Online Social Voting by Collaborative Filtering-Based Recommendation Using Advance Deep-Learning Algorithm

Mr Manjunath N, Mr Guru Prasad M, Mrs Kavya Shree M.S, Mrs Mahalakshmi M C, Prof N KempaRaju

Dept of ISE East Point College of Engineering and Technology

Abstract-The Social Voting offers the opportunity to promote our innovative concept, and to collect votes via social media channels. It is emerging as a new feature in online social media. It sets different challenges for recommendation. In this paper we show how to use a matrix-factorization and nearest-neighbor based recommender systems that investigate user social network and group information for voting recommendation. we also observe that social and group information is much more important than to normal users to regular users. In this paper meta-path based nearest-neighbor models outperform computation-intensive matrix-factorization models in hot-voting recommendation, while users interests for non-hot voting's can be better mined by matrix-factorization models.

Keywords: Social Voting, Collaborative filtering, Recommender systems(RSs), Online social networks(OSNs).

I. INTRODUCTION

The online social network (OSN) such as Facebook, Twitter, instagram etc, & Its an easy way of transferring or sharing data between different users. But the user not only can share his data he can also update his data in forms of audio, video, text and pictures, with a direct user, he can also quickly distribute the updated data to larger user around the globe, where there need not to be a mutual user to one another they just need a knowledge about update which main user has distributed. Now a days many Online Social Network(OSNs) offers us the social voting function, such as organizing or creating any unique event regards to the head user and gives us an options, e.g., like or dislike, on various subjects ranging from user status, profile picture, to game played, product purchased, websites visited, and so on.

Speaking about like and dislike type of voting's some of OSN, e.g., Facebook allow user to initiate their own voting fight on their topics which the user admin are interested with voting options. The friends of the admin user can participate in the voting process or they can give authorized to the use were they have the idea or interest about the post which has been created by the admin user. Social voting also has a political commercial values, their Advertise can create to conduct an voting to conduct an market research.

In the present system there is an increasing in popularity of social voting where they brings "information overload" problem where user can be easily swamp by various voting that were initiated, participated or rewetted by the direct user or indirect user to which there are mutual friends or not to the admin user of the present post which is in for voting recommendation.

It is very difficult and very challenging to present the "right voting" to the "right user" so as to improve user experience and maximize user engagement in social voting. We present recommending interesting voting to the user which differs from the traditional items for a recommendation, such as books, movies, bikes, car and popularity etc. A user is more likely to participate in voting when voting has initiated from the admin .

Toward addressing the challenges, we develop a set of novel Recommended System models, including matrix-factorization based models and nearest-neighbour based models. We systematically evaluate and compare the performance of the proposed models using real social voting traces collected from Facebook. The contribution of this paper is twofold.

1) Online social voting has not been much investigated to our knowledge. We develop matrix-factorization based and nearest-neighbour based Recommended System models.

2) Our experiments on nearest-neighbour based models suggest that social network information dominates group affiliation information. And social and group information is more valuable to cold users than to heavy users.1

II. RELATED WORK

Bond *et al.* experimented about social influence on 61 million persons in Facebook. This experiment says us the following

Human usually spread their thoughts through social networks like Facebook and other medias, but results were reported from a randomized controlled trial of political mobilization messages delivered to 61 million Facebook users during the 2010 US congressional elections, observational studies are difficult to identify in a social networks its effects are studies in and it is unknown to us whether online social networks operate in the same way shown in 14–19. The political self-expression, real-word voting behaviours and information seekers of millions of people were influenced directly, the messages are not only influenced it also influences the friends and friends of friends.

Collaborative-filtering and RSs are principally concerned with two related problems:

Rating prediction and Ranking. The rating prediction goal is to predict the rating of users. The ranking task will help in making digital or hard copies.

Adomavicius and Tuzhilin did an survey on collaborative filtering and RSs. Jiang-et-al presented top-k recommendations. Rendle et al. came up with generic optimization (Opt) which is taken from maximum posterior estimator for personalised ranking there are many pervious other methods to increase recommendation accuracy. Ma et al proposed user's rating which is influenced by his/her friends. Jamali and Ester says that user's interest is claimed by their friends, thus says us that user's latent relationship is similar to his/her latent features. MF Yang et al. claims to split the original social network into circles. Ratings predicted of an item is predicted by different circles under different categories. Jiang et al. He considered of enriching the information by user-item link prediction by representation star-structured hybrid graph. Sun et al uses collaborative nowcasting model along with mobile assistant to increase the accuracy of an item recommendation. In contrast traditional recommendation is different from our online social recommendation in term of social propagation, our model explores following, users group affiliation information in the we come to know about how exactly to improve voting recommendation using social and group information simultaneously.

