IJDACR
ISSN: 2319-4863

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

A Diversity Recommendation Method Based on Product Marginal Utility

Wang Qian[1] Yu Jijun[2]

[1, 2](*Department of Information System and Engineering, Business School of Sun Yat-sen University Guang Dong Guang Zhou 510275, China*)

Abstract: **Recommender system as Intelligence information tool to help users to find items what they really preferred. However, in electronic commence field, research in consumer behavior theory indicates that consumers usually make purchase decisions based on the product's marginal utility. Utility is defined as the satisfaction that the user gets when purchasing the product. User satisfaction from buying durable products and perishable products with the increase of purchase count could be different. For instance, with smart phones and computers, users are not likely to purchase the same or similar product again in a short time if they already purchased it before, so it is necessary to recommend more diverse products for the user next time. At the same time, some products such as fruit and baby diapers would be purchased again and again, so it is appropriate to recommend the same or similar products for the user. Traditional algorithms do not differentiate between these products when they make diversity recommendations. This paper therefore proposes a method of adaptive diverse and accurate recommendation by constructing a constrained Binary Quadratic Programming (BQP) model to maximize the cumulative marginal utility of the total recommendation list for the user. The proposed method can recommend the highest marginal utility of the single product to the user according to the type of product in the user's previous purchase history. We evaluate the proposed method using an e-commerce (Tmall.com) dataset. The experimental results based on real purchase records show that the proposed recommendation method has high stability and provides superior performance in recommending products with accuracy and diversity.**

Keywords: **Utility; Diversity recommendation; Adaptively; Binary Quadratic Programming.**

1. Introduction

In recent years, various kinds of shopping websites have emerged, offering millions of different products. Information overload has become a problem. Recommender systems can be effective in tackling this issue. Researchers have proposed various recommender algorithms that generate individual recommendations to satisfy user's preferences and needs. Because users have a wide range of interests, researchers have suggested diversity recommendation as a way to meet user requirements [1]. The pursuit of accuracy and diversity in recommendation results has become a popular topic.

In the field of the diversity recommendation algorithms usually diversify the results in a way that assumes that the value/utility of a product is the same and does not change over time for a user. Researchers always focus on the balance between diversity and accuracy [2-3], look to

IJDACR International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

efficiently obtain recommendation results with high diversity [4-6], and use empirical data to study the influence of the recommender system on enterprises and consumers [7-9]. They give little consideration to the marginal utility of products that belong to different categories in the diversity recommendation such as durable and perishable products.

According to consumer behavior theory, a rational consumer chooses the product with the highest marginal utility. A product's marginal utility is dependent on the user's previous purchase history, which means that products with higher marginal utility are more likely to be purchased. However, according to the Law of Diminishing Marginal Utility in economics, products have diminishing marginal utility. For example, the utility of purchasing a second computer is less than the utility of purchasing the first computer. Users are not likely to repeatedly purchase these products in a short time period. At the same time, some products (such as pet food and bottled water) are likely to be purchased regularly. Wang et al. capture this phenomenon and analyze marginal utility in relation to the recommendation [10] but do not examine both the diversity and accuracy of the products.

This paper proposes an adaptive diversity recommendation method that recommends the highest marginal utility single product to users according to their previous purchase history. The paper also constructs a Binary Quadratic Programming (BQP) model to maximize the cumulative marginal utility of the recommendation list for the user, to fill the gap between the utility and the diversity recommendation. Specifically, this paper makes three main contributions:

1. We introduce the method of utilizing marginal utility based on the users' previous purchase history in the diversity recommender system. This system can make adaptive recommendations based on the marginal utility for accuracy and diversity. The method can select the accurate algorithm to make recommendations when the previously purchased products are perishable and can select the diverse algorithm to make recommendations when the previously purchased products are durable.

2. As the length of the recommendation list is limited, we also construct a constrained Binary Quadratic Programming (BQP) model to maximize the cumulative marginal utility of the recommendation list for the user. The new algorithm achieves significant improvement in the evaluation metrics.

3. The proposed method can recommend the same or similar products, and can also provide diverse product recommendations with the highest marginal utility.

This paper is organized as follows. In Section 2, we review related works. In Section 3, we present our proposed recommender method. Section 4 presents our experimental work. Finally, in Section 5, we present several conclusions and suggest directions for future work.

2. Related work

In this section, we briefly describe several classical approaches that are widely applied in recommender systems and highlight related work that also considers the factors. We also show the differences between their methods and ours.

Recommendation technology was first proposed in 1992s as a tool to solve the problem

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

of "information overload" and provide personalized service for users. There are several types of recommendation algorithms, such as content-based recommendation, collaborative filtering(CF) recommendation, knowledge-based recommendation, hybrid recommendations and recommendation based on the machine learning [11-16].In the e-commerce field, the aforementioned techniques are mostly focused on the accuracy of the results. However, a sole focus on accuracy creates a serious problem of homogenization in the recommended results, either the system recommends the same similar items to all consumers, or the similarity between the items is too high for the individual consumer [6, 9]. To avoid this issue, researchers proposed a number of techniques. A typical approach is to introduce diversity metric to balance diversity and accuracy, such as the number of categories or the percentage of the popular products occupied in the recommendations list. Furthermore, In order to avoid providing recommendations for a single user that is too similar (often by measuring an average dissimilarity between all pairs of recommended items, while maintaining an acceptable level of accuracy). Studies [17-18] used an intra-list similarity metric to determine individual diversity. Zhang and Hurley [2] used a new evaluation metric, item novelty, to measure the amount of additional diversity that one item brings to a list of recommendations. Moreover, the loss of accuracy resulting from the increase in diversity can be controlled by changing the granularity of the underlying similarity metrics in diversity-conscious algorithms [19]. These studies argue that one of the goals of recommender systems is to provide a user with highly idiosyncratic or personalized items, and more diverse recommendations result in more opportunities for such items to be recommended to users.

In contrast to individual diversity, recent studies [20] have examined the effect of recommender systems on sales diversity by considering the aggregate diversity of recommendations across all users. Several metrics can be used to measure aggregate diversity, including the percentage of items for which the recommender system is able to make recommendations (often known as coverage) [21]. Adomavicius et al. propose a re-ranking model based on collaborative filtering with percentage adjustment. The levels of accuracy and diversity can be controlled. It can provide significant improvements in recommendation diversity with only a small degree of accuracy loss and also can be used in conjunction with different rating prediction algorithms (i.e., they do not require the designer to use a specific algorithm) [3].Another representative method is to use an optimization method such as that proposed by Ashkan et al of Yahoo Labs who found that use of the information retrieval method to obtain the maximum marginal similarity threshold can yield higher accuracy and diversity in the recommendation [22]. Zhou et al. use a hybrid algorithm to solve the diversity of recommendations problem [23]. Kapoor et al. try to improve diversity by adding novel items to the recommendation list [24]. Puthiya Parambath, Usunier, and Grandvalet use unrated products for the diversity recommendations in order to maximize the coverage of products [25]. Narayana et al use techniques such as item popularity-based ranking, reverse predicted rating value, item average rating, item absolute likeability and item relative likeability that can also generate recommendations that have substantially higher

IJDACR International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

aggregate diversity across all users [26-27].

