

# To Personalize or Not: A Risk Management Perspective

Weinan Zhang, Jun Wang, Bowei Chen, Xiaoxue Zhao  
Department of Computer Science, University College London  
Malet Place, London, WC1E 6BT, UK  
{w.zhang, j.wang, bowei.chen, x.zhao}@cs.ucl.ac.uk

## ABSTRACT

Personalization techniques have been widely adopted in many recommender systems. However, experiments on real-world datasets show that for some users in certain contexts, personalized recommendations do not necessarily perform better than recommendations that rely purely on popularity. Broadly, this can be interpreted by the fact that the parameters of a personalization model are usually estimated from sparse data; the resulting personalized prediction, despite of its low bias, is often volatile. In this paper, we study the problem further by investigating into the ranking of recommendation lists. From a risk management and portfolio retrieval perspective, there is no difference between the popularity-based and the personalized ranking as both of the recommendation outputs can be represented as the trade-off between expected relevance (reward) and associated uncertainty (risk). Through our analysis, we discover common scenarios and provide a technique to predict whether personalization will fail. Besides the theoretical understanding, our experimental results show that the resulting *switch* algorithm, which decides whether or not to personalize, outperforms the mainstream recommendation algorithms.

## Categories and Subject Descriptors

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

## Keywords

Personalization; Collaborative Filtering; Recommender Systems; Portfolio Theory

## 1. INTRODUCTION

Personalization techniques have been widely adopted in many recommender systems. Personalized recommendations are generally derived by distinguishing individuals and their contexts. Example applications such as Amazon [17] and YouTube [7] show personalized recommendations have been well acknowledged and implemented by some of the most successful recommender systems. In particular, Collaborative Filtering (CF) techniques, which make use of the aggregated user preference data to make personalized interest predictions, have proven to be both efficient and effective,

as evidenced by a series of recommendation algorithm competitions [1, 10].

Nevertheless, it is somewhat unclear as to whether personalization will *always* outperform the non-personalization. The fact is, in some cases non-personalized recommendations are able to provide even more relevant (or interesting) items whereas the personalization techniques may fail to catch the users' current personal interest and result in low quality recommendations. The work in [5] found surprisingly good performance from popularity-based recommendations. Furthermore, Figure 1 illustrates a performance comparison between Bayesian personalized ranking (BPR) [22], a state-of-the-art ranking-based CF algorithm, and the popularity-based (non-personalized) algorithm on the MovieLens dataset. It illustrates that, although the personalized algorithm works better overall, on each evaluation measure, there are a large proportion of cases where the popularity-based recommendation outperforms the Bayesian personalized ranking.

From a learning model perspective, the personalized recommendation can reduce the bias but may increase the variance of prediction for user's preference [9]. More importantly, such bias and variance are different across different users and in different recommendation contexts. Therefore, compared with non-personalized recommendation, the overall performance of personalized recommendation is usually higher but it does not guarantee to outperform in all cases. In some research work, regularization and cross validation are adopted for model selection and the bias-variance balance of the model [12]. Thus, it is of great interest to study in what situations the personalization fails and develop a *switch* to judge which model should be adopted to provide recommendations to the target user.

In this paper, we analyze different attributes of personalized and non-personalized recommendations. As a by-product, we provide effective switch algorithms in order to decide whether *to personalize or not* for each user. Specifically, we introduce the concept of *risk* borrowed from finance [15] into the preference prediction model and the analysis of the (non-) personalization. We then leverage modern portfolio theory [11] to optimize the expected reward and risk of the recommendations from both of the personalized and non-personalized algorithms. Finally, a series of switch algorithms are proposed both by heuristics and learning-based methods. As an empirical study, we conduct our experiments on two collaborative filtering benchmark datasets: MovieLens and Netflix. The experimental results support our idea of introducing risk into the preference estimation and our proposed *mean-variance-aware switch algorithms*.

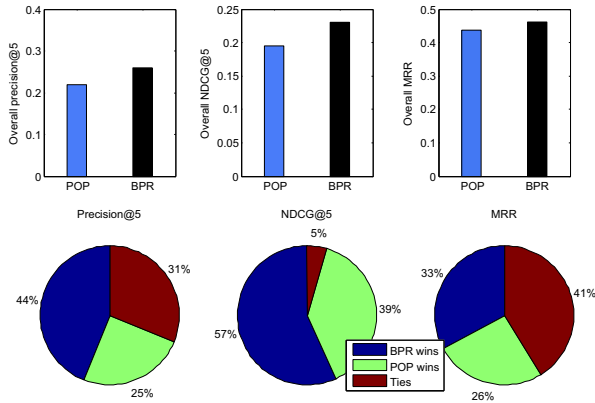
The rest of this paper is organized as follows. Related work is discussed in Section 2. Next we introduce the concepts and methods of risk management in item recommendation, and provide an analysis on whether to personalize

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

RecSys '13, October 12–16, 2013, Hong Kong, China.

Copyright 2013 ACM 978-1-4503-2409-0/13/10 ...\$15.00.

<http://dx.doi.org/10.1145/2507157.2507167>.



**Figure 1: Performance comparison of personalized and non-personalized recommendations.**

or not in a risk management perspective. Motivated by our analysis, we propose our methodology in Section 4. The experiments are described in Section 5, and finally, the paper is concluded in Section 6.

## 2. RELATED WORK

Collaborative Filtering techniques have been widely applied and researched in personalized recommender systems [23, 26]. It refines user-specific profiles by analyzing the historic behavior from a large corpus of users and makes the recommendation based on that. There are two major categories for CF algorithms: the memory-based methods explicitly figure out the user or item similarities [13, 8] or combine them together [30] and make recommendations directly based on the target user’s past behavior. The model-based methods learn a “model” of each user’s behavior and use it to provide recommendations. The latent factor models [14] implemented by matrix factorization [16, 25, 21] have become quite popular during recent years. Today both the memory and model-based methods are widely used in commercial personalized recommender systems [7, 6].

While many research efforts focus on the personalization algorithms for recommender systems, there are, to the best of our knowledge, very few work that investigates which situations the users need personalized recommendation or not. Such idea has been proposed in Web search scenario [29], where the variation of user search intent is modelled by characterizing the queries with a variety of features and the search results. Even this work is still essentially different from our scenario where there is no any query and the recommender system needs to judge the user’s preference on whether to personalize at each recommendation request. Another work [28] studies the relationship between recommendation accuracy and the item popularity but it does not address the problem of whether to personalize or not.

