# Recommender System Combining Popularity and Novelty Based on One-Mode Projection of Weighted Bipartite Network

**Yong Yu[1], Yongjun Luo[1], Tong Li[2], Shudong Li[3, *], Xiaobo Wu[4], Jinzhuo Liu[1, *] and Yu Jiang[3, *]**

**Abstract:** Personalized recommendation algorithms, which are effective means to solve information overload, are popular topics in current research. In this paper, a recommender system combining popularity and novelty (RSCPN) based on one-mode projection of weighted bipartite network is proposed. The edge between a user and item is weighted with the item's rating, and we consider the difference in the ratings of different users for an item to obtain a reasonable method of measuring the similarity between users. RSCPN can be used in the same model for popularity and novelty recommendation by setting different parameter values and analyzing how a change in parameters affects the popularity and novelty of the recommender system. We verify and compare the accuracy, diversity and novelty of the proposed model with those of other models, and results show that RSCPN is feasible.

**Keywords:** Personalized recommendation, one-mode projection, weighted bipartite network, novelty recommendation, diversity.

## 1 Introduction

The explosive growth of web information resources has resulted in large amounts of data and caused serious information overload. How to help users filter large amounts of information accurately is a popular topic of current research. Personalized recommender systems [Ricci, Rokach and Shapira (2011)] are an effective method to solve this problem. By analyzing the behavior of users, these systems can predict the interests of users and recommend information that users may be interested in, thus improving the efficiency of user filtering and reducing the time of information filtering. Current recommender systems include collaborative filtering [Chen, Teng and Chang (2015); Li, Wang, Pan et al. (2019)], content-based [Silvia, Javier, Jordi et al. (2015); Son and Kim (2017)], hybrid

[1] School of Software, Key Laboratory in Software Engineering of Yunnan Province, Yunnan University, Kunming, 650091, China.

[2] School of Big Data, Yunnan Agricultural University, Kunming, 650201, China.

[3] Cyberspace Institute of Advanced Technology, Guangzhou University, Guangzhou, 510006, China.

[4] School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou, 510006, China.

[*] Corresponding Author: Jinzhuo Liu. Email: jinzhuo.liu@hotmail.com; lishudong@gzhu.edu.cn.

Received: 12 June 2019; Accepted: 16 July 2019.

[Wu, Yue, Pei et al. (2016); Gan (2016)] and network-based [Zhou, Ren, Medo et al. (2007)] systems, which have been widely used in commercial environments. The collaborative filtering recommender system is an algorithm that calculates the similarity between users to identify neighbor users and recommends items according to the information of neighbors. However, data sparsity problems exist. The content-based recommender system determines the most similar commodity to recommend on the basis the user's favorite commodity information. The content-based recommender system establishes configuration files for users and items separately, analyses items that have been purchased (or browsed), establishes or updates the users' configuration file system, compares the similarity between the user and item configuration files and directly recommends the most similar items with their configuration files to the users. The content-based recommender system only considers the interest preference of users, and its recommendation results are intuitive and easy to understand. However, several data formats that are difficult to understand by a machine, such as music and images, often cannot be processed, and new interest points cannot be identified for users.

This study proposes a recommender system that combines popularity and novelty (RSCPN). The main contributions of this work are summarized as follows.

(1) We consider the number of users who have purchased the items during one-mode projection, thus weakening the impact of popular items on the similarity between users.

(2) The edge between a user and item is weighted with the item's rating, and the difference in the ratings of items given by different users is considered to determine a reasonable method of measuring the similarity between users.

(3) Compared with original methods, RSCPN can be selected as popularity or novelty recommendation in the same model according to the value of parameters, and it can analyze how a change in parameters affects the popularity and novelty of the recommender system. The results of RSCPN also exhibit accuracy, diversity and novelty.

The rest of this paper is organized as follows. Section 2 introduces the status of related research. Section 3 presents a method to measure user similarity based on one-mode projection of a weighted bipartite network. Section 4 proposes RSCPN and analyses the popularity and novelty of RSCPN. Section 5 demonstrates that RSCPN has good feasibility through an experimental verification and analysis of the model. Section 6 provides a summary of this research and directions for future work.

## 2 Related work

A recommender system 'collects information on the preferences of its users for a set of items' and 'seeks to predict the rating' or 'preference that a user would give to an item' [Bobadilla, Omega, Hernando et al. (2013)]. The task of a recommender system is to contact users and items. It helps users discover items that are valuable to them, and it allows items to be displayed before interested users to achieve a win-win situation for item consumers and producers. The most widely used recommendation methods include knowledge-based, content-based, collaborative filtering-based, and hybrid methods [Lu, Wu, Mao et al. (2015)].

A knowledge-based recommender system [Aggarwal (2016a)] is based on explicit knowledge of item classification, user preferences and recommendation criteria. It provides users with items based on knowledge that typically describes how the item meets users' needs. It is well-suited for areas where complex projects are not frequently purchased, such as apartments and cars. Other examples of project areas related to knowledge-based recommender systems are digital cameras, financial services and travel destinations.

A content-based recommender system [Aggarwal (2016b)] recommends products that are similar to the user's previous favorite products. Content-based recommendation aims to create an interest model for each user and recommend the content that matches the interest model to the user. For example, a peripheral product similar to the purchased product is recommended through the past purchase history of the user.

Collaborative filtering-based recommendation [Adomavicius and Kwon (2015); Wang, Zhang and Hendersonb (2017); Sun, Wang, Cheng et al. (2015); Jiang, Qian, Shen et al. (2015); Guo, Zhuang and Rabczuk (2019); Koren and Bell (2015); Chu, Mu, Liu et al. (2019)] refers to collecting the user's past behavior to obtain explicit or implicit information about a product. That is, item, content or user relevance is determined based on the user's preference for the item or information. Then, recommendations are made based on these associations. Recommendations based on collaborative filtering can be divided into user-, item- and model-based recommendations. In the user-based method Jia et al. [Jia, Yang, Gao et al. (2015); Chu, Hu, Shen et al. (2019)], a 'neighbor' user group with a similar taste and preference as the current user is identified based on the preference of all users for the item or information. In general applications, the calculation of 'K-neighbor' is used. The algorithm then recommends the current user based on the history preferences of K neighbors. Project-based collaborative filtering Li et al. [Li, Zhao, Wu et al. (2015); Zhang, Min and Shi (2017)] uses all user preferences for items or information to discover similarities between items and recommends similar items to users based on their historical preference information. Owing to their ease of deployment and efficiency, collaborative filtering-based methods are widely used in commercial systems, such as Amazon. However, recommendation performance is difficult to improve due to data sparseness and diversity and other issues. To improve recommendation accuracy, the model-based method uses the user's item scoring matrix to train a highly accurate scoring model. Examples include clustering [Nilashi, Esfahani, Roudbaraki et al. (2016)], Bayesian belief network [Ricci, Rokach and Shapira (2011)], Markov decision process [Liu and Wang (2018)] and latent semantic model [Kumar, Shrivastva and Singh (2016)]. Although model-based approaches improve the accuracy of predictions, they suffer from issues, such as model complexity, various parameters and strong dependence on large statistical properties of the dataset. Thus, applying model-based methods to actual recommender systems is difficult.

