Recommending Users to Follow based on User Taste Homophily for Content Curation Social Networks

Yuchen Jing
Beijing University of Technology, China
jjyycchh@emails.bjut.edu.cn

Lifang Wu*
Beijing University of Technology, China
lfwu@bjut.edu.cn

Xiuzhen Zhang†
School of CS and IT
RMIT University, Australia
xiuzhen.zhang@rmit.edu.au

Dan Wang
Beijing University of Technology, China
danwang2013@emails.bjut.edu.cn

Changwen Chen
Computer Science And Engineering
SUNY Buffalo, U.S.A.
chencw@buffalo.edu

ABSTRACT

Online content curation social networks are an increasingly popular type of social networks where users pin items they find on the Web into categorical boards defined by users (hence the "curation"). In this paper, we study the problem of recommending users to follow on such social networks. Different from existing friendship-oriented or content-oriented user recommendation approaches, we design a recommendation scheme by collaborative filtering based on homophily for users' tastes. We propose to discover homophily for users' tastes from their "repin paths". Specifically we measure the similarity between repin paths using the Levenshtein Distance. The similarity and the number of common users for repin paths are combined to compute the homophily for users' tastes. Experiments on a content curation social network show that our recommendation algorithm of collaborative filtering based on user taste homophily performs better than user popularity based recommendation.

Categories and Subject Descriptors

H.3.3 [**Information Search and Retrieval**]: Information filtering; H3.5 [**Online Information Services**]: Web-based services

General Terms

Algorithms, Experimentation

*Corresponding author

†Corresponding author

SRS '15 Sydney, Australia

Copyright ©2015 for this paper by its authors. Copying permitted for private and academic purposes.

Keywords

Social recommendation, content curation social networks, user homophily, levenshtein distance, recommender systems

1. INTRODUCTION

With the development of the Web 2.0 technology, social networks have been widely used. From Blog to Twitter and to Flickr, social networks are becoming easier and easier for user interactions. Since the launch of Pinterest in 2009 content curation social networks (CCSNs). CCSNs quickly become popular worldwide with new additions to the family, such as `Huaban.com` and `snip.it`.

CCSNs are interest-sharing networks. On these networks, users can publish items they created, and can also collect items they are interested in from other users' collection by 'repin'. The "curation" aspect of these social networks comes from the fact that users organize their pins into categories called "boards" and thus these sites produce a huge volume of personalized curated content.

CCSNs are an increasingly popular type of social networking sites, most research has focused on the statistical overview of user demographics and behaviors [20], the content statistical overview[10] or differences between the image-based social curation networks and the textual message-based social networks[19]. Notably Chang et al [6] found that "homophily drives repining (on Pinterest): people repin content from other users who share their interests". Limited work has been done on recommendation for the CCSNs. In the literature recommendation of "pins" and "boards" using content-based filtering techniques is proposed [14, 17]. But neither the homophily nor the "repin" relationship between users are considered for recommendation.

There has been previous work on recommendation of users to follow on social networks. [1, 7, 9, 11, 12, 15, 24, 25, 27]. Existing approaches for people recommendation are either friendship-oriented [7, 9] or content-oriented [1, 11, 12, 15, 24, 25, 27]. In the first approaches the context defined by service items are largely ignored whereas the latter approaches are mainly content-driven and enhanced by user social interaction.

In this paper we study the problem of recommendation of users to follow for CCSNs. Considering the heterogeneity of contents on the CCSNs we focus on making use of only the social interaction of users for recommendation. We face two challenges: How to characterize the social interactions between users? How to make use of the social interactions for homophily-oriented user recommendation?

Motivated by the observation in [6] we further formulate “user repin by similar interests” as homophily for user tastes. Indeed two types of homophily are distinguished ÍC baseline homophily based on demographics and inbreeding homophilgy based on other factors [18]. Especially the latter includes personal preferences, which in our context we call taste homophily. We model the repin behaviour of users and make recommendation based on the taste homophily of users derived from the repin profiles of users. The contributions of our scheme are as follows:

(1) We propose the novel application of repin path to profile users. We measure the similarity between repin paths using the Levenshtein Distance and quantify user taste homophily from the similarity and the number of common users for repin paths.