One-class collaborative filtering (OCCF) handles binary rating data for multiple channels, which will say user-voting activity's action or not, in this only positive actions are observed. As of our knowledge we are the very first to study online social votings recommendation.

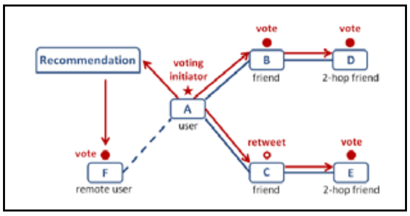

Fig 1: Social Voting Propagation Para diagram.

In the social propagation the user A is the admin and he is the one who can initiate the voting mechanism and based on the latent relationship and matrix factorization the votes will only be recommended to the users who has relationship among the items of matrix factorization.

Imagine user B as a friend of user A and he well known about the item, which user A as initialized voting then this vote will be visible to the user B and he can vote to that question. And this vote can hop 2, that means the friend of B who knows about that item and well aware of that item but not friend of an user A then even he can also answer to that vote as he is well known about the item.

Now user C is friend of user A but the recommendation system will not recommend as he is not in MF as well as he/she will not be knowing about that particular latent item, so it won't display the vote to user C but if friend of user C knows about the latent item then he can vote . this is how our new social voting recommendation system works.

III. SOCIAL VOTING RECOMMENDATION

For effective recommendation we use the following equations and following algorithms for recommendation for cold starters as well as heavy users.

Weibo-MF Model: User latent feature Qu determines the user-voting interaction Ru,i and voting latent feature Pi, user-group interaction Gu,n is determined by user latent feature Qu and group latent feature Yn, and user–user interaction S∗u,v is determined by user latent feature Qu and factor feature Zv.

$$\begin{aligned}&\sum_{\text{all } u}\sum_{\text{all } i} W_{u,i}\left(R^{o\&i}_{u,i} - \hat{R}_{u,i}\right)^2\\ &+\sum_{\text{all } u}\sum_{\text{all } v} W^{(S)}_{u,v}\left(S^{*o\&i}_{u,v} - \hat{S}^{*}_{u,v}\right)^2\\ &+\sum_{\text{all } u}\sum_{\text{all } n} W^{(G)}_{u,n}\left(G^{o\&i}_{u,n} - \hat{G}_{u,n}\right)^2\\ &+\lambda\left(||P||_F^2 + ||Q||_F^2 + ||Y||_F^2 + ||Z||_F^2\right)\end{aligned}$$

Algorithm 1 Algorithm of Weibo-MF Model

Data: Sina Weibo voting dataset
Result: Top-k Hit Rate

```
// Training part
```
Load sina weibo voting training data;
Initialize latent feature matrices Q and P;
```
// Update latent features by ALS
```
while *Not Converge & Iteration Number is less than Iter_Num* **do**
  Update Q by fixing P and minimizing Eq. (5);
  Update P by fixing Q and minimizing Eq. (5);
end
```
// Testing part
```
for *each user u in Sina Weibo voting dataset for testing* **do**
  for *each voting i in test dataset for user u* **do**
    Calculate the predicted rating of user u on voting i as $\hat{R}_{u,i} = r_m + Q_u P_i^T$;
    Put $\hat{R}_{u,i}$ into the queue *recomm_pool*;
  end
  Sort *recomm_pool* in an decreasing order according to the value of $\hat{R}_{u,i}$;
  Select foremost K votings with largest $\hat{R}_{u,i}$ from *recomm_pool* as the items for recommendation;
  Calculate top-k hit rate for user u;
end
Return average top-k hit rate for entire system;

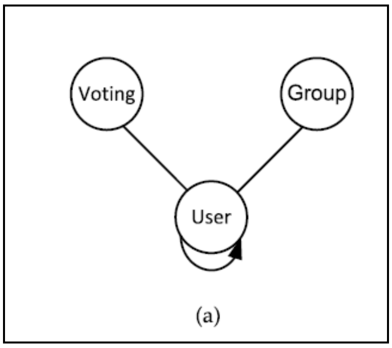

Fig 2a : Weibo heterogeneous information

Fig 2a Shows the schema of Weibo heterogeneous information network. It has three types of objects, namely, user (U), voting (V), and group (G).

We consider some different metapaths for the purpose of NN voting recommendation **2b-2d shows different metapaths**.

The solid lines are social connections, the dashed indicates the lines between users and groups, a user joins a group.

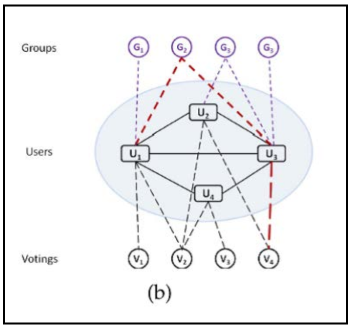

Fig 2b U-G-U-V metapath

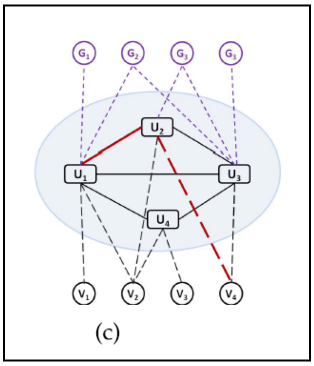

Fig 2c U-U-V metapath.