Various hybrid recommendation technologies [Wu, Yue, Pei et al. (2016); Gan (2016)] have been introduced and tested. Recommendations do not simply use only one type of recommended mechanism and strategy. They often combine multiple methods to achieve improved recommendations. The most common hybrid method combines different types of technologies, such as hybrid content-based collaborative filtering methods. In addition, different technologies of the same type can be combined, such as K nearest neighbor-based, content-based and naive Bayes-based content-based methods.

The concept of novelty recommendation was proposed by Herlocker et al. [Herlocker, Konstan, Terveen et al. (2004)] to recommend items to target users that they have potential interests in but do not know. Compared with accuracy recommendations, novelty recommendations can better develop user interests and make them relatively small. Items that are unpopular but can create great value are highly recommended. Oh et al. [Oh, Park, Yu et al. (2011)] proposed modelling the user's rating pattern in the user-item scoring matrix as the personal popularity tendency (PPT) and establishing corresponding items for the item. The scored model designs a PPT matching algorithm, in which the greater the difference is between the item's scored model and the target user's PPT, the higher the novelty of the item is. Onuma et al. [Onuma, Tong and Faloutsos (2009)] modelled the user-item scoring matrix. For a bipartite graph, the user and item are nodes, and the scoring association is an edge. This method uses the random walk approach to calculate the degree of association amongst all nodes and defines the 'TANGENT' value of the item node based on the degree of association. The higher the value is, the higher the novelty of the item is. Nakatsuji et al. [Nakatsuji, Fujiwara, Tanaka et al. (2010)] combined the classification information of the user-item scoring matrix and the item then defined the distance between the classification of the item and the classification of the user's rating as the novelty of the item to the target user; this method generates a recommendation list according to the novelty rank of the items. In [Vargas and Castells (2011); Yu, Peng, Hong et al. (2014)], the popularity of items was used to measure the items' novelty. The more popular an item is, the more likely the user knows the item and the less novelty the item has. In the user-item scoring matrix used in Kawamae [Kawamae (2010)], the scoring time is introduced; the user who scores the item earlier is regarded as an innovator, and the user who has not scored the item is a potential follower who thinks that the innovator has scored. The item has a high degree of novelty to the follower. The method regards the target user as a follower, calculates the probability that the other user is its innovator and recommends the innovator-rated item to the target user based on the probability value. Zhang et al. [Zhang, Seaghdha, Quercia et al. (2012)] constructed a graph with items as nodes, and items with similarity make up an edge. The user's scored items correspond to the subgraphs of the graph. This method adds specific item nodes to the subgraphs of the target user and calculates the clustering factor of the items.

Recommender systems based on bipartite networks [Zhang, Wang and Xiao (2017); Wang, Li, Luo et al. (2018)] are popular research topics in the field of personalized recommendation, and they elicit increasing attention because of their low recommendation complexity, high accuracy and diversity of recommendation content. In this study, we propose a recommender system combining popularity and novelty (RSCPN) based on one-mode projection of a weighted bipartite network. Compared with original methods, RSCPN can be selected as a popular or novelty recommendation according to the value of parameters in the same model. Moreover, the results of the algorithm have good accuracy, diversity and novelty.

## 3 User similarity measure of weighted one-mode projection

### *3.1 Weighted bipartite graph representation of user-item relationship in recommender system*

The bipartite graph is a network with a special structure. Fig. 1 shows that if an undirected network $G=(V,E,W)$ is a bipartite graph, it should have two sets of nodes

$X=\{x_1,x_2,\cdots x_m\}$ and $Y=\{y_1,y_2,\cdots y_n\}$, and these nodes should meet the following conditions.

1) X $\cap$ Y = $\varnothing$;

2) X $\cup$ Y = V;

3) Any edge of E must have exactly one node in the set X, and the other node in Y, i.e., $E: X \to Y \cup Y \to X$ and $(x_i,y_j)=(y_j,x_i), x_i \in X, y_j \in Y$;

4) W is the adjacency matrix of the bipartite graph G for the unweighted network, $W(i,j)=\begin{cases}1 & (x_i,y_j)\in E\\ 0 & (x_i,y_j)\notin E\end{cases}$.

$W(i,j)$ is the specific weight for the weighted network.

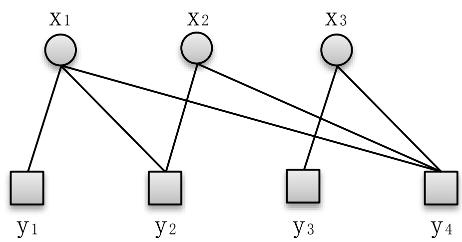



**Figure 1:** Bipartite graph

In the user-item recommender system, the collection of M users can be expressed as $U=\{u_1,u_2,\cdots u_m\}$, and the collection of N items can be expressed as $I=\{I_1,I_2,\cdots I_n\}$. The purchase relationship between users and items can form a bipartite graph called the user-item weighted bipartite graph, which is defined as follows:

1) $W$ represents the purchase relation matrix between users and items, $W(i,j)=1$ indicates that user $u_i$ has purchased the item $I_j$ and $W(i,j)=0$ indicates that the user $u_i$ has not bought item $I_j$.

2) $W^M$ is the purchase rating matrix between users and items and represents the weight of corresponding edges in the user-item weighted bipartite graph. $W^M(i,j)\neq 0$ indicates that user $u_i$ has purchased item $I_j$, and the rating is $W^M(i,j)$. $W^M(i,j)=0$ indicates that user $u_i$ has not bought item $I_j$.