(2) We propose a lightweight collaborative filtering algorithm based on the homophily of users.

Our experimental results on `Huban.com`, one of the CCSNs, demonstrate the effectiveness of our proposed recommendation algorithm.

2. RELATED WORK

The problem of user recommendation on social networks has been studied in the literature [12]. Conventional user recommendations are mostly friendship oriented [7, 9]. For example Twitter itself recommends popular users or friends' friends for people to follow. However such friendship-oriented recommendation service may not offer the most relevant followees for users. content-oriented user recommendation approaches were proposed [1, 11, 12, 15, 24, 25, 27]. To achieve interesting recommendation for usersąŕ information need, the interaction between user and service items (Twitter posts, news and product) and social interaction between users are combined for recommendation. Notably Yang et al. [27] proposed to exploit homophily among users discovered from these two types of interactions to achieve recommendation that considers both friendship and interests. These existing approaches differ from our proposed approach, which is based on the observed principle that the repin behaviour of users characterizes homophily [6, 8] of users on CCSNs.

The limited work in the literature on recommendation for CCSNs has been focused on recommendation of “boards” and “pins” using content-based filtering. Kamath [14] recommended boards with topic modeling method using pins' description of the board. Liu [17] compared three types of traditional recommendation algorithms on Pinterest with the goal of recommending pins to user. His work showed that content-based method has the best performance and the bi-graph recommendation has good performance as well because of the sparsity of the relationship between pins.

Possibly because the CCSN is a relatively new form of social networks, most reported research of CCSNs has focused on the statistical overview of user demographics and behaviours [2, 8, 20], contents [2, 10] and the differences between CCSNs and the text-based social networks like Twitter [19]. Although the statistical overview analysis of CCSNs [6, 8] has shown that homophily drives “repin” and ‘Like’, ‘Comment’ and ‘Follow’ lead to the ‘repin’ behavior of users, none of the existing studies analysed the user repin behaviour and use it for followee recommendation.

Conventionally rating-based collaborative filtering methods have been based on using the previous ratings between users and service items rather than relationships between users [16]. Memory-based collaborative filtering is based on some similarity measure between pairs of users for prediction [4, 21, 22, 26]. The memory-based approaches have scalability issues for some application domains. Model-based approaches to collaborative filtering are based on linear algebra like SVD, PCA or Eigenvectors [5, 23] or machine learning [3, 13].

3. THE PROBLEM

On a CCSN, people can collect an interesting pin (item) by the behavior ‘repin’. With this behavior, a copy of the source pin is created and added to one's board. If a user edits or removes a pin, its ancestor or child pins will be kept as the original pins. There is a data element recording the source for pins. Owing to the source link, we can construct a ‘repin path’ from a pin to the original pin. Each node on the ‘repin path’ is represented by the owning user ID.

Let $R = \{U_0, U_1, U_2, ...U_M\}$ be a repin path, where U_0 is the current user. In this repin path, user U_0 re-pinned this item (pin) from user U_1 directly, but from other users such $U_2, U_3...U_M$ indirectly. Here, we define user U_1 as the direct user of user U_0 , and other users are the indirectly users. In some cases, user U_0 repins many pins from one of indirect users such as user U_m, we can estimate that user U_0 and user U_m possibly have common interests. But user U_0 does not know user U_m. On the other hand, user U_0 has captured the information of user U_m partially and indirectly. His knowledge about user U_m is limited by the taste of the middle users between them such as $U_2, U_3...U_{m-1}$. Therefore, it is necessary and meaningful to recommend user U_m to user U_0.

Given a target user u, the problem of recommendation of users for u to follow is to recommend users that s/he has re-pinned a lot of pins from them indirectly. The problem is represented as:

$$Recommendation(u) = topnmax_r\{r_{u^d_j} | r_{u^d_j} \in \Theta_u\} \quad (1)$$

where, the recommendation to the target user u is generated by the top N maximum of $r_{u^d_j}$ which stands for the recommendation score calculated by our algorithm. Θ_u stands for the set of the candidate recommended users, which are users shared by at least one pair of repin paths.