Risk management and modelling are a significant component for financial institutions, where the companies need to evaluate the risk (i.e, the possibility of failing) for each new candidate investment and how it fits in their previous portfolio [15]. Recently, this idea has been introduced to the information retrieval area: risk-sensitive models are proposed to measure the uncertainty of document ranking performance [34, 33]; optimization algorithms are introduced to improve the robustness of ranking models [2, 33]; the modern portfolio theory is applied to the ranking model (as a document recombination process) to reduce the risk [31]. In addition, a risk-sensitive retrieval task is added in TREC Web track 2013 to maintain the good robustness of IR models [3]. For recommender systems, the risk of delivering an item or a

list of items is actually much significant. However, quite few work explicitly models the risk to improve the performance of recommender systems. In [27], the idea of portfolio theory for ranking documents is borrowed to collaborative filtering to adaptively diversify the recommended items. Compared with our work, it mainly focuses diversification of recommended items instead of carefully modelling the individual item risk and the decision making of whether to personalize.

## 3. RECOMMENDATION RISK ANALYSIS

In this section, we make use of the concepts of portfolio retrieval [31, 20] to conduct our analysis. To make this paper self-contained, we first present the basic concepts of portfolio retrieval in the preliminaries section, and for details we refer to [31, 20]. We then analyze the (non-) personalized item portfolios using the expected reward and its associated risk and study their relationships in the risk-reward diagram.

### 3.1 Preliminaries

**DEFINITION 1 (ITEM PORTFOLIO).** *An item portfolio  $p$  is a collection of  $n$  items with a weight  $w_i$  assigned to each item  $i$ :*

$$p = \{(i, w_i)\}_{i=1..n}, \text{ where } \sum_i w_i = 1. \quad (1)$$

As will be detailed later, the weight is relevant to the importance of each item, which will finally determine the ranking of it. In our proposal, the weight  $w_i$  can be any positive or negative value in  $\mathbb{R}$ , and the larger the weight the more important the item is. Such setting is different with [31] but this is a more general case of portfolio weight, just as the asset short-selling in finance [18]. Without loss of generality, we assume the portfolio includes  $n$  items with indices  $i = 1 \dots n$ . Suppose the user  $u$ ’s preference on item  $i$  is a random variable, denoted as  $r_{u,i}$ , and its mean and variance are  $\mu_{u,i}$  and  $\sigma_{u,i}^2$  respectively, then the user  $u$ ’s preference for an item portfolio  $p$ , denoted as  $R_{u,p}$ , is calculated as a linear function over the item weights:

$$R_{u,p} = \sum_{i=1}^n w_i r_{u,i}. \quad (2)$$

The expected reward and variance of  $R_{u,p}$  are

$$\mu_{u,p} \equiv E[R_{u,p}] = \sum_{i=1}^n w_i E[r_{u,i}] = \mathbf{w}^T \boldsymbol{\mu}, \quad (3)$$

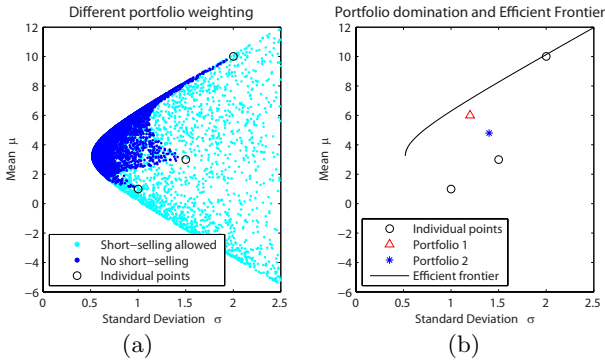
$$\sigma_{u,p}^2 \equiv Var[R_{u,p}] = \sum_{i=1}^n \sum_{j=1}^n w_i w_j Cov(i, j) = \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w}, \quad (4)$$

where we have already used vector representations with  $\mathbf{w}$  as weight vector for the items;  $\boldsymbol{\mu} = (\mu_{u,1} \dots \mu_{u,n})^T$ ;  $\boldsymbol{\Sigma}$  as the covariance matrix about the user’s preference on items in  $p$ . The element of  $\boldsymbol{\Sigma}$  is  $\Sigma_{ij} = \rho_{ij} \sigma_{u,i} \sigma_{u,j}$  [18, 31], where  $\rho_{ij}$  is the correlation between item  $i$  and  $j$ . With the above definition and the calculation of the mean and variance of the user’s preference on an item portfolio, Figure 2(a) gives an illustration of the mean and standard deviation of three individual items, as well as the different portfolios based on these items.

**DEFINITION 2 (ITEM PORTFOLIO DOMINATION).** *For a user  $u$ , an item portfolio  $p_1$  is said to dominate another item portfolio  $p_2$ , denoted as  $p_1 \triangleright p_2$ , if and only if it satisfies*

$$\mu_{u,p_1} > \mu_{u,p_2} \quad (\text{larger expected reward}), \text{ and} \quad (5)$$

$$\sigma_{u,p_1}^2 < \sigma_{u,p_2}^2 \quad (\text{smaller uncertainty}). \quad (6)$$



**Figure 2: An example of portfolio mean-variance, domination and efficient frontier.**

An example of item portfolio domination is given in Figure 2(b) where portfolio 1 dominates portfolio 2. Obviously, in the  $\mu$ - $\sigma$  plot, left-upper points dominate the right-lower ones as the former is with larger expectation and smaller standard deviation than that of the latter.

**DEFINITION 3 (EFFICIENT FRONTIER).** *Given a set of items with their individual means, variances, and correlations, the efficient frontier is a set of item portfolios, each of which is dominated by none of other possible item portfolios.*

Given an expected relevance  $\mu_p$ , the corresponding point on the efficient frontier can be calculated by minimizing the risk; examples include the well-known Markowitz's Objective [18]. Its variant for web search diversification was also proposed in [20]. More specifically, we have

$$\begin{aligned} \min_{\mathbf{w}} \quad & \mathbf{w}^T \Sigma \mathbf{w} \\ \text{s.t.} \quad & \mathbf{w}^T \mathbf{1} = 1 \\ & \mathbf{w}^T \boldsymbol{\mu} = \mu_p. \end{aligned} \quad (7)$$

The closed form solution of Eq. (7) is

$$\mathbf{w}_M = \frac{\begin{vmatrix} 1 & \mathbf{1}^T \Sigma^{-1} \boldsymbol{\mu} \\ \mu_p & \boldsymbol{\mu}^T \Sigma^{-1} \boldsymbol{\mu} \end{vmatrix} \Sigma^{-1} \mathbf{1} + \begin{vmatrix} \mathbf{1}^T \Sigma^{-1} \mathbf{1} & 1 \\ \boldsymbol{\mu}^T \Sigma^{-1} \mathbf{1} & \mu_p \end{vmatrix} \Sigma^{-1} \boldsymbol{\mu}}{\begin{vmatrix} \mathbf{1}^T \Sigma^{-1} \mathbf{1} & \mathbf{1}^T \Sigma^{-1} \boldsymbol{\mu} \\ \boldsymbol{\mu}^T \Sigma^{-1} \mathbf{1} & \boldsymbol{\mu}^T \Sigma^{-1} \boldsymbol{\mu} \end{vmatrix}}. \quad (8)$$