### *3.2 User similarity measure of weighted one-mode projection*

In the user-item weighted bipartite graph $G=(U\cup I,E,W,W^M)$, one-mode projection can be carried out on the set of user nodes according to the purchase relationship between users and items, thereby creating an association between different users who have purchased the same item. Thus, the one-mode projection network of items to users based on the purchase relationship can be obtained. It is called the projection network from items to users and is presented as $G_{I\to U}=(U,E_U,W_U)$. It is defined as follows:

1) $E_U : U \rightarrow U$ represents the association between different users $u_i, u_j (i \neq j)$ based on one-mode projection.

2) $W_U(i,j)$ contains the weights of association relationships between different users, which are used to measure the similarity between different users, and it is called user similarity matrix based on one-mode projection.

The value of $W_U(i,j)$ is defined as follows:

(1) When considering only the purchase relationship in $G=(U \cup I, E, W, W^M)$ without considering the rating matrix $W^M$, the value of $W_U(i,j)$ is

$$W_U(i,j)=\begin{cases}\sum\limits_{I_l \in \Gamma(u_i) \cap \Gamma(u_j)} \dfrac{2}{K_{I_l}}, (u_i,u_j) \in E_U \\ 0, (u_i,u_j) \notin E_U \\ 1, i=j \end{cases} \tag{1}$$

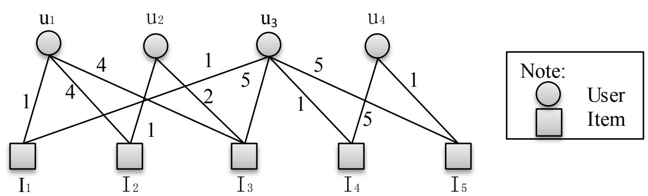
$\Gamma(u_i)$ and $\Gamma(u_j)$ are collections of items purchased by users $u_i$ and $u_j$, respectively. $K_{I_l}=\sum\limits_i W(i,l)$ is the number of users who have bought item $I_l$. The value of $W_U(i,j)$ is not the quantity of item jointly purchased by users $u_i$ and $u_j$. It considers the number of users who purchase item $I_l$, which is jointly purchased (i.e. the degree of item $I_l$ in bipartite graph $G=(U \cup I, E, W, W^M)$). When the number of users who purchase item $I_l$ is large, the contribution to the correlation degree between $u_i$ and $u_j$ in one-mode projection, which is expressed as $\frac{2}{K_{I_l}}$, is small. On the contrary, when the number of users who purchase item $I_l$ is small, the contribution to the correlation degree between $u_i$ and $u_j$ in one-mode projection, which is expressed as $\frac{2}{K_{I_l}}$, is large. In this manner, the influence of the popular item is weakened. The item that is purchased by many users is identified based on the similarity between different users, thus making the result of user similarity reasonable.

(2) When considering the rating matrix $W^M$ in $G=(U \cup I, E, W, W^M)$, the value of $W_U(i,j)$ is

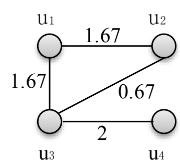
$$W_U'(i,j)=\begin{cases}\sum\limits_{I_l \in \Gamma(u_i) \cap \Gamma(u_j)} \dfrac{2e^{-\frac{|W^M(i,l)-W^M(j,l)|}{\alpha}}}{K_{I_l}}, (u_i,u_j) \in E_U \\ 0, (u_i,u_j) \notin E_U \\ 1, i=j \end{cases} \tag{2}$$

$\alpha$ is a number whose value is greater than 0, and $\Gamma(u_i)$ and $K_{I_l}=\sum\limits_i W(i,l)$ are similar to those described above.

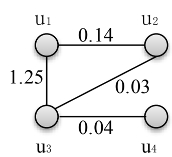
The value of $W_U'(i,j)$ considers users' ratings of purchased items to measure the similarity of different users $u_i, u_j$. The definition of $W'_U(i,j)$ in Formula (2) can guarantee that the closer the ratings of different users to buy the same item are, the greater the value of $W'_U(i,j)$ is. In the case of the same rating difference, the popular item purchased by different users contributes less to the value of $W'_U(i,j)$, whereas the purchase of slow-selling items greatly contributes to the value of $W'_U(i,j)$. This condition also meets the requirements of personalized and novel recommendation.




(a) User-item weighted bipartite network




(b) One-mode projection without considering rating




(c) One-mode projection with considering rating

**Figure 2:** One-node projection of weighted bipartite network

Fig. 2 shows that when we consider only the user's purchase of items and the number of users who purchase each item and not the rating, the following holds true

$W_U(1,2) = 1.67$, $W_U(1,3) = 1.67$, $W_U(3,4) = 2$.

Although the number of items purchased by $u_1$ and $u_2$, $u_1$ and $u_3$, $u_3$ and $u_4$ is all 2, the number of users buying item $I_3$ is large, resulting in $W_U(3,4) > W_U(1,3) = W_U(1,2)$.

When considering a user's purchase situation, the number of users who purchase each item, the users' rating of the item and $\alpha = 1$, we have

$W'_U(1,2) = 0.14$, $W'_U(1,3) = 1.25$, $W'_U(3,4) = 0.04$.

Although the number of items purchased by $u_1$ and $u_2$, $u_1$ and $u_3$ and $u_3$ and $u_4$ is all 2, the ratings of $u_1$ and $u_2$ for the same items purchased are very similar, $u_3$ and $u_4$ have slightly similar ratings for the same items purchased and the rating similarity of $u_1$ and $u_3$ to the same items is between the two above. Thus, $W_U'(1,3) > W_U'(1,2) > W_U'(3,4)$, and the value of $W_U'(1,3)$ is much greater than the value of $W_U'(3,4)$, i.e. the similarity between $u_1$ and $u_3$ is far greater than the similarity between $u_3$ and $u_4$. This result is more proper because the closer the ratings of different users for the same items are, the higher their similarity is; otherwise, the lower their similarity is.

To obtain a unified standard and make the value of $W_U'(i,j)$ fall within [0,1], we can implement normalization on $W_U'(i,j)$ as follows:

$$W_U'(i,j) = \begin{cases} \dfrac{W_U'(i,j)}{\max W_U'(i,j)}, i \neq j \\ 1, i = j \end{cases} \tag{3}$$

The term $\max W_U'(i,j)$ represents the maximum of all values in the original $W_U'(i,j)$.