Let suppose that user u has total n pins, then u can be represented as n repin paths:

$$U_u = \{R_1, R_2, ..., R_n\} \quad (2)$$

where, R_i represents a repin path of length n_i. If we use the user ID to represent the user at the node of the ‘repin path’, the ‘repin path’ could be treated as a set of user IDs.

$$R_i = \{u_{i0}, u_{i1}, ..., u_{in_i}\} \quad (3)$$

let $S_{i,j}$ represent the intersection of ‘repin path’ R_i and R_j.

$$S_{i,j} = R_i \cup R_j \quad (4)$$

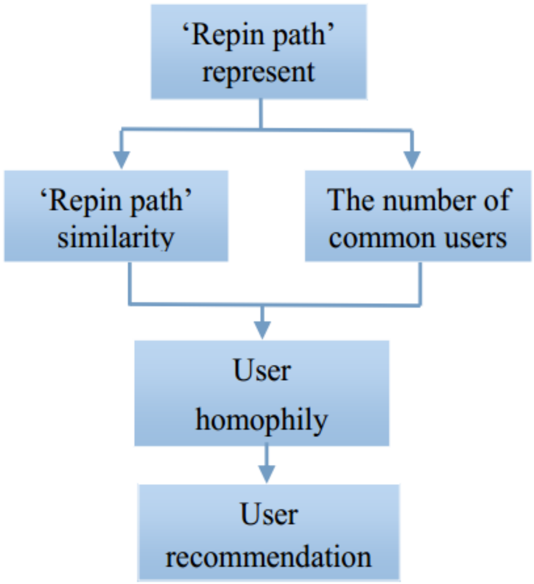


Figure 1: The framework of the proposed algorithm

For user u, we could get the candidate user set Θ_u of user u as follows.

$$\Theta_u = \bigcup_{i=1 to\ n-1; j=2 to\ n} S_{i,j} \tag{5}$$

4. RECOMMENDATION BY LIGHTWEIGHT COLLABORATIVE FILTERING

4.1 Framework of the recommendation scheme

Given a user u, our goal is to discover and recommend other users whom user u is possibly to follow but does not know. Such users should have common interests with u. From this point, if users are included in more repin paths of user u, very likely they have more common interests with u. Furthermore, the different information flow on repin paths possibly presents different interests. For example, the repin path $U_A \rightarrow U_B \rightarrow U_C$ represents that a pin gets to user u from User U_A, by User U_B and User U_C. While the repin path $U_C \rightarrow U_B \rightarrow U_A$ represents the item is start from User U_C to User u by U_B and U_A. Because the sources of two pins are different, they are possibly different although they share the common users. Therefore, the intuitive popularity based scheme to counting the number of shared users of repin paths can not present such difference. In section 4.2, we use Levenshtein Distance to measure the similarity between two repin paths.

Furthermore, if two pin paths have more than one common users, it means that their similarity is represented by all these shared users. The similarity should be distributed to these users with equal weight. In contrast, if two repin paths have only one common user, this user represents the similarity of this pair of repin paths. Therefore, besides the similarity, we also find the common users. From the repin path similarity and the number of common users, we could estimate the user homophily, by which users to follow are recommended. The framework of the proposed algorithm is shown in Figure 1.

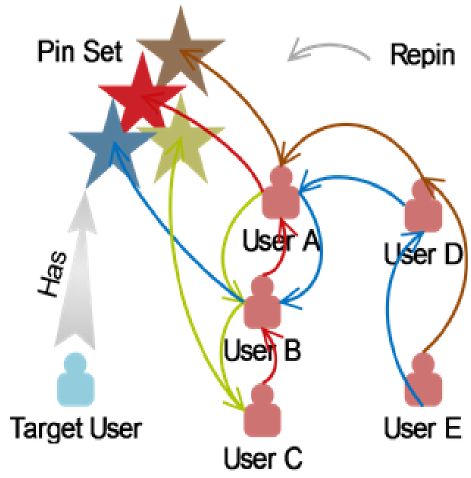


Figure 2: Illustration of user’s repin path

4.2 Measure the similarity between two repin paths and discover common users

In this section we compute the similarity between a pair of repin paths R_i and R_j of user u, and the length of R_i and R_j are n_i and n_j respectively.