There are recent efforts to bring principles of economics and psychology into e-commerce recommendation systems, such as Liu Jianguo's study on the recommendation accuracy in regard to consumer taste [28]. In [29], a mechanism is developed to estimate consumers' willingness-to-pay (WTP) in E-commerce settings, and the estimated WTP is used to price products at the individual level so that the seller's profit is maximized. In [30-31], a total surplus- based recommendation framework is proposed to match producers and consumers so that the total benefit is maximized and repeat purchases occur. More closely related to our paper, some studies on basket recommendation problem model the sequential pattern of user purchases and recommend a set of items for the user's next visit based on previous purchases. They provide a series of methods for basket recommendation [32-35], of which the Hierarchical Representation Model (HRM) [35] is the state of the art, but they assume that the value/utility of a product is constant over time and do not focus on the diversity recommendation.

As noted above, although a number of works look to improve diversity, they do not examine perishable and durable products for diversity recommendation from the perspective of the utility of the product. According to the Law of Diminishing Marginal Utility, different products have different decreasing marginal utilities as the purchase count increases. Therefore, we develop algorithmic techniques using marginal utility to improve the individual diversity of recommendations, inspired by the work of Wang et al [10] and construct a Binary Quadratic Programming (BQP) model to maximize the cumulative marginal utility of the recommendation list for the user.

3 Proposed method

Our proposed recommendation process involves three stages: estimating the marginal utility of product *i* for user *u* at time *t* with the basic accuracy algorithm (we choose *SVD*), adaptively selecting the candidate set accurate and diverse products through our proposed method according to the marginal utility, and incorporating them into a constrained Binary Quadratic Programming model and solving it to form the final recommendation list. In this section, we present the details of our recommender system and show how the recommendations are generated.

3.1 Marginal Utility

The Law of Diminishing Marginal Utility suggests that the marginal utility of a product drops as the consumption of the product increases [36]. For example, for user *u*, the utility of consuming the first iPhone 7 might be 5, the utility of consuming the second might be 2, and the utility of consuming a third might be only 1.The Linear utility function is one of the simplest functions in consumer behavior theory, as shown in Equation (1):

$$U(X) = \sum_j a_j x_j \quad (1)$$

where X is the set of products the user consumed and x_j is the consumption quantity of product j. a_j is the basic utility of product j, indicate the purchasing intention for product j. *U(X)* is the utility of the entire purchase list *X*. It can be calculated as marginal utility $\Delta U(X,i)$ of purchasing one additional unit of product i in the following equation (2):

$$\Delta U(X,i) = U(X,i) - U(X) = a_i \quad (2)$$

IJDACR International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

where $U(X,i) = \sum_{j:j\neq i} a_j x_j + a_j x_i'$ is the utility of the entire purchase list with one additional product i. $x_i' = x_j + 1$ is the updated quantity of product i. The summary marginal utility is calculated as follows:

$$U(X) = \sum_j a_j \qquad (3)$$

However, the linear utility does not capture the diminishing return characteristic. The previous purchase count x_i of product *i* does not affect the marginal utility of purchasing one additional unit of product *i*. The Cobb-Douglas utility function [36] is widely used due to its attractive mathematical characteristic: its ability to model the diminishing marginal return. The functional form is

$$U(X) = \sum_j a_j log(x_j) \qquad (4)$$

where the definitions of x_j and a_j are the same as before. The marginal utility of purchasing one additional unit of product *i* is

$$\Delta U(X,i) = U(X,i) - U(X) = a_i(log(x_i') - log(x_i)) = a_i(log(x_i+1) - log(x_i)) \qquad (5)$$

The above equation shows that the marginal utility of product *i* decreases as the consumption quantity of product *i* increases. The diminishing rate is $log(x_i+1) - log(x_i)$.

Following the analysis of Wang [10], Equation (5) shows two major drawbacks of the Cobb-Douglas utility function. It does not differentiate two types of products: products that the user would not purchase often versus products that the user would purchase again and again. Second, the basic utility a_i of a product *i* does not depend on the particular user and is thus at odds with the goal of a personalized recommender system. Motivated by the constant elasticity of substitution (CES) utility function in Equation (6) [37] below, a real-world e-commerce system could collect a large amount of consumer purchase data.

$$U(X) = \sum_j a_j {x_j}^{\lambda} \qquad (6)$$

It can therefore learn the product-specific diminishing rate and the user-specific basic utility from the data. A new marginal utility function is used, as follows:

$$\Delta U_{u,i}(X,i) = a_{u,i}((x_{u,i,t}+1)^{\lambda_i} - (x_{u,i,t})^{\lambda_i}) \quad (7)$$

where $x_{u,i,t}$ is user *u*'s consumption quantity of product *j* by time *t*.

The above Equation (7) includes λ_i , $x_{u,i,t}$ and the parameters of function $a_{u,i}$. The $a_{u,i}$ is discussed in the next section. First, to determine$x_{u,i,t}$, we define the purchase count as the number of purchase orders that the current user has made. Each order is counted once for the same product, regardless of the product quantity in the order. For example, if the user purchases four window panels in one order, the purchase count of window panel is one. Second, it is assumed that the marginal utility is affected not only by previous purchase(s) of the same product but also by previous purchase(s) of similar products. For example, the previous purchase of the iPhone 5 affects the marginal utility of the current purchase of iPhone 6. It can find products that are similar to product *i* based on metadata such as the product title. Then, let $x_{u,i,t}$ be the total similarity between these similar products and the current product *i* for user *u* at time *t*:

$$x_{u,i,t} = \sum_{j:sim(i,j)\geq\theta} C_{u,i,t} \times sim(i,j) \qquad (8)$$

where $C_{u,i,t}$ is user *u*'s purchase count of product *j* by time *t*, and $sim(i,j)$ is the similarity between product *i* and product *j*. θ is a similarity threshold, which will be estimated by cross-validation in the experiments. According to the Constant Elasticity of Substitution (CES) in Equation (7), where λ_i is a parameter to tune the diminishing return rate, it is product-specific and can be learned based on the purchase history of each product.