## 4 Popular and novel recommendation based on user similarity

### *4.1 Calculation of an item's recommended predictive value combining popularity and novelty for a single user*

For a user $u_i$ in the recommender system, the principle of commodity recommendation is as follows:

1) Determine the most similar k users.

2) From the items purchased by k similar users, identify p items with the highest rating that have not been purchased by user $u_i$ for recommendation to user $u_i$.

The details are as follows:

The k users that are most similar to user $u_i$ are the users corresponding to the k maximum values in $W_U'(i,j)(i \neq j, j = 1,2,\cdots m)$ marked as $U_i = \{u_{i1}, u_{i2}, \cdots, u_{ik}\}$. The item collection that has been purchased by $u_{i1}, u_{i2}, \cdots, u_{ik}$ and has not been purchased by $u_i$ is marked as $I(u_i)$, resulting in $I(u_i) = \Gamma(u_{i1}) \bigcup \Gamma(u_{i2}) \bigcup \cdots \bigcup \Gamma(u_{ik}) - \Gamma(u_i)$.

When recommending p items to user $u_i$ from $I(u_i)$, the items that have been purchased by the user of the most similar k users to user $u_i$ and with higher ratings have higher priority of recommendation. Therefore, we can calculate the weighted average value of each item in $I(u_i)$ initially as the basis of the recommendation's predicted value. To combine popular and novel recommendation, the predictive value recommended to user $u_i$ for an item in $I(u_i)$, marked as $I_l \in I(u_i)$, can be defined as

$$R(i,l) = \frac{\sum_{u_j \in U_i} W_U'(i,j) W(j,l)}{|\Gamma(I_l) \bigcap U_i|} \cdot e^{-\frac{\beta}{|\Gamma(I_l) \bigcap U_i|}} \tag{4}$$

$|\Gamma(I_l) \bigcap U_i|$ represents the number of users who have purchased item $I_l$ and belong to

$U_i = \{u_{i1}, u_{i2}, \cdots, u_{ik}\}$, the most similar k users to $u_i$.

$\frac{\sum_{u_j \in U_i} W'_U(i,j)W(j,l)}{|\Gamma(I_l) \cap U_i|}$ represents the weighted average of rating value of item $I_l$, which is given by $u_j \in \Gamma(I_l) \cap U_i$, the users who have purchased the item $I_l$ and belong to $U_i = \{u_{i1}, u_{i2}, \cdots, u_{ik}\}$, the most similar k users to $u_i$. The weight is $W'_U(i,j)$, which is the similarity between user $u_i$ and user $u_j$.

$e^{-\frac{\beta}{|\Gamma(I_l) \cap U_i|}}$ represents the correction to $\frac{\sum_{u_j \in U_i} W'_U(i,j)W(j,l)}{|\Gamma(I_l) \cap U_i|}$, which is a part of the weighted average value, considering the number of users buying $I_l$ in $U_i = \{u_{i1}, u_{i2}, \cdots, u_{ik}\}$. $\beta$ is a correction factor.

### *4.2 Analysis of popularity and novelty in RSCPN*

For the recommended prediction value $R(i,l)$ expressed by Formula (4), the value range of correction factor $\beta$ is generally [-1,1]. When $\beta > 0$, $e^{-\frac{\beta}{|\Gamma(I_l) \cap U_i|}}$ is the increasing function of $|\Gamma(I_l) \cap U_i|$, which is the number of users who have purchased the item $I_l$ and belong to $U_i = \{u_{i1}, u_{i2}, \cdots, u_{ik}\}$. When $|\Gamma(I_l) \cap U_i|$ increases, the larger the number of users who have purchased the item $I_l$ and belong to collection of similar users is, the greater the recommended predictive value, marked as $R(i,l)$, is. Therefore, $\beta \geq 0$ denotes a popular recommendation.

When $\beta < 0$, $e^{-\frac{\beta}{|\Gamma(I_l) \cap U_i|}}$ is the decreasing function of $|\Gamma(I_l) \cap U_i|$, which means that the larger the number of users who have purchased item $I_l$ and belong to collection of similar users is, the smaller the recommended predictive value is. The smaller the number of users who have purchased item $I_l$ and belong to the collection of similar users is, the greater the recommended predictive value is. Therefore, $\beta < 0$ denotes a novel recommendation. In the case of the same weighted average value, the smaller the number of users who have purchased item $I_l$ and belong to $U_i = \{u_{i1}, u_{i2}, \cdots, u_{ik}\}$ is, the greater the recommended predictive value of item $I_l$, marked as $R(i,l)$, is. This means that item $I_l$ is more likely to be recommended to the user.

When $\beta = 0$, the recommended value does not consider the sales of $I_l$, and the recommended predicted value is equal to the weighted average value of ratings.

Here, novelty and popular recommendations assume that the purchase situation of item $I_l$ provided by users in $U_i = \{u_{i1}, u_{i2}, \cdots, u_{ik}\}$ can reflect the overall purchase situation of item $I_l$ provided by the entire users. If highly accurate results are needed, the value of k can be appropriately increased.

### *4.3 Overall recommendation of user-item in RSCPN*

To facilitate the representation and calculation of user-item recommendation by matrix operation, user similarity matrix $W_U$ can be processed to $W''_U = W'_U - I$ ($I$ is a unit matrix). Hence, the items bought by user $u_i$ do not affect the recommended predicted value in the recommended predicted process of the items that have been purchased by user $u_i$. For any item $I_l$, the recommended predicted value of user $u_i$ can be expressed as follows:

$$R(i,l) = \frac{\sum_{u_j \in U} W''_U(i,j) W^M(j,l)}{\sum_{u_j \in U} W''_U(i,j) W(j,l)} \cdot e^{-\frac{\beta}{\sum_{u_j \in U} W''_U(i,j) W(j,l)}} \tag{5}$$

We define $e^{-\frac{\beta}{W''_U W}}$ as a matrix, and the element at row i and column l is $e^{-\frac{\beta}{W''_U W}}(i,l) = e^{-\frac{\beta}{\sum_{u_j \in U} W''_U(i,j) W(j,l)}}$.

As a result, $W'_U W^M$, $W'_U W$ and $e^{-\frac{\alpha}{W''_U W}}$ are homotypic matrixes. We can define the $\cdot *$ and $\cdot /$ operations as follows:

(1) $[(W'_U W^M) \cdot * e^{-\frac{\beta}{W''_U W}}](i,j) = (W'_U W^M)(i,j) \cdot e^{-\frac{\beta}{W''_U W}}(i,j)$, i.e., the $\cdot *$ operation of two homotypic matrixes, is the multiplication of elements corresponding to two matrixes.