The Levenshtein distance (or edit distance) has been widely used in evaluating the differences between two strings or sequences of characters (for example DNA sequences). In this paper, a repin path can be seen as a string, and the user IDs can be considered as characters. The Levenshtein distance can be used to measure the similarity repin paths. The Levenshtein distance between R_i and R_j is defined as follows:

$$L(R_i, R_j) = \begin{cases} n_i, if\ n_j = 0 \\ n_j, if\ n_i = 0 \\ f(n_i, n_j), Otherwise \end{cases} \tag{6}$$

where $f(.)$ is a function defined as

$$\begin{aligned} f(n_i, n_j) = \min(&f(n_i - 1, n_j - 1) + g(R_i[n_i], R_j[n_j]), \\ &f(n_i - 1, n_j), f(n_i, n_j - 1)) \end{aligned} \tag{7}$$

$R_i[n_i]$ and $R_j[n_j]$ are the nodes on the repin path R_i and R_j represented by the user IDs, $g(.)$ is a function defined as

$$g(id_1, id_2) = \begin{cases} 1, if\ id_1 \neq id_2 \\ 0, if\ id_1 = id_2 \end{cases} \tag{8}$$

If two repin paths have many shared user IDs in the same order, it is very likely these two paths are correlated and the users that appear in these two paths have shared interests. Specifically the complementary of the edit distance represents the homophily between the users on these repin paths and user u. Therefore, the similarity between R_i and R_j could be computed from the Levenshtein distance.

Table 1: the similarity between each pair of repin paths in Fig 2

	R_1	R_2	R_3	R_4
R_1	0	1/4	1/3	1/4
R_2	1/4	0	3/4	1/4
R_3	1/3	3/4	0	1/3
R_4	1/4	1/4	1/3	0

Table 2: The common user set of each 'repin path' pair in Fig 2

	R_1	R_2	R_3	R_4
R_1	–	{A,B}	{A}	{A,B,C}
R_2	{A,B}	–	{A,D,E}	{A,B}
R_3	{A}	{A,D,E}	–	{A}
R_4	{A,B,C}	{A,B}	{A}	–

$$sim(R_i, R_j) = \begin{cases} 0, if\ i = j \\ 1 - \frac{L(R_i, R_j)}{\max(n_i, n_j)}, if\ i \neq j \end{cases} \tag{9}$$

In Figure 1, there are 4 repin paths for the Target User. They are represented in Red (R_1), Blue (R_2), Brown (R_3) and Green (R_4) respectively. We could compute their similarity from Equation (9) , as shown in Table 1.

Furthermore, we could search the common user set for each pair of re-pin paths by their shared users. For example, R_1 and R_2 share user A and user B, they have the common user set as $CUS_{R_1,R_2} = \{u_A, u_B\}$. The common user set of each pair of repin path in Figure 2 is shown in Table 2.

4.3 The recommendation algorithm

As in section 3, suppose that there are total n repin paths for user u. The candidate user set for recommendation $\Theta_u = \{u_1, u_2, ..., u_M\}$ can be obtained from Equation (5).

By the above assumption, there are $K = \frac{n(n-1)}{2}$ pairs of repin paths. For the K^{th} pair of repin paths, we could compute the similarity sim_k between them. We could also get the common user set CUS_k. The number of users in the common user set CUS_k is num_k.

We first compute the recommendation score RW_{U_m} for candidate user u_m in Θ_u. Then we rank these users based on their recommendation score. The ranking score for a candidate user U_m in Θ_u can be interpreted as the aggregated homophily between user U_m and user u.

The recommendation algorithms are as follows:

Step 0: Initialization $RW_{U_m} = 0, m = 1, 2, ..., M$

Step 1: $k = 1$.

Step 2: $m = 1$.

Step 3: If $u_m \in CUS_K$

$$RW_{U_m} = RW_{U_m} + sim_k \times \frac{1}{num_k} \tag{10}$$

Step 4: If $(m < M)$ $m = m + 1$ and go to step 3. Otherwise, go to step 5.

Step 5: If $(k < K)$ $k = k + 1$ and go to step 2. Otherwise, go to step 6.

Step 6: Sort all the candidate users by their recommendation weight RW_{U_m} in descending order and select top n candidates for recommendation.