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

3.2 Recommendation model

The existing recommendation algorithm can be viewed as a function $f(u,i)$ to estimate the value of product *i* for user *u* without considering the diminishing return. To reflect a user's true decision behavior in reality, as mentioned in [10], we can model $\upsilon_{u,i,t}$ as the marginal utility of product *i* for user *u* at time *t*, as follows:

$$\upsilon_{u,i,t} = f(u,i)\left(\left(x_{u,i,t}+1\right)^{\gamma_i} - \left(x_{u,i,t}\right)^{\gamma_i}\right) \quad (9)$$

Whether product i is likely to be purchased is dependent on the product's basic utility, its diminishing rate, and the product's price, and is not considered in our model for convenience. The difference between Equation (9) and Equation (7) is that $a_{u,i}$ is replaced by $f(u,i)$ to estimate $a_{u,i}$, the basic utility of product *i* for user u. At each decision point *t*, a higher marginal utility $\upsilon_{u,i,t}$ indicates that user u is more likely to purchase product i.The following logistic function can be used to capture this intuition and model the conditional probability of making the purchase:

$$\Pr\left(\mathrm{r}_{u,i,t} \middle| \upsilon_{u,i,t}\right) = \frac{1}{1+e^{-\mathrm{r}_{u,i,t} * \upsilon_{u,i,t}}} \quad (10)$$

where $r_{u,i,t}$= 1 if user u purchases i at time *t*. Otherwise $\mathrm{r}_{u,i,t} = -1$.

The parameters of Equation (10) include γ_i and parameters of function f. To learn these parameters, we can order the entire user purchase history by the purchase time. At each time point *t*, user u's purchase decision of product i is considered to be training point. If user u purchased the product, it is a positive training point with $\mathrm{r}_{u,i,t}$= 1. Otherwise, it is a negative training point with $\mathrm{r}_{u,i,t} = -1$.When a new recommendation list is needed, the system can estimate each product's marginal utility based on Equation (9) and rank them accordingly. The system can predict the likelihood that a user will purchase an item using Equation (10) and decide whether to recommend an item to the user accordingly.

To diversify recommendations, we should maintain the accuracy first, and then construct a constrained integer programming model to adjust the diversity and accuracy. We choose *SVD* as a basic algorithm to estimate $\upsilon_{u,i,t}$ because it is a popular recommendation algorithm with decent performance. The basic *SVD* algorithm operates over a user-product matrix $R_{M\times N}$. It assumes that each entry $R_{u,i}$ in the matrix R can be estimated using the form [38-39]:

$$R_{u,i} = q_i^T p_u \quad (11)$$

Where q_i and p_u are vectors, which are the hidden representations of product i and user *u*. These vectors can be estimated using all given entries in $R_{M\times N}$. The value of $R_{u,i}$ in the observed matrix $R_{M\times N}$ is determined by the user purchase history. It is user *u*'s rating of product i. When a recommender system ranks all products by their estimated $R_{u,i}$ values and selects the top ones to recommend, it is equivalent to maximizing the linear utility. Set the basic utility $a_{u,i} = f(u,i) = q_i^T p_u$. Based on Equation (9), the marginal utility is

$$\upsilon_{u,i,t} = q_i^T p_u\left(\left(x_{u,i,t}+1\right)^{\lambda_i} - \left(x_{u,i,t}\right)^{\lambda_i}\right) \quad (12)$$

For simplicity, and as is common in the literature, we assume that prior distributions of user vector p_u and item vector q_i are Gaussian distributions. Pr (p_u) and Pr (q_i) have mean zero and variance $1/\lambda_1$. We assume that the prior distribution of λ_i is a Gaussian distribution with mean λ_0 and variance $1/\lambda_2$. We treat each purchase order made by a user as a decision point. The purchase history of all users can be viewed as the training data= $(r_{u,i,t}$, u , $t)$. The

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

joint probability (likelihood) of all parameters and the training data is

$$L = \prod_u \Pr(p_u) \prod_i \Pr(q_i) \prod_i \lambda_i \prod_i \Pr(r_{u,i,t} | \upsilon_{u,i,t}) \quad (13)$$

The model parameters can be found by maximizing the joint probability of all parameters and the training data. According to Equation (13) and Equation (10), this is equivalent to minimizing the negative log likelihood of the data as follows:

$$(p_u, q_i, \lambda_i) = \operatorname{argmin}(-\log(L)) = \operatorname{argmin} \frac{1}{2}\lambda_1 \sum_u \| p_u \|^2 + \frac{1}{2}\lambda_1 \sum_i \| q_i \|^2 + \frac{1}{2}\lambda_1 \sum_i (\gamma_1 - \gamma_0)^2 + \sum_{u,i,t} \log(1 + e^{-r_{u,i,t} * \upsilon_{u,i,t}}) \quad (14)$$

where λ_* can also be viewed as regularization factors to avoid the overfitting problem. The first-order derivatives are

$$\frac{\partial(-logL)}{\partial(q_i)} = \lambda_1 q_i + g_{u,i,t} \cdot [p_u \cdot d_{u,i,t}]$$

$$\frac{\partial(-logL)}{\partial(p_u)} = \lambda_1 p_u + g_{u,i,t} \cdot [q_i \cdot d_{u,i,t}]$$

$$\frac{\partial(-logL)}{\partial(\gamma_i)} = \lambda_2 \gamma_i + g_{u,i,t} \cdot [q_i^T p_u \cdot ((x_{u,i,t}+1)^{\gamma_i} \cdot \log(x_{u,i,t}+1) - (x_{u,i,t})^{\gamma_i} \cdot \log(x_{u,i,t}))]$$

where

$$d_{u,i,t} = (x_{u,i,t}+1)^{\gamma_i} - (x_{u,i,t})^{\gamma_i}$$

$$\upsilon_{u,i,t} = q_i^T p_u d_{u,i,t}$$

$$g_{u,i,t} = \frac{\partial(\log(1 + e^{-r_{u,i,t} * \upsilon_{u,i,t}}))}{\partial(\upsilon_{u,i,t})} = \frac{e^{-r_{u,i,t} * \upsilon_{u,i,t}}}{1 + e^{-r_{u,i,t} * \upsilon_{u,i,t}}} \cdot (-r_{u,i,t})$$

We then use the stochastic gradient descent method to find the optimal parameters. Following the standard stochastic gradient descent method, update rules at each iteration are shown in the following equations. The algorithm stops when the change in an iteration is small enough:

$$p_u = p_u - \beta_1 \frac{\partial(-logL)}{\partial(p_u)}$$

$$q_i = q_i - \beta_1 \frac{\partial(-logL)}{\partial(q_i)}$$

$$\gamma_i = \gamma_i - \beta_2 \frac{\partial(-logL)}{\partial(\gamma_i)}$$

where β_* controls the learning rate at each iteration. β_* and γ_* can be set by the cross-validation.

