{c(n)a}^2 E[(U_{ua} - E(U_{ua}))^2] \\ & + \sum_{n=1}^N \sum_{\substack{m=1 \\ m \neq n}}^N w_n w_m \sum_{a=1}^A V_{c(n)a} V_{c(m)a} E[(U_{ua} - E(U_{ua}))^2]. \end{aligned}$$

The Eq. (9) is obtained as above with the definition of σ_{ua}^2 .

B. SEQUENTIAL RANKING

The detailed derivation of $\Delta F(R_{uk})$ in Eq. (11) is given below:

$$\Delta F(R_{uk}) = F(R_{uk}) - F(R_{uk-1})$$

$$\begin{aligned} & = \sum_{n=1}^k w_n \sum_{a=1}^A V_{c(n)a} U_{ua} - b \left(\sum_{n=1}^k w_n^2 \sum_{a=1}^A V_{c(n)a}^2 \sigma_{ua}^2 \right. \\ & \quad + \sum_{n=1}^k \sum_{\substack{m=1 \\ m \neq n}}^k w_n w_m \sum_{a=1}^A V_{c(n)a} V_{c(m)a} \sigma_{ua}^2 \left. \right) \\ & - \left(\sum_{n=1}^{k-1} w_n \sum_{a=1}^A V_{c(n)a} U_{ua} - b \left(\sum_{n=1}^{k-1} w_n^2 \sum_{a=1}^A V_{c(n)a}^2 \sigma_{ua}^2 \right. \right. \\ & \quad \left. \left. + \sum_{n=1}^{k-1} \sum_{\substack{m=1 \\ m \neq n}}^{k-1} w_n w_m \sum_{a=1}^A V_{c(n)a} V_{c(m)a} \sigma_{ua}^2 \right) \right) \\ = & w_k \sum_{a=1}^A V_{c(k)a} U_{ua} - b \left(w_k^2 \sum_{a=1}^A V_{c(k)a}^2 \sigma_{ua}^2 \right. \\ & \quad + \sum_{n=1}^k \sum_{\substack{m=1 \\ m \neq n}}^k w_n w_m \sum_{a=1}^A V_{c(n)a} V_{c(m)a} \sigma_{ua}^2 \\ & \quad - \sum_{n=1}^{k-1} \sum_{\substack{m=1 \\ m \neq n}}^{k-1} w_n w_m \sum_{a=1}^A V_{c(n)a} V_{c(m)a} \sigma_{ua}^2 \left. \right) \\ = & w_k \sum_{a=1}^A V_{c(k)a} U_{ua} - b \left(w_k^2 \sum_{a=1}^A V_{c(k)a}^2 \sigma_{ua}^2 \right. \\ & \quad + \sum_{m=1}^{k-1} w_k w_m \sum_{a=1}^A V_{c(k)a} V_{c(m)a} \sigma_{ua}^2 \\ & \quad + \sum_{n=1}^{k-1} \sum_{\substack{m=1 \\ m \neq n}}^k w_n w_m \sum_{a=1}^A V_{c(n)a} V_{c(m)a} \sigma_{ua}^2 \\ & \quad \left. - \sum_{n=1}^{k-1} \sum_{\substack{m=1 \\ m \neq n}}^{k-1} w_n w_m \sum_{a=1}^A V_{c(n)a} V_{c(m)a} \sigma_{ua}^2 \right). \end{aligned}$$

Combining the last two terms results in:

$$\begin{aligned} \Delta F(R_{uk}) = & w_k \sum_{a=1}^A V_{c(k)a} U_{ua} - b \left(w_k^2 \sum_{a=1}^A V_{c(k)a}^2 \sigma_{ua}^2 \right. \\ & + \sum_{m=1}^{k-1} w_k w_m \sum_{a=1}^A V_{c(k)a} V_{c(m)a} \sigma_{ua}^2 \\ & \left. + \sum_{n=1}^{k-1} w_n w_k \sum_{a=1}^A V_{c(n)a} V_{c(k)a} \sigma_{ua}^2 \right). \end{aligned}$$

Note that in above m and n are interchangeable. We, thus, have:

$$\begin{aligned} \Delta F(R_{uk}) = & w_k \sum_{a=1}^A V_{c(k)a} U_{ua} - b \left(w_k^2 \sum_{a=1}^A V_{c(k)a}^2 \sigma_{ua}^2 \right. \\ & \left. + 2 \sum_{m=1}^{k-1} w_k w_m \sum_{a=1}^A V_{c(k)a} V_{c(m)a} \sigma_{ua}^2 \right). \end{aligned}$$

Swapping the summation order over space m and a in the last term, we obtain Eq. (11):

$$\begin{aligned} \Delta F(R_{uk}) = & w_k \sum_{a=1}^A \left(V_{c(k)a} U_{ua} - b w_k \sigma_{ua}^2 V_{c(k)a} \right. \\ & \left. - 2b \sigma_{ua}^2 \sum_{m=1}^{k-1} w_m V_{c(k)a} V_{c(m)a} \right). \end{aligned}$$