Table 3: percentage of the incomplete 'repin paths'

Cate	Home	Travel & Places	Food & Drink
Incomplete repin paths	4.11% (7927/192705)	2.74% (1541/56288)	4.23% (7244/171148)

The recommendation score of $\{u_A, u_B, u_C, u_D, u_E\}$ in Figure 2 are {23/18, 13/36, 1/9, 1/4, 1/4}, As a result, the users are recommended by the following order:

$$u_A \rightarrow u_B \rightarrow u_D \rightarrow u_E \rightarrow u_C \tag{11}$$

5. EXPERIMENTS

The algorithm is tested using the data from a content curation social network `huaban.com`.

5.1 The Dataset

We crawled data from `http://huaban.com/`. The data set includes 75 users (we said them the observed users), who are randomly sampled from three categories: 'Home', 'Travel & Places' and 'Food & Drink'. For each observed user, his/her followings and pins are collected. For the followings of the observed users, the user ID and the creating time at which the observed users begin to follow them.

For each pin its description and repin path are collected, the repin path is a linked list organized by series pins which includes pin IDs, user IDs (or creator IDs) and creating times. The pin IDs increase as the time, a pin with larger ID is created later than which with smaller ID. The data set includes 420,081 pins and the total length of 'repin paths' is 2,335,087,558.

As discussed earlier 'repin' in CCSNs is different from 'retwitter' in Twitter. If a pin has been deleted, the 'repin path' could not be completely recorded. This will somehow affect the algorithmąŕs precision. So we counted the incomplete 'repin paths' in the dataset in three categories, as shown in Table 3.

It is obvious that the missing data in the data set is less than 5%. Therefore, we think these repin paths would cause little influence to our algorithm, and they are deleted from the data set. The whole data set is separated into two parts. The repin paths of the former 80% pins are selected as the training set. The verification set includes the rest 20% records. In fact, the pin in user collection are order by time, therefore, the former 80% training data corresponds to a cut time t. In another word, the data (the followings, the repin paths of pins) before time t are used for training; the data after the time t are used for validation.

5.2 The similarity of repin path for different users

For a user, we could compute the similarity between each pair of repin paths from Equation (9). Obviously it is diagonally symmetric. we get 8 similarity matrices corresponding to 8 randomly selected users as shown in Figure 3.

In Figure 3 the color changes from "red" to "yellow" to "white" as the similarity value increases. "red" means the similarity is zero, "Yellow" means the middle similarity, "white"

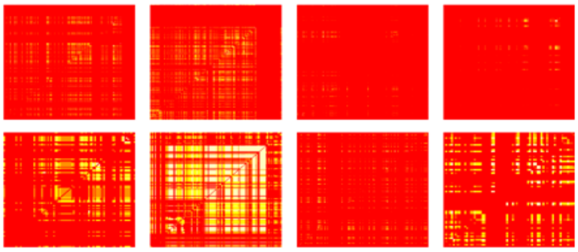


Figure 3: 'repin path' similarity matrices from 8 users (From Top Left to bottom right the corresponding users $u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8$)

means the similarity is maximum. The items(repin paths) are ordered by the time when they were repin by the target user. The target user and their followings are not included in the repin paths. They are frequently the last two nodes in the re-pin paths. From this Figure, It can be observed that the re-pin paths of most users have similarity to some extent. From u_6, to u_5, u_8, u_2, u_1, u_7, u_4, u_3 the similarity values are generally decreased. Furthermore, the distributions of similarity of different users are much different. For some users such as user u_5 and u_6, the high similarity points concentrated in some period. It indicates the strong interest of the corresponding user in this period. For some users such as user u_1 and u_7, the high similarity points distribute across all pins along the time axis, it indicates that the corresponding user has stable interests along the time axis.

In summary, the different distribution of similarity values possibly presents different user interest. Therefore, it is reasonable to recommend user to follow by considering the similarity between repin paths.

5.3 Recommendation Results

Our algorithm was implemented in R and was run on a computer with Windows 7 (Flagship edition, 64-bit), Intel Core i5-3470 3.20GHz and 4.00 GB ram.