Based on the above derivation, after capturing the utility value of each product, we can use $\upsilon_{u,i,t}$ to support the user's decision making. In particular, each user u is recommended a list of top-N items using the following as the ranking criterion:

$$rank_{\text{accuracy}}(i) = \upsilon_{u,i,t}^{-1} \quad (15)$$

The power of -1 in the above expression indicates that the items with the highest marginal utility are recommended to the candidate set for the user. In this paper, we refer to this as the accurate ranking approach. By definition, recommending the most highly predicted items selected by the accurate ranking approach is designed to improve recommendation accuracy, but not recommendation diversity. Therefore, new ranking criteria are needed to achieve diversity improvement. As noted in section 2, if the marginal utility of the product is high, it can be recommended using the method of $rank_{\text{accuracy}}(i)$, and when the marginal utility of the product is low, it should be appropriate to recommend more diverse products for the user. Following this motivation, we can utilize the marginal utility to adjust the diversity and the accuracy and propose the diversity recommendation approach combined with the marginal utility, which is defined as:

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

$$rank_x(i, \upsilon_{u,i,t}) = \begin{cases} rank_{ItemPop}(i), \ if \ \upsilon_{u,i,t} \le \varepsilon \\ rank_{\text{accuracy}}(i), if \ \upsilon_{u,i,t} > \varepsilon \end{cases} \quad (16)$$

$$where \ rank_{ItemPop}(i) = |User(i)|, User(i) = \{u \in User | \exists \upsilon_{u,i,t}\}$$

ε is parameterized with a ranking threshold that allows the user to choose a certain level of recommendation accuracy. The value can be set by the cross-validation. Items that are predicted above ε are ranked according to $rank_{\text{accuracy}}(i)$, and items that are below ε are ranked according to the ranking approach $rank_{ItemPop}(i)$. For convenience, with the assumption that recommending less popular items should increase recommendation diversity, we choose the $rank_{ItemPop}(i)$ as the diverse recommendation method, which can be substituted for others. Model (16) can give diverse recommendations when $\upsilon_{u,i,t}$ is small in that the marginal utility of product i for user u in the time of t is small, so it is appropriate to recommend diverse products for the user by $rank_{ItemPop}(i)$ instead of $rank_{\text{accuracy}}(i)$ and vice versa.

After obtaining the candidate set of accuracy and diversity, because the recommendation list length is constrained, we conduct further analysis to select the final product from the candidate set to maximize the cumulative marginal utility of the recommendation list. After capture the utility value of each item is $\upsilon_{u,i,t}$ and the all-item utility in the candidate set is

$$\psi(R, u) = \upsilon_{u,t}{}^T y(u, i, t) \quad (17)$$

$y(u, i, t)$ is a binary value indicating whether product i has been selected or not by user u in the time t. We use the cumulative dissimilarity between the product list defined as diversity [3]. Specifically, the $d(i, j)$ is given to indicate the dissimilarity or distance between products i and j; then, we can model the diversity recommendation of the Top-N as an optimization problem:

$$\max\ [(1-\eta)\ \frac{1}{N(N-1)} y^T Dy + (\eta)(q_i^T p_u)^T y] \quad (18)$$

$$\text{s.t.} \begin{cases} 1^T y = N \\ y(u, i, t) = \{0,1\}, \forall\ i = 1,2, \dots, M \\ \upsilon_{u,i,t}{}^T y \ge \omega \times \psi(R, u)_{max} \end{cases}$$

where $y^T Dy = \sum_{i \in R} \sum_{j \ne i \in R} d(i, j)$, $d(i, j) = \frac{1-s(i,j)}{2}$, $sim(i, j) = \frac{\sum_{i \in I_{i,j}} (r_{u,i} - \overline{r_u})(r_{v,i} - \overline{r_v})}{\sqrt{\sum_{i \in I_{i,j}} (r_{u,i} - \overline{r_u})^2} \sqrt{\sum_{i \in I_{i,j}} (r_{v,i} - \overline{r_v})^2}}$

The parameter $\omega \in [0,1]$ indicates the maximized diversity of the recommended list when the utility value is not less than the maximum value of the ω times. $\eta \in [0,1]$ can adjust the weight of diversity and accuracy. D is a diversity symmetric matrix and implies that the distance of each $(q_i^T p_u)^T$ is a vector ranking according to the *SVD* algorithm ($rank_{\text{accuracy}}(i)$), which represents the candidate set of accuracy. All elements on the main diagonal are zero. The items in the matrix D are selected using the $rank_{ItemPop}(i)$ ranking methods adopted in model (16). The above model is the Binary Quadratic Programming (BQP) problem of formula (18), This paper adopts the Tabu search algorithm[40-41] method for iterative solutions. Because it has the constrained condition, we convert it to an unconstrained Binary Quadratic Programming problem. The constraints are $\psi(R, u) = \upsilon_{u,t}{}^T y$, which can be converted into a penalty function:

$$f_{penalty} = 2^{u(\omega \times \psi(R,u)_{max} - \upsilon_{u,i,t}{}^T y)} - 1 \quad (19)$$

The function $u(.)$ is a step function. This means that it will be punished when the utility value does not satisfy the constraints. The detailed process is as follows:

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

Step1. Choose an initial vector $y_0 = (y_1, y_2, y_3, \dots, y_m)$ as a randomized initial solution, it is assumed that the initial solution can obtain the maximum $\psi(R, u)$.

Step2. Set the optimal solution $y^* = y_0$ and the optimal objective function value $d^* = d(y_0)$, and initialize the Tabu list.

Step3. Calculate the objective function value (or change value) of the neighbor solution of the current solution. The neighbor solution is generated by selecting one of the y_i from the current 1 to flip to 0 and another from the current y_j of 0 to 1. This ensures that all solutions satisfy the constraints equation (19).

Step 4. Successively select the maximal solution of the objective function value from the neighborhood solution set to update the optimal solution and the optimal objective function value. Do this according to the two kinds of updates in the Tabu list:

(a) if the solution is taboo, but the solution is better than the optimal solution of the historical conditions, then make an amnesty of this solution.