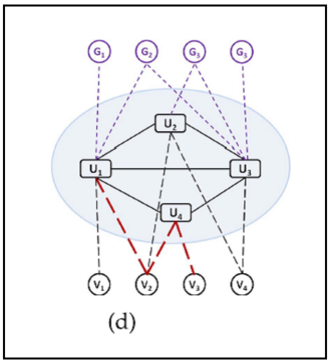

Fig 2d U-V-U-V metapath.

UGUV Metapath : As shown in Fig. 2(b), Algorithm(2) U−G−U−V metapath finds the users in a same group and recommends voting for the target user. More specifically, UGUV works as follows.

Algorithm 2 Algorithm of UGUV Metapath

Data: Sina Weibo voting dataset
Result: Top-k votings for recommendation
Initialization;
for *each target user* u **do**
  Find all groups g's that user u has joined and put them in a set G^u;
  for *each joined group* $g \in G^u$ **do**
    Find all user v's in group g;
    for *each user* v *in group* g **do**
      User v reports its relevant votings and put them in a set I_v;
      for *each candidate voting* $i \in I_v$ **do**
        $Score_{u,i} += w(g)$;
      end
    end
  end
  Sort $\{Score_{u,i}\}$ in a decreasing order;
  Return and recommend top k votings with highest scores to user u;
end

1) Let target user be u, UGUV searches 'u' in all the groups. Denote as Gu.

2) For each joined group, search for all the users that belong to group g. 'g ∈ Gu'

3) Group users g report their relevant votings.

4) Combining the reports of all groups.

UUV(m-hop) Metapath: As in Fig.2(c), Algorithm(3). U−U−V (m−hop) metapath-based recommendation is to recommend a target user the relevant votings of his follow's with in m-hops

Algorithm 3 Algorithm of UUV(m-Hop) Metapath

Data: Sina Weibo voting dataset
Result: Top-k votings for recommendation
Initialization;
for *each target user* u **do**
  Find all followees v's within m-hops by BFS;
  Put all those v's in a set $N_u^{(s)}$;
  for *each user* $v \in N_u^{(s)}$ **do**
    User v reports its relevant votings and put them in a set I_v;
    Set weight parameter $w_s(u, v)$ according to the depth of user v in the BFS tree rooted at user u;
    for *each voting* $i \in I_v$ **do**
      $Score_{u,i} += w_s(u, v)$;
    end
  end
  Sort $\{Score_{u,i}\}$ in a decreasing order;
  Return and recommend top k votings with highest scores to user u;
end

Algorithm 3 of UUV(m-Hop) Metapath

UVUV metapath: As in Fig.2(d), Algorithm(4) the U−V−U−V metapath based recommendation is to recommend their relevant votings for the target user and find users that share votings with the target user.

Algorithm 4 Algorithm of UVUV Metapath

Data: Sina Weibo voting dataset
Result: Top-k votings for recommendation
Initialization;
for *each target user* u **do**
  Find all votings j's that user u has participated;
  Put all those voting j's into a set I_u;
  for *each voting* $j \in I_u$ **do**
    Find all users v's who ever participated in voting j and put them in a set N_j;
    for *each user* $v \in N_j$ **do**
      Find all votings i's that user v has participated and put them in a set I_v;
      for *each voting* $i \in I_v$ **do**
        $Score_{u,i} += w(v)$;
      end
    end
  end
  Sort $\{Score_{u,i}\}$ in a decreasing order;
  Return and recommend top k votings with highest scores to user u;
end

For a target user u, UVUV works as follows.

1) Find all voting that user u has participated in, and this denote as Iu.

2) For each of the, find the set of users who have participated in j. 'j ∈ Iu' Denote by Nj .

3) Each user v ∈ Nj reports all the voting.

4) Aggregate the reports of all users to assign scores to voting.

Neighbourhoods in Latent Feature Space: neighbourhoods in the user latent feature space derived from MF models.

a) UNN: UNN uses MF to obtain the user latent features.
b) VNN: This approach works similarly as UNN. By combining both we get following formulas.

$$\begin{aligned}\mathrm{Score}_{u,i} = {} & \rho_1 \times \sum_{g\in G^u}\sum_{v\in g}\sum_{i} w(g)\delta_{i\in I_v} \\ & + \rho_2 \times \sum_{v\in N_u^{(s)}}\sum_{i} w_s(u,v)\delta_{i\in I_v} \\ & + \rho_3 \times \sum_{j\in I_u}\sum_{v\in N_j}\sum_{i} w(v)\delta_{i\in I_v} \\ & + \rho_4 \times \sum_{v\in N_u}\sum_{i} sim(u,v)\delta_{i\in I_v}\end{aligned}$$

Where ρ1, ρ2, ρ3, and ρ4 denotes the weights of UUV(m-hop), UVUV, UGUV and UNN approaches.