With such setting of  $\mu_p$  and  $\mathbf{w}_M$ , the variance of  $p$  is

$$\sigma_p^2 = \mathbf{w}_M^T \Sigma \mathbf{w}_M, \quad (9)$$

which is a quadratic polynomial of  $\mu_p$

$$\sigma_p^2 = \theta_1 \mu_p^2 + \theta_2 \mu_p + \theta_3, \quad (10)$$

where

$$\begin{aligned} \theta_1 &= \frac{(\mathbf{x}\boldsymbol{\mu} - \mathbf{y}\mathbf{1})^T \Sigma^{-1} (\mathbf{x}\boldsymbol{\mu} - \mathbf{y}\mathbf{1})}{(xz - y^2)^2}, \\ \theta_2 &= \frac{2(\mathbf{x}\boldsymbol{\mu} - \mathbf{y}\mathbf{1})^T \Sigma^{-1} (z\mathbf{1} - \mathbf{y}\boldsymbol{\mu})}{(xz - y^2)^2}, \\ \theta_3 &= \frac{(z\mathbf{1} - \mathbf{y}\boldsymbol{\mu})^T \Sigma^{-1} (z\mathbf{1} - \mathbf{y}\boldsymbol{\mu})}{(xz - y^2)^2}, \end{aligned} \quad (11)$$

and  $x = \mathbf{1}^T \Sigma^{-1} \mathbf{1}$ ,  $y = \mathbf{1}^T \Sigma^{-1} \boldsymbol{\mu} = \boldsymbol{\mu}^T \Sigma^{-1} \mathbf{1}$ , and  $z = \boldsymbol{\mu}^T \Sigma^{-1} \boldsymbol{\mu}$ . From Eq. (10) we can see  $\sigma_p$  is a squared root of a quadratic polynomial of  $\mu_p$  and it forms part of a hyperbola in the  $\mu$ - $\sigma$  plot. Figure 2(b) gives an illustration of the corresponding frontier curve.

Up to this point, we have introduced the basic building blocks of portfolio theory for recommendation tasks, such as item portfolio, item portfolio domination and efficient frontier. The detailed information on such concepts in finance can be found in [11, 15]. Next, we shall introduce our initial risk analysis of two recommendation portfolios.

### 3.2 Risk Analysis of Two Rec. Portfolios

We are now ready to analyze the expected reward and its risk on two item portfolios (one from the personalized and one from the non-personalized). Without loss of generality, in this paper we consider two well-known recommendations algorithms. For the personalized recommendation algorithm, we consider Bayesian personalized ranking (BPR) [22], one of state-of-the-art ranking-based CF algorithms. And for the non-personalized recommendation algorithm, we consider the popularity-based recommendation (POP), which simply recommends the items with the highest number of positive feedbacks.

The relationship between the personalized recommendation and non-personalized recommendation can be illustrated by four typical cases in the risk-reward diagram, as shown in Figure 3. For each case, the three bar diagrams show the actual ranking performance comparison between BPR and POP after portfolio optimization, which picks an optimal point on the efficient frontier. The right  $\mu$ - $\sigma$  plot gives the mean (expected reward) and the standard deviation (risk) for two sets of individual items, two efficient frontiers, and two optimized item portfolios. In this section, we focus the analysis on whether to personalize or not, given these two item portfolios optimized independently. The detailed discussion of estimating the individual item mean and standard deviation and optimizing the item portfolio for reranking will be presented in the following sections.

Figure 3(a) and 3(b) show two cases that one item portfolio dominates the other. We can see the dominating one always obtains higher ranking performance than the dominated one. In both cases, the selection (switch) criterion is straightforward: given two item portfolios, the one with higher expected reward and lower risk (i.e., the dominating portfolio) will be chosen.

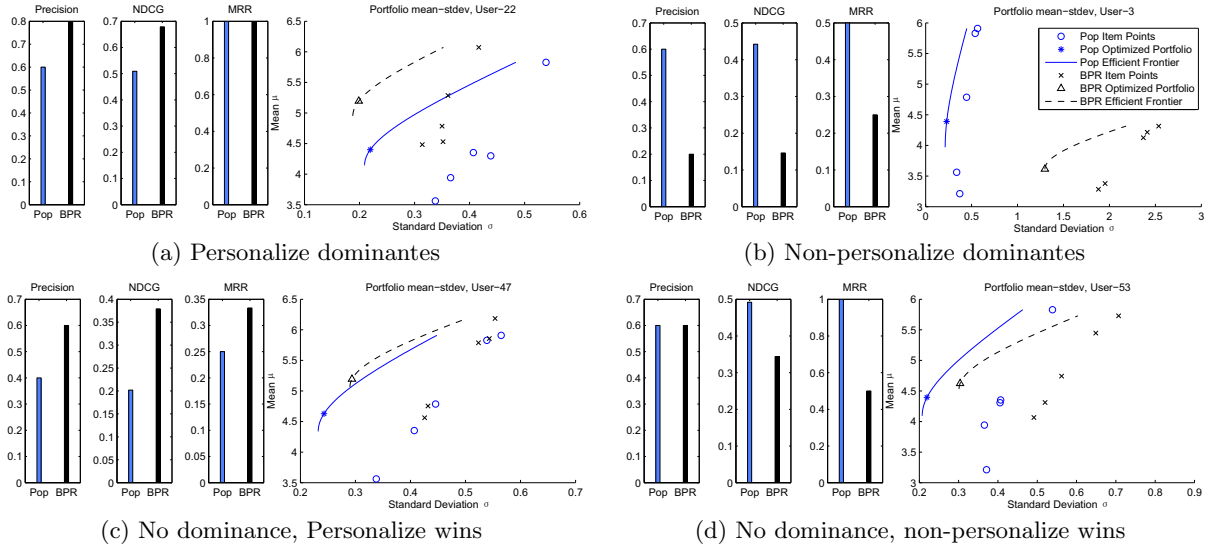
Figure 3(c) and 3(d) present another two cases of no direct dominance, where one portfolio has higher expected reward while the other has lower risk. The item portfolio cannot be easily selected as the cases of direct dominance. For example, as shown in Figure 3(c), the portfolio with lower risk fails to outperform the one with higher expected reward. A different result is exhibited in Figure 3(d), where the item portfolio with higher expected reward is outperformed by the one with lower risk. In such situations, a switch algorithm that considers how to select an optimal portfolio from those of no direct dominance is needed.

Based on previous analysis, given two recommended item lists from personalized and non-personalized recommendation algorithms, what we need to do is threefold:

**Item Level** - Estimate the expected reward and risk for each individual item. That is to obtain the individual item points on  $\mu$ - $\sigma$  plots in Figure 2 and 3. In our paper, we leverage Bayesian linear regression based on latent factor model to estimate the mean and variance of the preference value of user  $u$  on each candidate item  $i$ , which will be briefly discussed in Section 4.1.