(b) if the solution is not taboo, directly update the optimal solution and the optimal objective function value and update the Tabu list.

Step 5. Repeat steps 3 and 4 until the convergence condition is satisfied or the maximum iteration number is reached.

Step 6. Output the y^* as the final solution of the products contained in the recommendation list.

Based on the above discussion, we summarize the ideas behind the proposed approaches as follows. In the first step, we use a traditional recommendation technique such as *SVD* to predict unknown ratings based on the purchase records. In the second step, we use the marginal utility method discussed in section 3 to analyze the product, which has been selected using the traditional recommendation algorithm according to the law of diminishing marginal utility. The third step involves adaptively selecting the candidate set of the accurate and diverse products using the corresponding algorithm according to model (16) together with marginal utility. The fourth step is to construct a constrained Binary Quadratic Programming (BQP) to select the final product from the candidate set to maximize the cumulative marginal utility of the recommendation list.

In the next section, we use real data to show how this approach can improve the accuracy and diversity of recommendations.

4. Experiment and empirical results

Because this paper focuses on recommendations in e-commerce sites, we collect a dataset from the well-known Chinese e-commerce website Alibaba (the TianChi Dataset[1]) to evaluate the performance of our approach. We compare the classic collaborative filtering algorithms denoted as CB [42], SVD , SVD_{util} ,a hybrid recommendation algorithm denoted as $Hybrid$ [43], and the re-ranking approaches provided by Adomavicius et al, denoted as $Re\_ranking$ [3], and our basic method. The dataset is obtained from TianChi Big Data Race Project and includes taobao.com and Tmall online actual consumer behavior data, which is used to test the recommendation algorithm. The data are from May to October 2014 and include "double 11" purchase records, including the user ID, product ID, product category, brand ID, order ID, repeat purchase (multiple purchase, one purchase, click-to-browse), and date of purchase. The total count of this dataset is nearly one million. Data

[1] https://tianchi.aliyun.com/

IJDACR
ISSN: 2319-4863

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

descriptions are provided in Tables 1–3.

Table 1 Descriptions of consumer data

Data Field	Definition
User ID	A unique ID for the shopper.
Age range	1 for <18; 2 for 18–24; 3 for 25–29; 4 for 30–34; 5 for 35–39; 6 for 40–49; 7 and 8 for ≥ 50; 0 and NULL for unknown.
Gender	0 for female, 1 for male, 2 and NULL for unknown.

Table 2 Descriptions of repeat buyer data

Data Field	Definition
User ID	A unique ID for the shopper.
Merchant ID	A unique ID for the merchant.
Label	Enumerated {0, 1}, where 1 signifies repeat buyer, 0 signifies non-repeat buyer. This field is empty for test data.

Table 3 Descriptions of transaction data

Data Field	Definition
User ID	A unique ID for the shopper
Item ID	A unique ID for the item.
Cat ID	A unique ID for the category that the item belongs to.
Merchant ID	A unique ID for the merchant.
Brand ID	A unique ID for the brand of the item.
Time stamp	Date the action took place (format: mmdd)
Action type	Enumerated {0, 1, 2, 3}, where 0 is for click, 1 is for add-to-cart, 2 is for purchase and 3 is for add-to-favorite.

4.1 Data pre-processing

We randomly select 100,000 consumer purchase records from the total records and split them by month according to the time, using the SVML functions by 10-fold cross-validation with R language. The data are divided into training data and testing data at a ratio of 8:2. Tail users that made less than five unique product purchases are filtered out from the training data. The remaining training data contains 10,399 users and 65,551 products. There are 92,915 unique (user, product) pairs. As we can see, the user-product matrix for *SVD* is quite sparse, with only 0.0055 density. Each value was obtained using an average of 10 experiments. The descriptive statistics are shown in Table 4.

Table 4 Descriptive statistics (mean value)

	Training data	Testing data
Consumers (n)	8000	2000
Products (n)	7,1754	2,1362
Maximum number of consumer purchases	14 (total 28)	24 (total 14)
Minimum number of consumer purchases	1 (total 19)	1 (total 2)
Average number of consumer purchases	9	11
Density	0.0055	0.0047

4.2 Evaluation metrics and performance of proposed method

After splitting the training data, we use the last 10% of the training data as the validation data to set (β_*,γ_*, θ). We have β_1= 0.015， β_2= 0.035， γ_1= 0.05, γ_2= 0.01, $\varepsilon = 2$ for positive training points, γ_2= 0.001 for negative training points, γ_0= 1, and θ = 0.7, rather than using all negative training points. The sample percentage is determined by cross-validation. Both positive and negative training data are used to learn model parameters. Finally the model is used to generate the recommendation list in the testing stage. For every decision point *t* in the testing data, all products are ranked by the marginal

IJDACR
International Journal Of Digital Application & Contemporary Research
International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

utility $\upsilon_{u,i,t}$ (Equation 12). The solution of the model (18) represents the products that are recommended to the user.

Conversion rate, a commonly-used metric in e-commerce, is used as an additional evaluation metric in our experiments [44]. If the user purchased at least one product from the recommended top *K* list, we consider that the user has converted from a browser into a buyer. The calculation of conversion rate for one testing point is shown in the following equation (20). Each order D_u corresponds to a testing point, which is uniquely identified by a (user, order time) pair. Let $S_{recommend}$ be the set that contains all products in this purchase order. Let S_{buy} be the set that contains the top *K* products using formula (20) to measure conversion rate *CR*:

$$CR = \begin{cases} 1, S_{buy} \cap S_{recommend} = \emptyset \\ 0, others \end{cases} \quad (20)$$

Conversion rate reflects whether a user receives at least one good recommendation. The average value of all testing points will be used to compare different algorithms. Statistical significance tests are used when comparing two methods.

The conversion rate performance of all methods is shown in Table 5. In the evaluation step, Table 5 reports the performance for returning users that purchased from 1) all products and 2) $products_{n>0}$, which represents the returning products with at least one purchased in the training data. It is clear that all personalized methods are significantly better than the method *CB*. Although the precision and recall are not reported here, we obtain similar observations when evaluating with these two metrics. Comparing SVD_{utilty} with *SVD* matrix (just be *SVD*), we can see that our proposed marginal utility function provides an improvement. The proposed method is significantly better than the *SVD* matrix for all evaluation metrics.

Table 5: Conversion rate CR for the recommendation task. Value* is significantly better than the other method.