For a target user, we evaluated the recommendation result by comparing with the followings of the target user and the users from whom the target user frequently 'repins'. For example, If we recommend user U_m to the target user u, if any of the following cases is satisfied, it is thought a successful recommendation:

1) After the cut time t, the target user u have followed the user U_m;

2) In the testing set of the target user u, there is at least a pin, whose pin path includes user U_m. In our opinion, if the target user repins the pin from user U_m indirectly, it means the target user still follows user U_m indirectly after time t. therefore, user U_m should be recommended to the target user u.

We compared our algorithm (ursrp) with random selection (rnd) recommendation and popularity-based recommendation scheme (pop), where users are ranked by the appearance frequency in 'repin paths'.

The recall, precision and F_1-score are used for evaluation. The comparison results are shown in Figure 4.

Precision: $P = \frac{TP}{TP+FP}$

Recall: $R = \frac{TP}{TP+FN}$

F_1-score: $F_1 = \frac{P \times R}{P+R}$

From the Figure 4 (a) we can see that the Precisions of our algorithm (URSRP) and POP are similar, but Recall of URSRP is much bigger than POP as shown in Figure 4 (b). It demonstrates that URSRP improves the Recall with the almost same precision, compared with POP. And the F- value combining Precision and Recall of our algorithm is bigger than that of POP as shown in Figure 4 (c). From these results, we could say that our algorithm is better than POP. From Figure 4, we also can see that our algorithm is much better than random recommendation.

6. CONCLUSION

In this paper we study the problem of recommending users to follow on content curation social networks. The main contribution of this paper is that we estimate the user recommendation score from the similarity of repin path. The experimental results using the data from Huaban.com show that our scheme is more efficient than random selection and user popularity based recommendation.

This is only a preliminary work on content curation social networks. In future, we will focus on achieving highly accurate recommendation by combining content filtering with collaborative filtering. We will also improve the diversity of recommendations towards serendipitous recommendation for users.

7. REFERENCES

[1] M. G. Armentano, D. Godoy, and A. Amandi. Topology-based recommendation of users in micro-blogging communities. *Journal of Computer Science & Technology*, 27(3):624–634, 2012.

[2] C. Bernardini, T. Silverston, and O. Festor. A pin is worth a thousand words: Characterization of publications in pinterest. In *2014 IntÕl Conf on Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 322–327, 2014.

[3] D. Billsus and M. J. Pazzani. Learning collaborative information filters. In *In Proc. 15th International Conf. on Machine Learning*, pages 46–54, 1998.

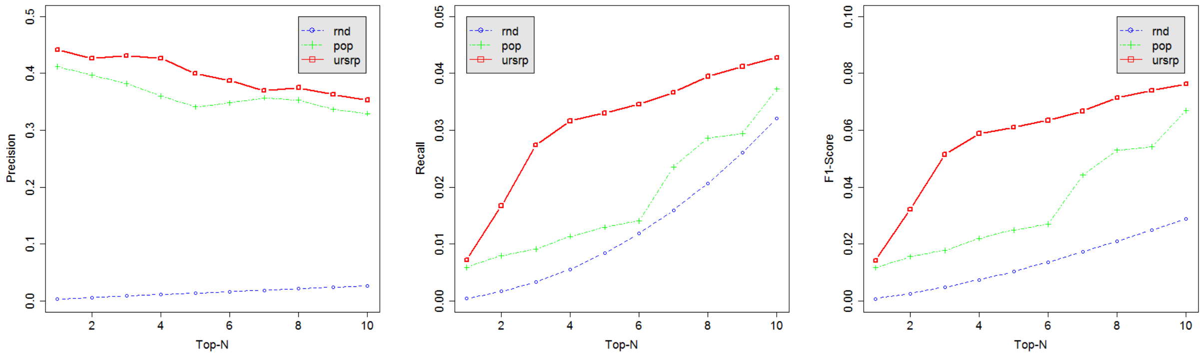
[4] J. S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. *Proceedings of the Conference on Uncertainty in Artificial Intelligence*, 7(7):43–52, 2013.

[5] J. Canny. Collaborative filtering with privacy via factor analysis. In *Proc of the 25th annual IntÕl ACM SIGIR conference on Research and development in information retrieval*, pages 238–245, 2002.

[6] S. Chang, V. Kumar, E. Gilbert, and L. Terveen. Specialization, homophily, and gender in a social curation site: findings from pinterest. In *Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing*, pages 674–686, 2014.