**Portfolio Level** - Given the mean and variance of each item and their correlations, find the optimal weighting combination of them. That is to obtain the optimal weighting points on the efficient frontiers in Figure 3. We propose our portfolio optimization algorithm in Section 4.2.

**Two-Portfolio Level** - Given the two optimized item port-



**Figure 3: Typical cases for the expected reward and risk analysis of the personalized and non-personalized portfolios. Data source: MovieLens.**

folios from personalized and non-personalized recommendations respectively, design the mean-variance-aware switch algorithms to choose whether to personalize or not. We propose a series of switch algorithms in Section 4.3.

## 4. THE ALGORITHM

### 4.1 Individual Item Estimation

On the item level, we focus on the estimation of  $\mu_{u,i}$ ,  $\sigma_{u,i}$  for each item. Here we leverage the Bayesian latent factor model to estimate them, inspired by [25, 12]<sup>1</sup>. Latent factor model is a typical well-known personalized recommendation algorithm. Considering the probability distribution of the user’s rating, we can use the probabilistic matrix factorization (PMF) [25] to model, that is

$$p(r_{u,i}|\mathbf{U}_u, \mathbf{V}_i, \sigma^2) = \mathcal{N}(r_{u,i}|\mathbf{U}_u^T \mathbf{V}_i, \sigma^2). \quad (12)$$

PMF suggests to provide prior distributions for  $\mathbf{U}$  and  $\mathbf{V}$  and then the *maximum a posteriori* (MAP) solutions for  $\mathbf{U}$  and  $\mathbf{V}$  for the given rating data can be directly obtained. Here we relate the uncertainty in the user-item rating  $r_{u,i}$  with the uncertainty from the user factors [27], while taking the item factors as learnt and fixed. Therefore, here we fix the item latent factors  $\mathbf{V}$  from this MAP solution [25], and the distribution of  $\mathbf{U}$  in a form of Bayesian inference in linear regression can be obtained.

Denote the user  $u$ ’s behavior observation as  $\mathbf{D}_u$ , where each row is the latent factor  $\mathbf{V}_i^T$  of the item  $i$  that he/she has rated, and denote the corresponding rating vector as  $\mathbf{r}_u$ . Suppose the prior distribution of  $u$ ’s latent factor  $\mathbf{U}_u$  is a Gaussian  $p(\mathbf{U}_u) = \mathcal{N}(\mathbf{U}_u|\mathbf{U}_0, \mathbf{\Sigma}_0)$ . Then its posterior distribution is still a Gaussian

$$p(\hat{\mathbf{U}}_u) = \mathcal{N}(\mathbf{U}_u|E[\hat{\mathbf{U}}_u], Var[\hat{\mathbf{U}}_u]), \quad (13)$$

where

$$\begin{aligned} E[\hat{\mathbf{U}}_u] &= (\mathbf{\Sigma}_0^{-1} + \sigma^{-2} \mathbf{D}_u^T \mathbf{D}_u)^{-1} (\mathbf{\Sigma}_0^{-1} \mathbf{U}_0 + \sigma^{-2} \mathbf{D}_u^T \mathbf{r}_u), \\ Var[\hat{\mathbf{U}}_u] &= (\mathbf{\Sigma}_0^{-1} + \sigma^{-2} \mathbf{D}_u^T \mathbf{D}_u)^{-1}. \end{aligned} \quad (14)$$

<sup>1</sup>Note that other alternatives to estimate these parameters can also be adopted [24, 35].

Therefore, the Gaussian distribution of the rating  $r_{u,i}$  can be written as

$$p(r_{u,i}) = \mathcal{N}(r_{u,i}|\mu_{u,i}, \sigma_{u,i}^2), \quad (15)$$

where the mean and variance are

$$\mu_{u,i} = E[\hat{\mathbf{U}}_u]^T \mathbf{V}_i, \quad \sigma_{u,i}^2 = \mathbf{V}_i^T Var[\hat{\mathbf{U}}_u] \mathbf{V}_i. \quad (16)$$

We also need to estimate the mean values and variances of the items from a non-personalized perspective. To do this, we assume an “average user” who has an averaged rating behavior across the whole user set. The posterior distribution of latent factors of the “average user” is calculated by taking average across the parameters of the posterior distributions of all users. Then the mean and variance of individual items are estimated by Eq. (16)<sup>2</sup>.

In addition, for the item correlation  $\rho_{i,j}$ , a routine solution is to leverage the item-item rating similarity in item-based  $k$ NN algorithms [8]. Note that the Pearson correlation of item latent factors can also be adopted here.

### 4.2 Item Portfolio Optimization

In this section, we focus on the item portfolio weighting optimization analysis. Investors in finance always adjust their shares in an investment portfolio in order to have better expected reward as well as to reduce risk [15], here in the recommendation scenario, it is preferred to provide highly relevant items while the risk of the user dissatisfaction being controlled. This target can be achieved by refining the item ranking list according to the weights of the items.

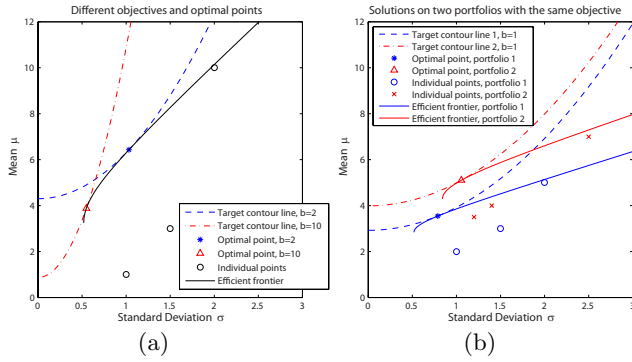
#### 4.2.1 Objective and Solution

In [31], Wang et al. provided a general form of the objective function in practice of information retrieval.

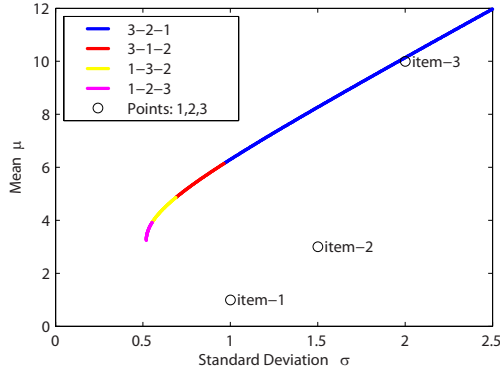
$$\begin{aligned} \max_{\mathbf{w}} \quad & \mathbf{w}^T \boldsymbol{\mu} - b \mathbf{w}^T \boldsymbol{\Sigma} \mathbf{w} \\ \text{s.t.} \quad & \mathbf{w}^T \mathbf{1} = 1 \end{aligned} \quad (17)$$

Here the parameter  $b$  indicates the risk-averse level of the user. Thus the contour line of objective function forms a

<sup>2</sup>Due to the rating prediction model of PMF, mean values of some items are possibly higher than 5 (rating upper bound), as shown in Figure 3.