Method	K = 5	K = 10	K = 15	K = 20	K = 25	K = 30
			for all products			
CB	0.0014	0.0036	0.0053	0.0056	0.0056	0.0079
SVD	0.0121	0.0352	0.0390	0.0445	0.0562	0.0618
SVD_{utilty}	0.0200	0.0360	0.0384	0.0467	0.0531	0.0648
Re_ranking	0.0310	0.0290	0.0374	0.0491	0.0556	0.0656
Proposed	**0.0360***	**0.0320***	**0.0429***	**0.4460***	**0.0560***	**0.0691***
			for $products_{n>0}$			
CB	0.0027	0.0062	0.0092	0.0148	0.0153	0.0209
SVD	0.0930	0.1322	0.1270	0.1428	0.1512	0.1626
SVD_{utilty}	0.1200	0.1460	0.1327	0.1670	0.1628	0.1743
Re-ranking	0.0510	0.0870	0.1366	0.1427	0.1592	0.1655
Proposed	**0.1931***	**0.2400***	**0.2121***	**0.2730***	**0.2246***	**0.2672***

In the previous section, we generated a recommended list that included both re-purchase products and new products. In addition to the general recommendation task, e-commerce sites also have more specific tasks. We now perform further analysis to compare these recommendation algorithms in three different recommendation tasks. The first task is to

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

recommend products for a user to re-purchase. The second is to recommend new but similar that a user has never purchased before. The third is to recommend new products (that a user has never purchased) that are different from those that the user has purchased before. The conversion rate for the re-purchase, new product recommendation task is shown in Table 6.

Table 6: Conversion rate CR for the three recommendation tasks. Value* is significantly better than the other method.

Method	K = 2	K = 4	K = 6	K = 8	K = 10	K = 12
for all products (19.63% orders contain re-purchase)						
CB	0	0.0016	0.0024	0.0002	0.0024.	0.0024
SVD	0.0053	0.0054	0.0060	0.0060	0.0067	0.0067
SVD_{utilty}	0.0176	0.0187	0.0180	0.0188	0.0190	0.0190
Re_ranking	0	0	0	0	0	0.0001
Proposed	**0.0349**	**0.0205***	**0.0319***	**0.0316***	**0.0320***	**0.0322***
for all products (69.29% orders contain new purchase with similar)						
CB	0	0.2316	0.2576	0.2483	0.2761.	0.2794
SVD	0.1423	0.1564	0.1862	0.1775	0.2483	0.2671
SVD_{utilty}	0.1536	0.2821	0.2946	0.2957	0.316	0.3246
Re-ranking	0	0.2946	0.2856	0.3128	0.3047	0.3372
Proposed	**0.2359**	**0.3085***	**0.3247***	**0.3197***	**0.3265***	**0.3394***
for all products (12.18% orders contain new purchase with diversity)						
CB	0	0	0	0	0	0.0002
SVD	0	0	0	0.0006	0.0006	0.0006
SVD_{utilty}	0.0007	0.0001	0.0025	0.0026	0.0036	0.0026
Re-ranking	0.0002	0.0002	0.0006	0.0006	0.0007	0.0007
Proposed	**0.0027**	**0.0152***	**0.0209***	**0.0236***	**0.0352***	**0.0379***

In addition to the conversion rate metrics, we also adopt commonly used metrics (area under the ROC curve (AUC), coverage, diversity, and precision) to evaluate the approaches. We use formula (21), formula (22), and the diversity value calculation formula (23) to measure the performance of the approach. The precision is used to measure the accuracy of the recommendation algorithm, the coverage rate is used to measure its overall diversity, and the diversity value is used to measure individual diversity. The AUC is calculated using formula (24) [45].

As the dataset does not provide users with explicit scoring information, we convert the consumer behavior data based on the user's implicit feedback (purchase, click and browse information) to explicit scoring information [46]. Let *u* be a set for the consumers and *N* be a set for the products. $s(x,y) \in [0,1]$ indicates the similarity between *x* and *y* (the product similarity calculation according to the distance of the category tree), *R(u)* indicates the recommendation list of the training set, and *T(u)* indicates the recommendation list of the testing set. Each value is obtained using the average of 10 experiments.

$$Precision = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|} \quad (21)$$

$$Coverage = \frac{|R(u)|}{|N|} \quad (22)$$

IJDACR
ISSN: 2319-4863
IJDACR
International Journal Of Digital Application & Contemporary Research
International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

$$Diversity = \frac{\sum_{u \in U} Diversity(R(u))}{|U|} \quad , \quad \text{where}$$

$$Diversity(R(u)) = 1 - \frac{\sum_{i,j \in R(u), i \neq j} s(i,j)}{\frac{1}{2}|R(u)|(|R(u)|-1)} \quad (23)$$

$$AUC = \frac{S - S_1(S_1+1)/2}{S_1 \times S_2} \quad (24)$$

S_1 and S_2 represent the numbers of relevant items and irrelevant items respectively. Additionally, $S = \sum_{u \in U} R_j$, where R_j is the rank of the jth relevant items in the ranked list for user u_i.

Figures 10–13 show how the proposed method compares with collaborative filtering, *SVD*, re-ranking algorithm recommendation, and hybrid algorithm recommendation in terms of the precision, diversity, and coverage of experimental results. Considering the utility of the product leads to clear improvements in precision, coverage, and diversity associated with the proposed method. The collaborative filtering recommendation algorithm considers precision alone, and with an increase in the number of consumers and products, the precision decreases, and the coverage and diversity declines markedly in comparison with the other approaches. The hybrid algorithm uses a waterfall model to generate recommendations, using the previous results as the input for the second step to finally obtain a highly accurate result. As the recommendation list increases in size, the precision of the result becomes closer to our results, as shown in Figure 10, but the diversity and coverage are nearly 5% lower than our proposed method due to filtering restrictions, as shown in Figures 11 and 12.

The AUC for all of the data were calculated using the average score over all users in the test set. The higher the value of the metric AUC, the better the system, as shown in Figure 13.