[7] J. Chen, W. Geyer, C. Dugan, M. Muller, and I. Guy. Make new friends, but keep the old: recommending people on social networking sites. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 201–210, 2009.

[8] E. Gilbert, S. Bakhshi, S. Chang, and L. Terveen. I need to try this?: a statistical overview of pinterest. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2427–2436, 2013.



(a) (b) (c)

Figure 4: Recommendation result on the dataset

[9] I. Guy, I. Ronen, and E. Wilcox. Do you know?: recommending people to invite into your social network. In *Proc of the 14th IntÕl conference on Intelligent user interfaces*, pages 77–86, 2009.

[10] C. Hall and M. Zarro. Social curation on the website pinterest.com. *Proceedings of the American Society for Information Science & Technology*, 49(1):1–9, 2012.

[11] J. Hannon, M. Bennett, and B. Smyth. Recommending twitter users to follow using content and collaborative filtering approaches. In *Proceedings of the fourth ACM conference on Recommender systems*, pages 26–30, 2010.

[12] J. Hannon, K. McCarthy, and B. Smyth. Finding useful users on twitter: Twittomender the followee recommender. In *European Colloquium on IR Research*, pages 784–787, 2011.

[13] T. Hofmann and J. Puzicha. Latent class models for collaborative filtering. *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence*, pages 688–693, 1999.

[14] K. Kamath, A. Popescu, and J. Caverlee. Board recommendation in pinterest. *UMAP*, 2013.

[15] Y. Kim and K. Shim. Twitobi: A recommendation system for twitter using probabilistic modeling. In *2013 IEEE 13th International Conference on Data Mining*, pages 340–349, 2011.

[16] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. *Proc of Acm Sigkdd Conference*, pages 426–434, 2008.

[17] D. Liu, C. Lu, and K. Mohan. Pinterest analysis and recommendations. *Course report. Stanford University*, 2014.

[18] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 15(4):344ÍC349, 2001.

[19] R. Ottoni, D. Las Casas, J. Pesce, W. Meira, Jr, C. Wilson, A. Mislove, and V. Almeida. Of pins and tweets: Investigating how users behave across image- and text-based social networks. In *Proceedings of the 8th International AAAI Conference on Weblogs and Social Media (ICWSM'14)*, page 344, 2014.

[20] R. Ottoni, J. Pesce, D. Las Casas, G. Franciscani Jr, W. Meira Jr, P. Kumaraguru, and V. Almeida. Ladies first: Analyzing gender roles and behaviors in pinterest. In *Proceedings of the 7th International AAAI Conference on Weblogs and Social Media (ICWSM'13)*, 2013.

[21] D. M. Pennock, E. J. Horvitz, S. Lawrence, and C. L. Giles. Collaborative filtering by personality diagnosis: A hybrid memory- and model-based approach. *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence*, pages 473–480, 1999.

[22] P. Resnick, N. Iacovou, M. Suchak, P. Bergstrom, and J. Riedl. Grouplens: an open architecture for collaborative filtering of netnews. In *Proceedings of the 1994 ACM conference on Computer supported cooperative work*, pages 175–186, 1994.

[23] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. T. Riedl. Application of dimensionality reduction in recommender system - a case study. In *In ACM WebKDD Workshop*, pages 82–90, 2000.

[24] J. E. A. Stan, J. Stan, P. Maret, V. Do, J. Stan, P. Maret, and V. Do. Semantic user interaction profiles for better people recommendation. In *Proceedings of the 2011 International Conference on Advances in Social Networks Analysis and Mining*, pages 434–437, 2011.

[25] S. Wan, Y. Lan, J. Guo, C. Fan, and X. Cheng. Informational friend recommendation in social media. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, pages 1045–1048, 2013.

[26] S. M. Weiss and N. Indurkhya. Lightweight collaborative filtering method for binary-encoded data. In *Proceedings of the 5th European Conference on Principles of Data Mining and Knowledge Discovery*, pages 484–491, 2001.

[27] S. H. Yang, B. Long, A. Smola, N. Sadagopan, Z. Zheng, and H. Zha. Like like alike: joint friendship and interest propagation in social networks. In *Proc of the 20th Intl Conf on World Wide Web*, pages 537–546, 2011.