**Figure 4: Contour lines of objective functions and the different optimal solution for different objective functions.**



**Figure 5: An example of rankings and their corresponding regions on the efficient frontier. The covariances are  $cov(1,2) = -0.5$ ,  $cov(1,3) = -0.7$ ,  $cov(2,3) = -0.4$ .**

quadratic curve on the  $\mu$ - $\sigma$  plot. To illustrate, the left panel of Figure 4 shows the efficient frontier of three individual items, and two contour lines with two different risk-averse levels ( $b = 2$  and  $b = 10$ ). The right panel of the figure shows two portfolios and two contour lines with the same risk-averse level tangent with them.

The optimal weight point is the point of tangency between the objective quadratic curve and the efficient frontier. The solution can be obtained by the following equations:

$$\begin{aligned} \mu_p &= t + b\sigma_p^2, \\ \sigma_p^2 &= \theta_1\mu_p^2 + \theta_2\mu_p + \theta_3, \\ \frac{\partial\sigma_p}{\partial\mu_p} &= \frac{1}{2(b\mu_p - bt)^{\frac{1}{2}}} = \frac{2\theta_1\mu_p + \theta_2}{2(\theta_1\mu_p^2 + \theta_2\mu_p + \theta_3)^{\frac{1}{2}}}, \end{aligned} \quad (18)$$

where the intercept  $t$  of the contour line  $\mu_p = t + b\sigma_p^2$  is the objective value in Eq. (17),  $\theta_i$ 's are the same as in Eq. (11). The solution of  $\mu_p$  is

$$\mu_p = \frac{1 - b\theta_2}{2b\theta_1}, \quad (19)$$

and the analytic solution for the optimal weight  $w^*$  can be further obtained by Eq. (8).

#### 4.2.2 Item Reranking via Portfolio Optimization

Compared to the portfolio weighting in finance, here the weight  $w_i$  for each item indicates how much attention the recommender system wants the target user to pay on the item  $i$ . Therefore, according to the position bias [4], the

final delivered items should be ranked by the descending order of their weights.

Figure 5 gives an illustration of the relationship between item rankings and their corresponding regions on the efficient frontier. We can see not all possible item rankings necessarily occur on the efficient frontier (no 2-1-3 and 2-3-1). Specifically, the mean is more emphasized in the portfolios with ranking 3-2-1 while the variance is more emphasized in the portfolios with ranking 1-2-3. The other two rankings try to strike a balance between mean and variance.

### 4.3 Proposed Switch Algorithms

From previous sections, we can now use the proposed risk management model to calculate the expected reward and risk for any item portfolio to the target user, and then find the optimal portfolio weighting strategy to rerank the recommended items. Now we aim to propose switch algorithms, which make decisions between personalized and non-personalized recommendation portfolios to get the highest utility, based on the expected reward and risk.

#### 4.3.1 Switch Problem Statement

Choosing whether to personalize or not in recommendation is essentially a switch decision problem. Given the information of two item portfolios (from personalized and non-personalized recommendation respectively), the switch policy  $\pi$  chooses one to deliver, denoted as  $\pi(u, p_1, p_2)$ .

Suppose there is a utility function  $r(u, p)$  which estimates the utility of recommending an item portfolio  $p$  to user  $u$ . Then the corresponding optimal policy can be defined according to the utility as follows:

$$\pi^* = \arg \max_{\pi} \sum_{(u, p_1, p_2)} r(u, \pi(u, p_1, p_2)). \quad (20)$$

#### 4.3.2 Switch Algorithms

Here we propose some heuristic and learning-based switch algorithms.

**Mean prior (Mean)** - Select the portfolio with higher expected reward.

$$\pi(u, p_1, p_2) = \arg \max_{p \in \{p_1, p_2\}} \mu_{u,p}. \quad (21)$$

**Variance prior (Var)** - Select the portfolio with lower risk.

$$\pi(u, p_1, p_2) = \arg \min_{p \in \{p_1, p_2\}} \sigma_{u,p}. \quad (22)$$

**Linear risk target (Lin)** - Select the portfolio with higher  $\mu - b\sigma$  value.

$$\pi(u, p_1, p_2) = \arg \max_{p \in \{p_1, p_2\}} \mu_{u,p} - b\sigma_{u,p}. \quad (23)$$

**Quadratic risk target (Quad)** - Select the portfolio with higher  $\mu - b\sigma^2$  value.

$$\pi(u, p_1, p_2) = \arg \max_{p \in \{p_1, p_2\}} \mu_{u,p} - b\sigma_{u,p}^2. \quad (24)$$

**Dominance motivated (Dom)** - Select the personalized portfolio by default, unless it is dominated by the non-personalized one.

$$\pi(u, p_1, p_2) = p_2 \triangleright p_1 ? p_2 : p_1. \quad (25)$$

**Golden Switch (Golden)** - In order to check the potential of these switch algorithms, here we provide an upper bound algorithm, which exactly switches to the better one.

**Adaptive risk parameter (X-A)**<sup>3</sup>- Different users have different risk-averse levels. For example, some users expect the recommender system could provide some serendipitous items for them to discover new interest, while some users may just need the recommendations to be consistent with their previous interest. This risk-averse level parameter is indeed the  $b$  in Eq. (17), denoted as  $b_u$  here.

In order to adaptively learn this parameter for each individual user, we conduct a cross validation on the training data to empirically tune the parameter  $b_u$  for each user  $u$  and pick the optimal value. Then during the portfolio optimization for each user  $u$ , we adopt this adaptive parameter  $b_u$ . And during the latter switch decision process, for the Lin and Quad algorithm, we also adopt  $b_u$  to make decision.

Note that the calculations on item level, portfolio level, and two-portfolio level are all independent from (non-) personalized recommendation algorithms, which indicates that our work can be seamlessly generalized on other (non-) personalized recommendation algorithms.

## 5. EXPERIMENT

In this section, we intend to answer the following questions from the experiments: (i) Will the portfolio optimization models improve the ranking performance in both personalized and non-personalized cases? (ii) Given the two optimized ranking lists, which switch algorithm works the best and why?

### 5.1 Experiment Setup

**Datasets** - We base our experiments on the two popular datasets MovieLens-100k<sup>4</sup> and Netflix. MovieLens contains 100K ratings from 943 users and 1,682 items, while Netflix contains 100M ratings from 480,189 users and 17,770 items. Following the experimental setting of [19] we pick the 4 or 5-star ratings as positive feedbacks and give a 4:1 random splitting for the training and test data.

**Compared Algorithms** - We compare the portfolio-based reranking algorithms and switch algorithms to the base recommendation algorithms. Also, we explore and discuss the combination of the three types of algorithms.

Base recommendation algorithms retrieve top- $N$  items for each target user. Here we choose BPR and POP as the personalized and non-personalized algorithms respectively.