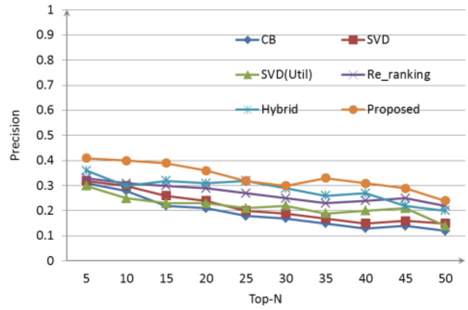

Figure 10. Compared with the others in accuracy

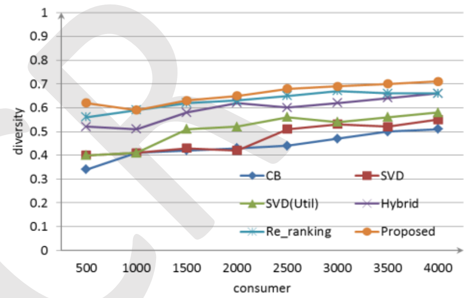

Figure11. Compared with the others in diversity

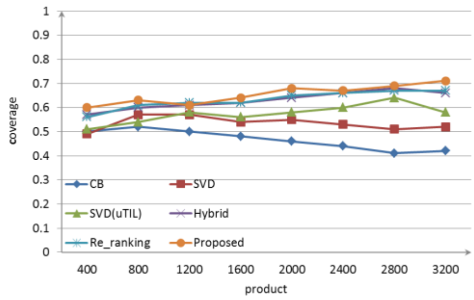

Figure12. Compared with the others in coverage

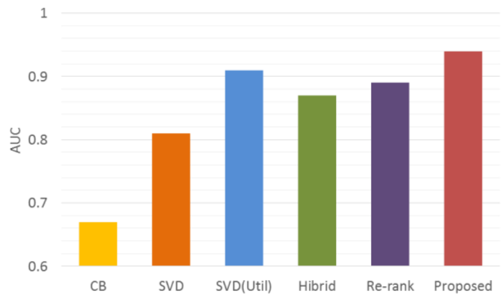

Figure 13. The AUC comparison with other approaches

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

Figure 14 shows the accuracy comparison of the recommendation results of the four main algorithms under different ω values.When the ω decreases gradually, the accuracy of the proposed method also gradually decreases to a certain extent. At the same time, when the ω value is close to 0, the decrease is not obvious or is even slightly higher, indicating that the higher utility value had little effect on the recommendation accuracy. It is reasonable that users prefer the utility value, which is equally highly ranked as accuracy. Compare this with other contrast algorithms, which do not consider the utility: in their cases, ω changes without impact. Figure 15 shows the diversity comparison of the recommendation results with the same four main algorithms under different ω values. As we can see, the CB algorithm has the lowest diversity and displays no impact under different ω values. The proposed method shows a rapid upward trend when the ω value decreases from 1 to 0.5 and then becomes stable. It fully illustrates that the proposed method is able to obtain the diversity recommendation under the condition of certain accuracy.

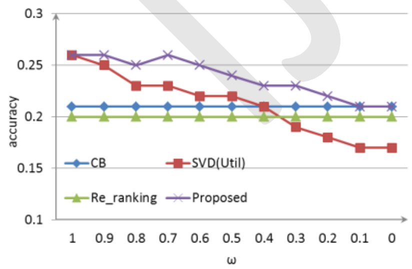

Figure.14 Compare with others under different ω in accuracy

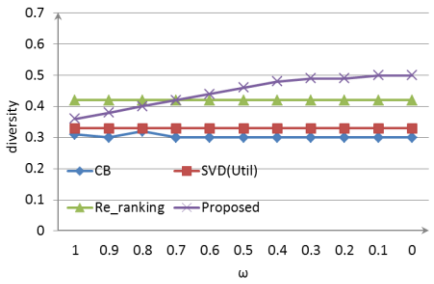

Figure.15 Compare with others under different ω in diversity

5 Conclusion

There has been significant progress in recommender systems in recent years, and there are many proposed techniques to improve diversity recommendation quality. However, the existing recommendation algorithms usually diversify the results with the assumption that the utility of a product is the same and constant over time for a user. Economists commonly use utility to characterize consumer preference over products, and it serves as a corner stone for consumer choice theory. We therefore propose a method for adaptive diversity recommendation according to the durable goods and perishable products in the user's transactions. The method can select the accurate algorithm to make recommendations when the previous purchase products are perishable and it can select the diverse algorithm to make recommendations when the previous purchase products are durable. Because the length of the recommendation list is limited, we also construct a constrained Binary Quadratic Programming (BQP) model to maximize the cumulative utility of the recommendation list for user. The model parameters are developed using existing consumer data. Experimental results on tmall.com e-commerce data sets demonstrate the effectiveness of the proposed approach for recommendations. The results show that this

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

method can guarantee accuracy, meet the requirement for recommendation diversity, and improve the total marginal utility of the recommendation list.

This study considers the diversity of recommendations based on marginal utility. Although our experiments are about diversity recommendation, the proposed framework could be applied to other usage scenarios in the future. The marginal utility diminishing rate depends not only on each product but also on the relationships between products, which are dynamic and complex. There is thus a need to comprehensively model the relationships between products for various recommendation tasks, such as package recommendation and basket recommendation. In the future, the relationships between products could be incorporated to reflect real-world scenarios, which could also be utilized to improve our model further.

6. Reference

[1] Zhang F. Research on Recommendation List Diversity of Recommender Systems[C]// International Conference on Management of E-Commerce and E-Government. IEEE, 2008:72-76.

[2] Ge M, Jannach D, Gedikli F. Bringing Diversity to Recommendation Lists – An Analysis of the Placement of Diverse Items[J]. *Lecture Notes in Business Information Processing*, 2012, 141:293-305.

[3] Zhang M, Hurley N. Avoiding monotony: improving the diversity of recommendation lists [C].*Proceedings of the 2008 ACM conference on Recommender systems ACM*, 2008, 123-130.

[4] Adomavicius G, Kwon Y O. Improving aggregate recommendation diversity using ranking-based techniques [J]. *Knowledge & Data Engineering IEEE Transactions*, 2012, 24(5):896-911.

[5] Guo J, Gao Z, Liu N, et al. Recommend products with consideration of multi-category inter-purchase time and price [J]. Future Generation Computer Systems, 2017.

[6] Ferwerda, B., Graus, M. P., Vall, A. How Item Discovery Enabled by Diversity Leads to Increased Recommendation List Attractiveness [C]. *Symposium on Applied Computing*, 2017, 1693-1696.

[7] Javari A, Jalili M. A probabilistic model to resolve diversity–accuracy challenge of recommendation systems [J]. Knowledge and Information Systems, 2014: 1-19.

[8] Xiao B, Benbasat I. E-commerce product recommendation agents: use, characteristics, and impact.[J]. Management Information Systems Quarterly, 2007, 31(1):137-209.

[9] Chen, Li, Wen Wu, and Liang He. How personality influences users' needs for recommendation diversity? [C]. *human factors in computing systems ACM*, 2013, 829-834.

[10] Kunaver M, Požrl T. Diversity in recommender systems – A survey [J]. Knowledge-Based Systems, 2017, 123:154-162.