(2) $[(W'_U W^M) \cdot / (W'_U W)](i,j) = \begin{cases} \frac{(W'_U W^M)(i,j)}{(W'_U W)(i,j)}, (W'_U W)(i,j) \neq 0 \\ 0, (W'_U W)(i,j) = 0 \end{cases}$, i.e., the $\cdot /$ operation of two homotypic matrixes, is the division of elements corresponding to two matrixes (if the divisor is 0, the result is 0).

Therefore, the matrix of the recommended predictive value between users and items is expressed as follows:

$$R = [(W'_U W^M) \cdot / (W'_U W)] * e^{-\frac{\beta}{W''_U W}} \tag{6}$$

During recommendation, to carry out novel recommendation, we can take $\beta < 0$ and recommend k items that have the largest recommended predictive value of users $u_i (i = 1, 2, \cdots, m)$ from the obtained recommendation matrix $R$ to user $u_i$. To carry out popular recommendation, we can take $\beta > 0$. In addition, if novelty and popularity are required simultaneously, we can take $k/2$ items with the largest value from the recommended predictive matrix when $\beta < 0$ and take $k/2$ items with the largest value from the recommended predictive matrix when $\beta > 0$. Thus, we can obtain k items. Algorithm flow chart:

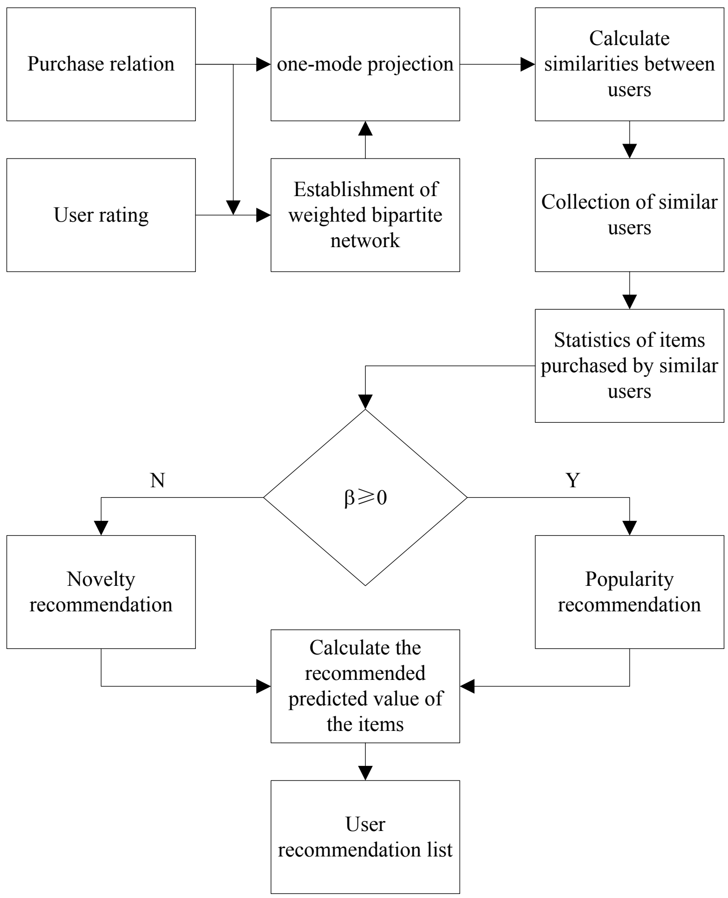



**Figure 3:** Flow chart of RSCPN

Next, the time and space complexity of RSCPN is analyzed below, assuming that there are n users and m items in the recommendation system. The algorithm is used to calculate the similarity between users, and its time complexity is $O(m^2 n)$; The algorithm is used to calculate the predicted value of the user for unpurchased items, and its time complexity is $O(n)$. In general, the time complexity of RSCPN is $O(m^2 n)$. RSCPN uses a matrix to store user ID, item ID and user rating with a spatial complexity of $O(nm)$.

## 5 Experiments

### *5.1 Dataset*

For the experiment, we use the MovieLens movie dataset, which was created by the GroupLens research group at the University of Minnesota in United States. The source site is http://www.grouplens.org. The dataset contains 100,000 pieces of rating data for 1,628 movies provided by 943 users. Each user evaluated at least 20 movies, and the rating is between 1 and 5. The higher the rating is, the more the user liked the movie. If the rating is greater than or equal to 3, the user likes the movie. If the rating is less than 3, the user does not like the movie. The algorithm proposed in this study divides the selected data into training (80%) and test (20%) sets. Each record of the dataset contains the following fields: user ID, item ID, user rating (1-5) and timestamp.

### *5.2 Evaluation metrics*

The mean absolute error (MAE), root mean squared error (RMSE) and hamming distance (HD) are used in this work to measure the performance of popular recommendation. The average popularity is used to evaluate the quality of novel recommendation.

(1) MAE and RMSE are used to represent the grade of accuracy according to the deviation between predictive and actual values. The smaller the error is, the higher the recommendation accuracy is. If the set of user's predictive rating is $\{p_1, p_2, p_3, p_4, \cdots p_{N-1}, p_N\}$ and the set of user's actual rating is $\{q_1, q_2, q_3, q_4, \cdots q_{N-1}, q_N\}$, then MAE and RMSE [Sexton and Laake (2009)] are defined as

$$\text{MAE}=\frac{\sum_{i=1}^{N}(p_i - q_i)}{N} \tag{7}$$

$$\text{RMSE}=\sqrt{\frac{\sum_{i=1}^{N}(p_i - q_i)^2}{N}} \tag{8}$$

(2) HD [Adomavicius and Kwon (2011)] evaluates the diversity of the predicted result based on the number of identical items in different users' recommendation lists. The HD between the recommended list of $u_i$ and the recommended list of $u_j$ is defined as

$$H_{ij}=1-\frac{|Q_{ij}|}{L} \tag{9}$$

where $Q_{ij}$ represents the collection of public items between the recommended list of $u_i$ and the recommended list of $u_j$, $|Q_{ij}|$ represents the number of elements in $Q_{ij}$ and L represents the length of a recommended list. If the two recommended lists are completely consistent, then $Q_{ij}=0$. If the two recommendation lists do not have any similar item, then $Q_{ij}=1$. The average value of all users' HD is the HD of the entire system.

$$\text{HD}=\frac{\sum_{i \neq j} H_{ij}}{m*(m-1)} \tag{10}$$

where m represents the number of users.