Portfolio-based reranking algorithms take the retrieved top- $N$  items (from based recommendation algorithms) and the target user  $u$ 's profile as inputs. It evaluates the expected reward and risk for each item and then performs the portfolio optimization to obtain the optimal weighting for the combination of items, which finally will determine the ranking of the items. In more detail, we propose to directly solve out the optimal weighting solution (denoted as **Opt**) as in Section 4.2.1 and then rerank the items by the descending order of the optimal weighting. Furthermore, the Bayesian latent factor model discussed in Section 4.1 is leveraged to estimate the individual item expected reward and risk. For comparison, a portfolio-based greedy ranking algorithm proposed in [31, 27] is used (denoted as **Greedy**) which sequentially chooses the item to maximize the incremental target in Eq. (17), and the expected reward and risk are calculated accordingly [27].

Switch algorithms take the expected reward and risk of the two optimized item portfolios (**Opt-BPR** and **Opt-POP**) along with the user's risk-averse level parameter  $b$  as inputs, and choose between the two portfolios and finally return the

<sup>3</sup>Here X is one of the previously introduced algorithms.

<sup>4</sup>This small dataset is chosen because of its convenience of providing case illustrations.

**Table 1: Item reranking performance on MovieLens**

Algorithm	P@3	P@5	NDCG@3	NDCG@5	MRR
BPR	<b>0.2865</b>	<b>0.2623</b>	0.2465	0.2289	0.4393
Greedy-BPR			0.2580	0.2312	0.4547
Opt-BPR			0.2496	0.2342	0.4640
Opt-BPR-A			<b>0.2593</b>	<b>0.2409</b>	<b>0.4863</b>
POP	0.2485	0.2202	0.2127	0.1955	0.4378
Greedy-POP			0.2237	0.2034	0.4772
Opt-POP			0.2193	0.2036	0.4760
Opt-POP-A			0.2223	0.2064	0.4847
Improvement	-	-	5.19%*	5.24%*	10.70%*

**Table 2: Item reranking performance on Netflix**

Algorithm	P@3	P@5	NDCG@3	NDCG@5	MRR
BPR	<b>0.1177</b>	<b>0.1142</b>	0.1198	0.1168	0.2069
Greedy-BPR			0.1286	0.1296	0.2472
Opt-BPR			0.1308	0.1317	0.2544
Opt-BPR-A			<b>0.1329</b>	<b>0.1328</b>	<b>0.2580</b>
POP	0.0887	0.0907	0.0881	0.0895	0.1642
Greedy-POP			0.0915	0.0969	0.1902
Opt-POP			0.0917	0.0970	0.1905
Opt-POP-A			0.0983	0.0999	0.2000
Improvement	-	-	10.93%*	13.70%*	24.70%*

corresponding item ranking list as the output. The compared switch algorithms are listed in Section 4.3, denoted as **Mean**, **Var**, **Lin**, **Quad**, **Dom**, and **Golden** respectively. Note that "A" denotes that the algorithm is based on the adaptive risk parameter setting, such as **Opt-BPR-A** and **Opt-Mean-A**.

**Evaluation Measures** - Three ranking measures are used in our experiments: precision ( $P@n$ ), normalized discounted cumulative gain ( $NDCG@n$ ) and mean reciprocal rank (MRR). Besides the absolute performance values, the improvement of the best algorithm against the baseline algorithm is also calculated. In addition, Wilcoxon signed-rank tests are conducted and statistically significant improvements are marked with \*.

### 5.2 Results and Analysis

In this section, we first compare portfolio-based reranking algorithms and then we test the effectiveness of different switch algorithms. Finally, we provide case studies to further illustrate the effectiveness of our models.

#### 5.2.1 Item Reranking via Portfolio Optimization

We present the performance for the two portfolio-based reranking algorithms (**Opt** and **Greedy**) based on the two base recommendation algorithms (**BPR** and **POP**). In addition, we add the algorithms with adaptive risk parameter (**Opt-X-A**) into the comparison. The overall results on MovieLens and Netflix datasets are shown in Table 1 and 2 respectively. The improvement is calculated between the best algorithm and the baseline BPR.

From the overall results we can have the following observations. (i) On all evaluation measures, the two portfolio-based reranking algorithms **Greedy** and **Opt** bring improvement on both of the output of base recommendation algorithms **BPR** and **POP**. (ii) Compared with **Greedy**, **Opt** brings a higher improvement, which verifies the effectiveness of our method that tries to find the optimal portfolio weighting for items and ranks them by the weight. (iii) Furthermore, **Opt-X-A** with the setting of adaptive learning on parameter  $b_u$  for each target user leads a further high improvement. This indicates different users have different risk-averse levels, which can be traced through analyzing each user's history data. (iv) Among the three evaluation measures, the improvements on MRR are the most significant, which is consistent with the findings in [31, 32] that reciprocal ranking metric is optimized when diversifying the ranking list.

To see the impact of the risk-averse level parameter  $b$ , we trace the performance of **Opt-BPR** and **Greedy-BPR** against  $b$  with comparison to non- $b$  BPR and adaptively tuned- $b$  **Opt**-

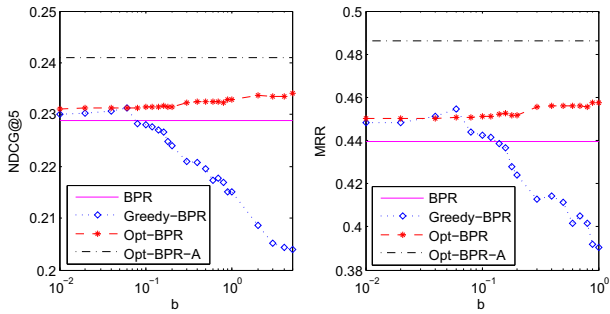


Figure 6: Performance of different item reranking algorithms against  $b$  - MovieLens.

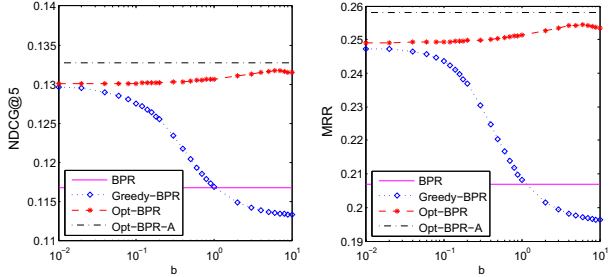


Figure 7: Performance of different item reranking algorithms against  $b$  - Netflix.

BPR-A. The results are shown in Figure 6 and 7. The corresponding results on POP are much similar. We can see that the performance of Greedy is much sensitive against  $b$  and will drop even lower than the original BPR when  $b$  gets larger than some value. This is mainly caused by the myopic picking strategy of Greedy. On the other hand, as has been shown in Figure 5, as the risk parameter  $b$  changes, the item ranking from Opt will not change frequently which is consistent with the results in Figure 6 and 7.