[11] Wang J, Zhang Y. Utilizing marginal net utility for recommendation in e-commerce[C]// International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2011:1003-1012.

[12] Balabanovic M. Content-based, collaborative recommendation [J]. Communications of the Acm, 1997, 40(3):66-72.

[13] Burke, Robin D. Hybrid Recommender Systems: Survey and Experiments [C]. *User Modeling and User-adapted Interaction*, 2002, 12 (4):331-370.

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

[14] Bobadilla J, Ortega F, Hernando A. Recommender systems survey [J]. Knowledge-Based Systems, 2013, 46(1):109-132.

[15] Bennett, Paul N. Overview of the Special Issue on Contextual Search and Recommendation [J]. *ACM Transactions on Information Systems*, 2015, 33 (1):1-7.

[16] Sharif, Mohammad Amir, and V. V. Raghavan. Link prediction based hybrid recommendation system using user-page preference graphs [C]. *International Joint Conference on Neural Networks IEEE*, 2017, 1147-1154.

[17] Portugal I, Alencar P, Cowan D. The use of machine learning algorithms in recommender systems: A systematic review [J]. Expert Systems with Applications, 2018, 97:205-227.

[18] Bradley, Keith. Improving Recommendation Diversity [M]. 2001, 1-10.

[19] Ziegler, Cainicolas. Improving recommendation lists through topic diversification [C]. *International World Wide Web conferences*, 2005, 22-32.

[20] Mcsherry D. Diversity-Conscious Retrieval[C]// European Conference on Advances in Case-based Reasoning. Springer Berlin Heidelberg, 2002:219-233.

[21] Brynjolfsson E, Hu Y, Simester D. Goodbye Pareto Principle, Hello Long Tail: The Effect of Search Costs on the Concentration of Product Sales [J]. Management Science, 2011, 57(8):1373-1386.

[22] Gunawardana A, Shani G. Evaluating Recommender Systems[C]// International Conference on Automated Solutions for Cross Media Content and Multi-Channel Distribution. IEEE Computer Society, 2015:211-217.

[23] Ashkan, Azin. DUM: Diversity-Weighted Utility Maximization for Recommendations [C]. *Information Retrieval*, 2014, 1-12.

[24] Zhou, Tao. Solving the apparent diversity-accuracy dilemma of recommender systems [C]. *Proceedings of the National Academy of Sciences of the United States of America*, 2010, 107 (10): 4511-4515.

[25] Kapoor, K., Kumar, V., Terveen, L. I Like to Explore Sometimes: Adapting to Dynamic User Novelty Preferences [C]. *Proceedings of the 9th ACM Conference on Recommender Systems*, 2015, 19-26.

[26] Parambath, Shameem A Puthiya, Nicolas Usunier, and Yves Grandvalet. A Coverage-Based Approach to Recommendation Diversity on Similarity Graph [C].*Conference on recommender systems*, 2016, 15-22.

[27] Fabiano M. Belém. Beyond Relevance: Explicitly Promoting Novelty and Diversity in Tag Recommendation [J]. *Acm Transactions on Intelligent Systems & Technology*, 2016, **7**(3): 1-34.

[28] R., Arun Kumar, B., and T. V. Narayana Rao. Ranking Based Techniques Using Optimised Recommendation Diversity [J]. *International Journal of Computer Science Engineering & Technology*, 2016, **6**(11):387-391.

[29] JianGuo Liu, Lei Hou, Yi-Lu Zhang, et al. Empirical analysis of the clustering coefficient in the user-object bipartite networks [J]. International Journal of Modern Physics C, 2013, 24(8):2917-2933.

[30] Zhang Y, Zhao Q, Zhang Y, et al. Economic Recommendation with Surplus Maximization[C]// International Conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2016:73-83.

[31] Zhao Q, Zhang Y, Friedman D, et al. E-commerce Recommendation with

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary Research
Website: www.ijdacr.com (Volume 6, Issue 10, May 2018)

Personalized Promotion[C]// ACM Conference on Recommender Systems. ACM, 2015:219-226.

[32] Tian Y, Ye Z, Yan Y, et al. A practical model to predict the repeat purchasing pattern of consumers in the C2C e-commerce [J]. Electronic Commerce Research, 2015, 15(4):571-583.

[33] Gatzioura A, Sanchezmarre M. A Case-Based Recommendation Approach for Market Basket Data [J]. IEEE Intelligent Systems, 2015, 30(1):20-27.

[34] Wang P, Guo J, Lan Y. Modeling Retail Transaction Data for Personalized Shopping Recommendation [J]. 2014:1979-1982.

[35] Rendle S, Freudenthaler C, Schmidt-Thieme L. Factorizing personalized Markov chains for next-basket recommendation[C]// International Conference on World Wide Web. ACM, 2010:811-820.

[36] Wang P, Guo J, Lan Y, et al. Learning Hierarchical Representation Model for Next-Basket Recommendation[C]// International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2015:403-412.

[37] Cobb C W, Douglas P H. A Theory of Production [J]. American Economic Review, 1928, 18(1):139-165.

[38] Uzawa H. Production Functions with Constant Elasticities of Substitution [J]. Review of Economic Studies, 1962, 29(4):291-299.

[39] Cremonesi P, Koren Y, Turrin R. Performance of recommender algorithms on top-n recommendation tasks[C]// ACM Conference on Recommender Systems. ACM, 2010:39-46.

[40] Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2008:426-434.

[41] Beasley J E. Heuristic Algorithms for the Unconstrained Binary Quadratic Programming Problem[J]. 1998.

[42] Gendreau M, Iori M, Laporte G, et al. A Tabu search heuristic for the vehicle routing problem with two-dimensional loading constraints [J]. Networks, 2008, 51(2):469-477.

[43] Ren, C., Zhu, P., Zhang, H. A new Collaborative Filtering technique to improve recommendation diversity [C]. *IEEE International Conference on Computer and Communications*, 2017, 147-169.

[44] Albadvi A, Shahbazi M. A hybrid recommendation technique based on product category attributes [J]. Expert Systems with Applications, 2009, 36(9):11480-11488.

[45] Cremonesi P, Koren Y, Turrin R. Performance of recommender algorithms on top-n recommendation tasks[C]// ACM Conference on Recommender Systems. ACM, 2010:39-46.

[46] Shani G, Gunawardana A. Evaluating Recommendation Systems [J]. Recommender Systems Handbook, 2016:257-297.

[47] Hu Y，Koren Y，Volinsky C. Collaborative filtering for implicit feedback datasets[C].Data Mining, 2008. ICDM 08. Eighth IEEE International Conference on. IEEE, 2008:263-272.