The larger HD is, the higher the diversity of the recommended result is.

(3) The commonly used method of novelty [Hurley and Zhang (2011)] evaluation utilizes the average popularity of the recommended result. Popularity <k> [Su and Khoshgoftaar (2009)] can reflect the recommendation for slow-selling items. The less popular items are

likely to make users feel novel. The popularity of the recommender system can be expressed by the mean <k> of the recommended item's degree. If <k> is high, the number of popular items in the recommended items is large. <k>, the average value of item's degree, can be defined as

$$\langle k \rangle = \frac{1}{nL}\sum_{j=1}^{n}\sum_{i=1}^{L} p(i) \tag{11}$$

where n represents the number of users, L represents the length of the recommended list and $p(i)$ represents the degree of item $I_i$. The lower the average value of the item's degree is, the higher the novelty is.

### *5.3 Experimental results*

We experiment with the effect of parameters $\alpha$, K and $\beta$ on the experimental result. Then, we compare the four algorithms, which include the algorithm of taking a certain value of $\alpha$, K and β proposed in this paper, the spreading activation approach for collaborative filtering (SA-CF) algorithm proposed by Liu et al. [Liu, Wang and Guo (2009)], the network-based inference (NBI) algorithm proposed by Zhou et al. [Zhou, Ren, Medo et al. (2007)] and the hybrid method of heat conduction and mass diffusion (HHM) algorithm proposed by Zhou et al. [Zhou, Kuscsik, Liu et al. (2010)].

#### *5.3.1 Calibration of dynamic adjustable parameter $\alpha$*

Fig. 4 shows the impact of dynamic factor $\alpha$ on algorithm accuracy. When $\alpha$ changes, the MAE and RMSE of the recommended result also change. The result is the best when the value of $\alpha$ is approximately 0.6.

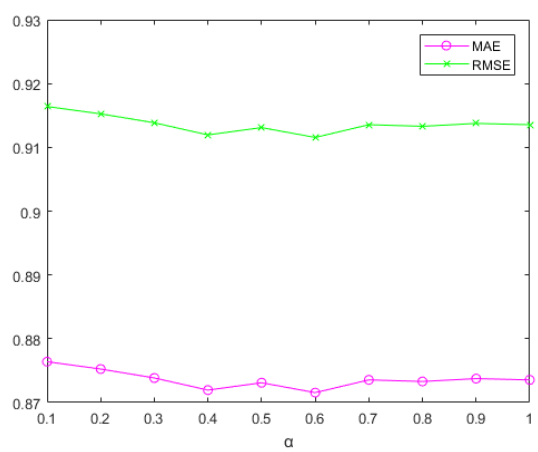



**Figure 4:** Impact of dynamic factor $\alpha$ on algorithm accuracy

#### *5.3.2 Impact of the number of nearest neighbor (K) on the recommendation accuracy*

Selecting an appropriate number of nearest neighbors can improve accuracy and reduce the computation time.

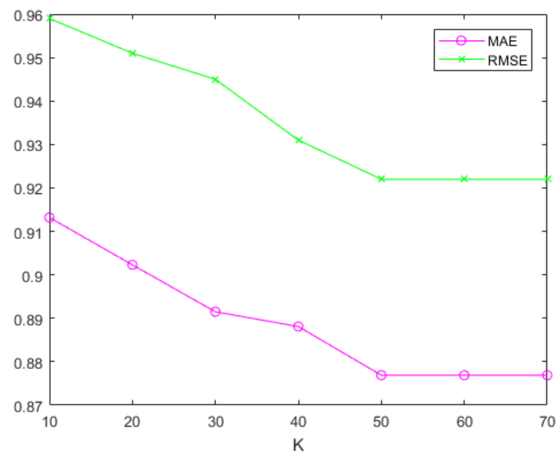



**Figure 5:** Impact of the number of nearest neighbors (K) on recommendation accuracy

Fig. 5 shows that MAE and RMSE decrease with the increase in K, the number of nearest neighbors. The reason is that when the number of users with high similarity increases, the recommended result becomes more accurate. However, when K reaches approximately 50, the function curves of MAE and RMAE become stable, so we set the value of K to 60.

*5.3.3 Analysis and comparison of experimental results*

We perform a comparative experiment. The threshold of the nearest neighbor is set to 60 in the experiment, and the results are described below.

1) Analysis of novel recommendation

(1) Impact of parameter β on novelty

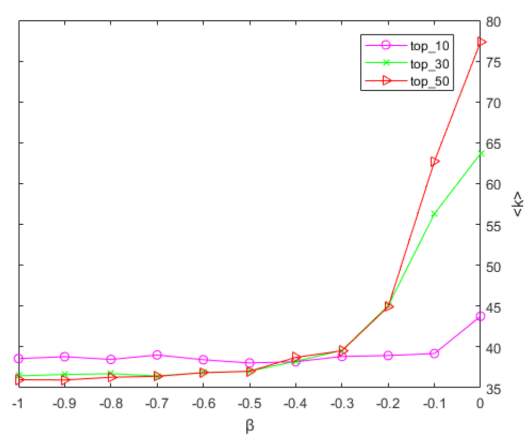



**Figure 6:** Impact of β on the novelty of RSCPN

Fig. 6 shows the impact of dynamic β on the novelty of RSCPN when the threshold of the nearest neighbor is set to 60, and 'top_10' indicates that the length of the recommended list is 10. When β changes, <k>, the mean of the recommended item's degree, also varies.

When $\beta \leq -0.5$, the function curve of <k> becomes stable. In this study, we take $\beta$=-0.8.