### 5.2.2 Switch Algorithms

Based on the portfolio-based reranking algorithm, for each user, consider there are two item recommendation lists (portfolios), from Opt-BPR (or Opt-BPR-A) and Opt-POP (or Opt-POP-A), the switch algorithm chooses one list (portfolio) to recommend. The results on two datasets are presented in Table 3 and 4 respectively. The switch/total improvements are calculated between the best algorithm and BPR-Opt-A/BPR, respectively.

From the results we can observe: (i) the baseline switch algorithm Mean usually results in a performance drop from Opt-BPR, which means that purely selecting a list by the higher expected reward is not sufficiently effective. (ii) The risk-aware baseline switch algorithm Var leads to improvement on MovieLens while bringing some decreases of NDCG and MRR on Netflix. Given the item lists from portfolio-based reranking algorithms, choosing the list with less risk is somehow reasonable but the performance drop on Netflix indicates the instability of this algorithm across different datasets. (iii) Compared with Mean and Var, the mean-variance-aware algorithms Lin and Quad bring improvement of all the measures on two datasets, exhibiting the stable effectiveness of such switch algorithms. Compared with each other, there is no obvious difference of the performance between Lin and Quad. (iv) The dominance-based switch algorithm Dom, although not as effective as Lin or Quad, still improves the performance. Hence the dominating portfolio should always be selected whenever it exists. (v) Equipped with the adaptive risk parameter setting, all the switch algorithms bring further improvements. On MovieLens, Opt-

Table 3: Switch performance on MovieLens

Algorithm	P@3	P@5	NDCG@3	NDCG@5	MRR
Opt-BPR	0.2865	0.2623	0.2496	0.2342	0.4640
Opt-POP	0.2485	0.2202	0.2193	0.2036	0.4760
Opt-Mean	0.2873	0.2500	0.2493	0.2242	0.4841
Opt-Var	0.3070	0.2789	0.2622	0.2454	0.5047
Opt-Lin	0.3048	0.2741	0.2609	0.2432	0.4981
Opt-Quad	0.3034	0.2693	0.2598	0.2390	0.4954
Opt-Dom	0.3056	0.2781	0.2617	0.2443	0.4883
Opt-Golden	0.3494	0.3101	0.3070	0.2749	0.5636
Opt-BPR-A	0.2865	0.2623	0.2593	0.2409	0.4863
Opt-POP-A	0.2485	0.2202	0.2223	0.2064	0.4847
Opt-Mean-A	0.2712	0.2456	0.2395	0.2253	0.4890
Opt-Var-A	0.3048	0.2671	0.2714	0.2430	<b>0.5085</b>
Opt-Lin-A	<b>0.3085</b>	<b>0.2728</b>	<b>0.2745</b>	0.2458	0.5057
Opt-Quad-A	0.3019	0.2662	0.2705	0.2436	0.5046
Opt-Dom-A	0.2990	0.2715	0.2661	<b>0.2483</b>	0.5051
Opt-Golden-A	0.3494	0.3101	0.3131	0.2805	0.5802
Switch Imprv.	7.68%*	4.00%	5.86%	3.07%	4.57%
Total Imprv.	7.68%*	4.00%	11.36%*	8.48%*	15.75%*

Table 4: Switch performance on Netflix

Algorithm	P@3	P@5	NDCG@3	NDCG@5	MRR
Opt-BPR	0.1177	0.1142	0.1308	0.1317	0.2544
Opt-POP	0.0887	0.0907	0.0918	0.0970	0.1905
Opt-Mean	0.1194	0.1186	0.1302	0.1319	0.2510
Opt-Var	0.1231	0.1180	0.1295	0.1211	0.2309
Opt-Lin	0.1241	0.1187	0.1335	0.1267	0.2493
Opt-Quad	0.1241	0.1188	0.1335	0.1319	0.2492
Opt-Dom	0.1188	0.1159	0.1311	0.1318	0.2522
Opt-Golden	0.1578	0.1393	0.1724	0.1578	0.3090
Opt-BPR-A	0.1177	0.1142	0.1329	0.1328	0.2580
Opt-POP-A	0.0887	0.0907	0.0983	0.0999	0.2000
Opt-Mean-A	0.1259	0.1145	0.1469	0.1351	0.2730
Opt-Var-A	0.0876	0.0913	0.0929	0.0993	0.1882
Opt-Lin-A	0.1289	<b>0.1206</b>	0.1483	<b>0.1391</b>	<b>0.2776</b>
Opt-Quad-A	<b>0.1291</b>	<b>0.1206</b>	<b>0.1486</b>	<b>0.1391</b>	0.2771
Opt-Dom-A	0.1178	0.1151	0.1329	0.1333	0.2586
Opt-Golden-A	0.1578	0.1393	0.1792	0.1605	0.3173
Switch Imprv.	9.69%*	5.60%*	11.81%*	4.74%	7.60%*
Total Imprv.	9.69%*	5.60%*	24.04%*	19.09%*	34.17%*

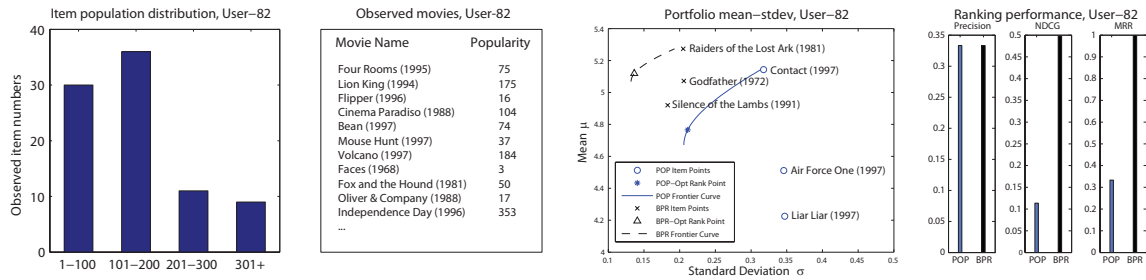
Lin-A obtains the best performance, while on Netflix, the best performance is brought by Opt-Quad-A in Precision and NDCG, and Opt-Lin-A in MRR.

### 5.2.3 Case Studies

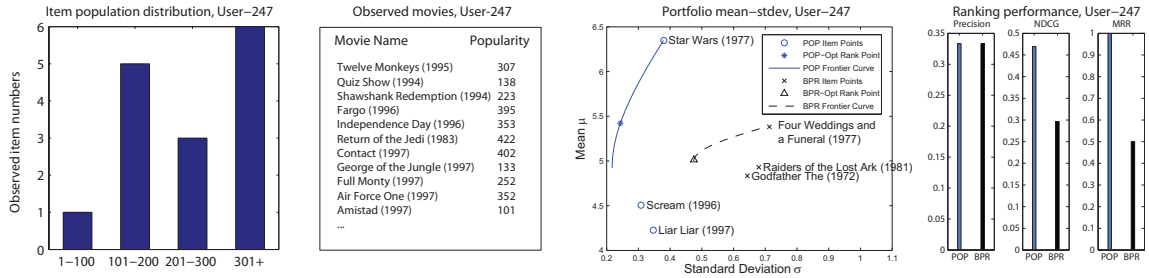
Figure 8 provides two typical MovieLens cases with the users' historic item information, mean-variance-aware analysis for (non-) personalized recommendations, and their ranking performances.

User-82 is a 50 years old male programmer. He has rated 86 movies, most of which are of relatively low popularity. It is typical that the mainstream movies do not match the interest of people of such an age. On the other hand, the 86 movies he has rated provide sufficient information for the recommender system to learn his profile and to catch his personal interest. In the mean-variance-aware analysis, the portfolio of the three recommended movies from personalized algorithm dominates that from non-personalized algorithm. Thus all the switch algorithms will choose the personalized recommendation. And the final ranking performances support such analysis.

User-247 is a 28 years old male engineer. Compared to User-82, he has only 15 feedbacks, most of which are popular ones. It is yet another common case as young people tend to find their interests in the mainstream movies. Also the relatively low number of feedbacks do not support a sufficiently learned preference profile for him. In the mean-variance-aware analysis, it shows the dominance of the non-personalized portfolio on the personalized one. Therefore, the non-personalized recommendation should be chosen for him. Again, the ranking performances are consistent with the analysis.



(a) Case study for User-82: a 50-year old male programmer.



(b) Case study for User-247: a 28-year old male engineer.

Figure 8: Examples of the user history, mean-variance-aware analysis, and the performance.

## 6. CONCLUSION AND FUTURE WORK

In this paper, the portfolio retrieval theory [31, 20] is used to study the relationships between personalized and non-personalized recommendations. Under a single risk-reward diagram, we studied the efficient frontiers for both recommendation solutions. In either case, diversifying a ranking list would reduce the risk of a recommended item list. We found that the personalized recommendation does not always dominate the non-personalized one. For each target user, we developed several strategies to determine whether or not to personalize the recommendation. The experimental results verified the effectiveness of the proposed portfolio-based reranking algorithms and switch algorithms. For the future work, we plan to deal with the blending of two or multiple item lists from different (non-) personalized recommendation algorithms, which is a generalization of the research problem in this paper.

## 7. REFERENCES

- [1] J. Bennett and S. Lanning. The netflix prize. In *KDD Cup*, 2007.
- [2] K. Collins-Thompson. Reducing the risk of query expansion via robust constrained optimization. In *CIKM*, 2009.
- [3] K. Collins-Thompson, P. Bennett, and F. Diaz. Trec 2013 web track proposal. 2013.
- [4] N. Craswell, O. Zoeter, M. Taylor, and B. Ramsey. An experimental comparison of click position-bias models. In *WSDM*, 2008.
- [5] P. Cremonesi, Y. Koren, and R. Turrin. Performance of recommender algorithms on top-n recommendation tasks. In *RecSys*, 2010.
- [6] A. S. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: scalable online collaborative filtering. In *WWW*, 2007.
- [7] J. Davidson et al. The youtube video recommendation system. In *RecSys*, 2010.
- [8] M. Deshpande and G. Karypis. Item-based top-N recommendation algorithms. *ACM Trans. Inf. Syst.*, 2004.
- [9] P. Domingos. A unified bias-variance decomposition. In *ICML*, 2000.
- [10] G. Dror, N. Koenigstein, Y. Koren, and M. Weimer. The yahoo! music dataset and kdd-cupaf11. In *KDD Cup*, 2011.
- [11] E. J. Elton, M. J. Gruber, S. J. Brown, and W. N. Goetzmann. *Modern portfolio theory and investment analysis*. John Wiley & Sons, 2009.
- [12] T. Hastie, R. Tibshirani, J. Friedman, and J. Franklin. The elements of statistical learning: data mining, inference and prediction. *The Mathematical Intelligencer*, 27(2):83–85, 2005.
- [13] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl. An algorithmic framework for performing collaborative filtering. In *SIGIR*, 1999.
- [14] T. Hofmann and J. Puzicha. Latent class models for collaborative filtering. In *Proc. of IJCAI*, 1999.
- [15] J. Hull. *Risk Management and Financial Institutions, + Web Site*, volume 733. Wiley, 2012.
- [16] Y. Koren, R. Bell, and C. Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 2009.
- [17] G. Linden, B. Smith, and J. York. Amazon.com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.
- [18] H. Markowitz. Portfolio selection\*. *The journal of finance*, 7(1):77–91, 1952.
- [19] U. Paquet and N. Koenigstein. One-class collaborative filtering with random graphs. 2013.
- [20] D. Rafiei, K. Bharat, and A. Shukla. Diversifying web search results. In *WWW*. ACM, 2010.
- [21] S. Rendle. Factorization machines. In *ICDM*, 2010.
- [22] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In *UAI*, 2009.
- [23] P. Resnick and H. R. Varian. Recommender systems. *Communications of the ACM*, 40(3):56–58, 1997.
- [24] R. Salakhutdinov and A. Mnih. Bayesian probabilistic matrix factorization using markov chain monte carlo. In *ICML*, 2008.
- [25] R. Salakhutdinov and A. Mnih. Probabilistic matrix factorization. *NIPS*, 2008.
- [26] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl. Analysis of recommendation algorithms for e-commerce. In *EC*, 2000.
- [27] Y. Shi, X. Zhao, J. Wang, M. Larson, and A. Hanjalic. Adaptive diversification of recommendation results via latent factor portfolio. In *SIGIR*, 2012.
- [28] H. Steck. Item popularity and recommendation accuracy. In *RecSys*, 2011.
- [29] J. Teevan, S. T. Dumais, and D. J. Liebling. To personalize or not to personalize: modeling queries with variation in user intent. In *SIGIR*, 2008.
- [30] J. Wang, A. P. de Vries, and M. J. T. Reinders. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. In *SIGIR*, 2006.
- [31] J. Wang and J. Zhu. Portfolio theory of information retrieval. In *SIGIR*, 2009.
- [32] J. Wang and J. Zhu. On statistical analysis and optimization of information retrieval effectiveness metrics. In *SIGIR*, 2010.
- [33] L. Wang, P. N. Bennett, and K. Collins-Thompson. Robust ranking models via risk-sensitive optimization. In *SIGIR*, 2012.
- [34] J. Zhu, J. Wang, I. J. Cox, and M. J. Taylor. Risky business: modeling and exploiting uncertainty in information retrieval. In *SIGIR*, 2009.
- [35] Y.-X. Zhu, W. Zeng, and Q.-M. Zhang. The effect of rating variance on personalized recommendation. In *CCSE*, 2010.