(2) Comparison of novelty

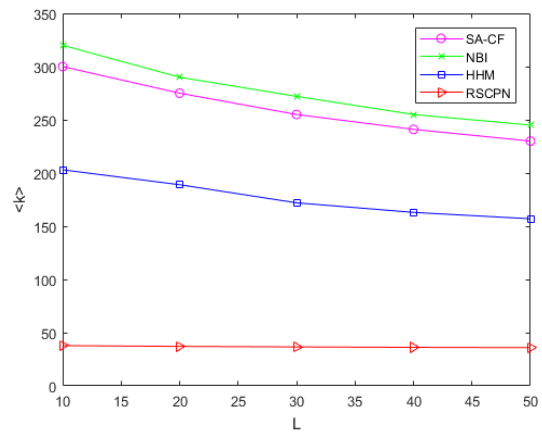



**Figure 7:** Comparison of popularity

The commonly used method of novelty evaluation utilizes the average popularity of the recommended result. Figs. 6 and 7 show that RSCPN significantly improves the recommendation effect of slow-selling items compared with the three traditional algorithms, namely, SA-CF, NBI and HHM. Fig. 7 shows that RSCPN significantly reduces the popularity of recommended items under the effect of parameter $\beta < 0$. The personalization and novelty of the recommender system are improved, and the items that are not very popular in the recommender system can be recommended to users.

2) Comparison of accuracy and diversity

Fig. 8 shows the comparison of SA-CF, NBI, HHM and RSCPN in terms of MAE, RMSE and HD.

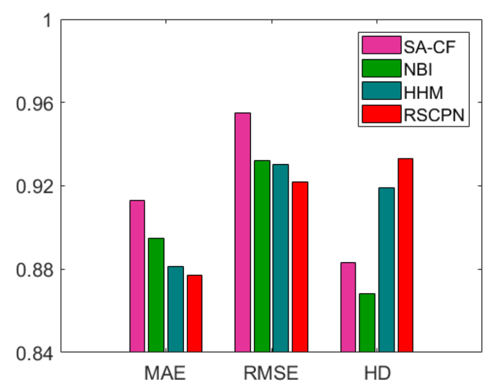



**Figure 8:** Comparison of different algorithms based on accuracy and diversity

The comparison of MAE and RMSE in Fig. 8 shows that RSCPN is clearly superior to SA-CF, NBI and HHM in terms of recommendation accuracy, which proves that the proposed algorithm has higher accuracy and has more chances of recommending favorite items to users. The comparison of the three algorithms in terms of HD shows that RSCPN has better diversity than SA-CF, NBI and HHM.

The experimental results show that RSCPN greatly improves the diversity of recommendation. The recommendation of popular movies is effectively suppressed in the movie recommendation because the influence of user's and item's degrees is considered, and users who contribute to the target user are deemed accurate. In addition, the personalization of recommendation is improved, thus meeting the multi-interest requirements of different users.

## 6 Conclusions and future work

Personalized recommender algorithms are effective tools for solving the problem of information overload and help users filter massive information accurately. In view of the characteristics of the traditional bipartite recommender algorithm, we perform one-mode projection on a set of user nodes based on a user-item bipartite graph and obtain a one-mode projection network from items to users. During one-mode projection, a highly reasonable method of measuring user similarity is obtained by considering the number of users who have bought the item and the difference in item ratings from different users. We present RSCPN based on the user similarity measure and analyze the influence of popularity and novelty in the recommender system. Then, we verify the proposed model and compare its accuracy, diversity and novelty with those of other models. The experimental results show that introducing ratings into the bipartite graph and considering the difference in user ratings are feasible and greatly improve the accuracy of the recommended results. During novel recommendation, the recommendation of cold movies increases the diversity of the algorithm. Apart from users' social behavior data, the socialized label information of users is also an important source of data for the user label. Therefore, recommending based on the user-item-label three-partite graph structure will be the next focus of our research.

**Acknowledgement:** Project funded by the National Science Foundation of China under Grant (Nos. 61462091, 61672020, U1803263, 61866039, 61662085), by the Data Driven Software Engineering innovation team of Yunnan province (No. 2017HC012), by Scientific Research Foundation Project of Yunnan Education Department (No. 2019J0008, 2019J0010), by China Postdoctoral Science Foundation (Nos. 2013M542560, 2015T81129), A Project of Shandong Province Higher Educational Science and Technology Program (No. J16LN61).

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

**Adomavicius, G.; Kwon, Y.** (2011): Improving aggregate recommendation diversity using ranking-based techniques. *IEEE Transactions on Knowledge and Data Engineering*, vol. 24, no. 5, pp. 1-15.

**Adomavicius, G.; Kwon, Y.** (2015): Multi-criteria recommender systems. *Recommender Systems Handbook*, pp. 847-880.

**Aggarwal, C. C.** (2016): Content-based recommender systems. *Recommender Systems*, pp. 139-166.

**Aggarwal, C. C.** (2016): Knowledge-based recommender systems. *Recommender Systems*, pp. 167-197.

**Bobadilla, J.; Omega, F.; Hernando, A.; Gutierrez, A.** (2013): Recommender systems survey. *Knowledge-Based Systems*, vol. 46, pp. 109-132.

**Chen, M.; Teng, C.; Chang, P.** (2015): Applying artificial immune systems to collaborative filtering for movie recommendation. *Advanced Engineering Informatics*, vol. 29, no. 4, pp. 830-839.

**Chu, C.; Hu, X.; Shen, C.; Li, T.; Boccaletti, S. et al.** (2019): Self-organized interdependence among populations promotes cooperation by means of coevolution. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 29, 013139.

**Chu, C.; Mu, C.; Liu, J.; Liu, C.; Boccaletti, S. et al.** (2019): Aspiration-based coevolution of node weights promotes cooperation in the spatial prisoner's dilemma game. *New Journal of Physics*, vol. 21, 063024.

**Gan, M.** (2016): COUSIN: a network-based regression model for personalized recommendations. *Decision Support Systems*, vol. 82, pp. 58-68.

**Guo, H.; Zhuang, X.; Rabczuk, T.** (2019): A deep collocation method for the bending analysis of kirchhoff plate. *Computers, Materials & Continua*, vol. 59, no. 2, pp. 433-456.

**Herlocker, J. L.; Konstan, J. A.; Terveen, L. G.; Riedl, J. T.** (2004): Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems*, vol. 22, no. 1, pp. 5-53.

**Hurley, N.; Zhang, M.** (2011): Novelty and diversity in top-n recommendation-analysis and evaluation. *ACM Transactions on Internet Technology*, vol. 10, no. 4, pp. 1-30.

**Jia, Z.; Yang, Y.; Gao, W.; Chen, X.** (2015): User-based collaborative filtering for tourist attraction recommendations. *IEEE International Conference on Computational Intelligence & Communication Technology*, pp. 22-25.

**Jiang, S.; Qian, X.; Shen, J.; Fu, Y.; Mei, T.** (2015): Author topic model-based collaborative filtering for personalized P01 recommendations. *IEEE Transactions on Multimedia*, vol. 17, no. 6, pp. 907-918.

**Kawamae, N.** (2010): Serendipitous recommendations via innovators. *Proceedings of the 33rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 218-225.

**Koren, Y.; Bell, R.** (2015): Advances in collaborative filtering. *Recommender Systems Handbook*, pp. 77-118.

**Kumar, V.; Shrivastva, K. M. P.; Singh, S.** (2016): Cross domain recommendation using semantic similarity and tensor decomposition. *Procedia Computer Science*, vol. 85, pp. 317-324.

**Li, C.; Zhao, P.; Wu, J.; Mao, J.; Cui, Z.** (2015): An item-based collaborative filtering framework based on preferences of global users. *Proceedings of the 4th International Conference on Computer Engineering and Networks*, pp. 1113-1120.

**Li, Y.; Wang, S.; Pan, Q.; Peng, H.; Yang, T. et al.** (2019): Learning binary codes with neural collaborative filtering for efficient recommendation systems. *Knowledge-Based Systems*, vol. 172, pp. 64-75.

**Liu, J.; Wang, B.; Guo, Q.** (2009): Improved collaborative filtering algorithm via information transformation. *International Journal of Modern Physics C*, vol. 20, no. 2, pp. 285-293.

**Liu, S.; Wang, L.** (2018): A self-adaptive point-of-interest recommendation algorithm based on a multi-order markov model. *Future Generation Computer Systems*, vol. 89, pp. 506-514.

**Lu, J.; Wu, D.; Mao, M.; Wang, W.; Zhang, G.** (2015): Recommender system application developments: a survey. *Decision Support Systems*, vol. 74, pp. 12-32.

**Nakatsuji, M.; Fujiwara, Y.; Tanaka, A.; Uchiyama, T.; Fujimura, K. et al.** (2010): Classical music for rock fans: novel recommendations for expanding user interests. *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pp. 949-958.

**Nilashi, M.; Esfahani, M. D.; Roudbaraki, M. Z.; Ramayah, T.; Ibrahim, O.** (2016): A multi-criteria collaborative filtering recommender system using clustering and regression techniques. *Journal of Soft Computing and Decision Support Systems*, vol. 3, no. 5, pp. 24-30.

**Oh, J.; Park, S.; Yu, H.; Song, M.; Park, S. T.** (2011): Novel recommendation based on personal popularity tendency. *IEEE 11th International Conference on Data Mining*, pp. 507-516.

**Onuma, K.; Tong, H. H.; Faloutsos, C.** (2009): Tangent: a novel, "surprise me", recommendation algorithm. *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 657-666.

**Ricci, F.; Rokach, L.; Shapira, B.** (2011): Introduction to recommender systems handbook. *Recommender Systems Handbook*, pp. 1-35.

**Schafer, J. B.; Frankowski, D.; Herlocker, J.; Sen, S.** (2007): Collaborative filtering recommender systems. *Adaptive Web*, pp. 291-324.

**Sexton, J.; Laake, P.** (2009): Standard errors for bagged and random forest estimators. *Computational Statistics & Data Analysis*, vol. 53, no. 3, pp. 801-811.

**Silvia, P.; Javier, P. A.; Jordi, F.; David, R. M.** (2015): On content-based recommendation and user privacy in social-tagging systems. *Computer Standards & Interfaces*, vol. 41, pp. 17-27.

**Son, J.; Kim, S. B.** (2017): Content-based filtering for recommendation systems using multiattribute networks. *Expert Systems with Applications*, vol. 89, pp. 404-412.

**Su, X.; Khoshgoftaar, T. M.** (2009): A survey of collaborative filtering techniques. *Advances in Artificial Intelligence*, vol. 10, no. 4, pp. 1-19.

**Sun, J.; Wang, G.; Cheng, X.; Fu, Y.** (2015): Mining affective text to improve social media item recommendation. *Information Processing & Management*, vol. 51, no. 4, pp. 444-457.

**Vargas, S.; Castells, P.** (2011): Rank and relevance in novelty and diversity metrics for recommender systems. *Proceedings of the Fifth ACM Conference on Recommender Systems*, pp. 109-116.

**Wang, F.; Zhang, S.; Hendersonb, L. M.** (2017): Adaptive decision-making of breast cancer mammography screening: a heuristic-based regression model. *Omega*, vol. 76, pp. 70-84.

**Wang, Q.; Li, J.; Luo, X.; Xu, J.; Shang, M.** (2018): Effects of the bipartite structure of a network on performance of recommenders. *Physica A: Statistical Mechanics and Its Applications*, vol. 492, pp. 1257-1266.

**Wu, H.; Yue, K.; Pei, Y.; Li, B.; Zhao, Y. et al.** (2016): Collaborative topic regression with social trust ensemble for recommendation in social media systems. *Knowledge-Based Systems*, vol. 97, pp. 111-122.

**Yu, Q.; Peng, Z.; Hong, L.; Liu, B.; Peng, H.** (2014): Novel community recommendation based on a user-community total relation. *Database Systems for Advanced Applications*, pp. 281-295.

**Zhang, H.; Min, F.; Shi, B.** (2017): Regression-based three-way recommendation. *Information Sciences*, vol. 378, pp. 444-461.

**Zhang, P.; Wang, D.; Xiao, J.** (2017): Improving the recommender algorithms with the detected communities in bipartite networks. *Physica A: Statistical Mechanics and Its Applications*, vol. 471, pp. 147-153.

**Zhang, Y.; Seaghdha, D.; Quercia, D.; Jambor, T.** (2012): Auralist: introducing serendipity into music recommendation. *Proceedings of the 5th ACM International Conference on Web Search and Data Mining*, pp. 13-22.

**Zhou, T.; Kuscsik, Z.; Liu, J.; Medo, M.; Wakeling, J. R. et al.** (2010): Solving the apparent diversity-accuracy dilemma of recommender systems. *Proceedings of the National Academy of Sciences*, vol. 107, no. 10, pp. 4511-4515.

**Zhou, T.; Ren, J.; Medo, M.; Zhang, Y.** (2007): Bipartite network projection and personal recommendation. *Physical Review E*, vol. 76, 046115.

**Zhou, T.; Ren, J.; Medo, M.; Zhang, Y.** (2007): Bipartite network projection and personal recommendation. *Physical Review E*, vol. 76, no. 4, pp. 6116-6123.