IV. EXPERIMENTS

In this section, we evaluate the proposed MF models and NN models using Sina Weibo voting data set.

A. *Methodology*

The performance of a set of voting RSs using the same trace. It uses a simple popularity-based RS as the baseline model.

• Most Pop**:** This RS recommends the most popular items to users, i.e., the voting's that have been voted by the most numbers of users.

For the Weibo-MF model proposed in (5), we evaluate several variants by setting different weights for social and group information.

1) Voting-MF**:** By setting $\gamma s = 0$ and $\gamma g = 0$ in (5), we Only consider user-voting matrix and ignore social and group information. Note that Voting-MF is essentially the same as All Rank model, which is proposed in [12].
2) Voting + Social-MF: By setting $\gamma s > 0$ and $\gamma g = 0$, we additionally consider social network information on top of Voting-MF.
3) Voting + Group-MF: By setting $\gamma s = 0$ and $\gamma g > 0$, we additionally consider user-group matrix information on top of Voting-MF.
4) Weibo-MF: By setting $\gamma s > 0$ and $\gamma g > 0$, we add both social and group information to Voting-MF.

For NN-based RSs, we evaluate UGUV metapath and UUV(mhop) metapath (with $m = 1, 2$) described in Section IVC1;UNN, VNN described in Section IV-C2; and the hybrid approach described in Section IV-C3 by setting different weights in (14).

B. MF-Based Approaches

We tune the regularization constant λ and the optimal value is 0.5. For the dimensionality, we choose $j0 = 10$.

In Voting-MF model, the parameters that lead to the best top-20 hit rate are: $wm = 0.01$ and $rm = 0$. As expected, Voting-MF significantly out performs the naive popularity based RS. Since user-voting data are binary, impute the missing value of user-voting as $rm < 1$, leading to the same result as $rm = 0$.

Voting + Group-MF, the optimal parameters are $\gamma g = 0.1$,$w(G)\ m = 0.001$, and $gm = 0$.

In Voting + Social-MF, the optimal parameters are $\gamma s = 0.1$,$w(S)m = 0.00005$, and $sm = 0$.

Due to the computation constraints, we only present the results of $j0 = 10$ for all different MF models here.

It is evident that Weibo-MF outperforms all other MF-based approaches, since more information used in the model leads to more prediction power. Regarding the results between Voting-MF and Voting + Social-MF, it is noticed that Voting-MF model is good to represent. Adding social information to Voting-MF leads to additional ten plus present relative gain.Another interesting observation is that Voting + Group-MF and Weibo-MF almost cannot or can only bring limited improvement over Voting + Social-MF approach.

C. NN-Based Approaches

Table shows the top-*k* hit rate for neighbourhood-based methods. The percentage numbers in each cell are the relative improvements over the Most Pop method. Among which UNN is based on user latent features obtained by Voting-MF at *j*0 = 80. In Table, we can see that UGUV + UNN outperforms UNN, and UGUV + UVUV outperforms UVUV.

This suggests that group information is helpful for social voting recommendation. Meanwhile, UGUV + UUV(2-hop) + UNN performs almost the same as UUV(2-hop) + UNN, with top-20 hit rate of 0.175 versus 0.174; and UGUV + UUV (2-hop) + UVUV performs almost the same as UUV(2-hop) + UVUV, with a top-20 hit rate of 0.138 versus 0.139.

TABLE

Top-k Hit Rate COMPARISON FOR VOTING-MF AND NEIGHBORHOOD-BASED METHODS. THE PERCENTAGE NUMBERS IN EACH CELL ARE THE RELATIVE IMPROVEMENTS OVER THE MOSTPOP BASELINE. THE STANDARD DEVIATIONS OF THE RESULTS ARE WITHIN 0.007

Top-K	10	20	50	100
MostPop	0.032	0.050	0.087	0.127
Voting-MF($j_0 = 80$)	0.094 193.8%	0.133 166.0%	0.204 134.5%	0.270 112.6%
UGUV	0.063 96.9%	0.094 88.0%	0.152 74.7%	0.210 65.4%
UUV(1-hop)	0.070 118.8%	0.100 100.0%	0.151 73.6%	0.196 54.3%
UUV(2-hop)	0.071 121.9%	0.107 114.0%	0.176 102.3%	0.246 93.7%
UVUV	0.093 190.6%	0.128 156.0%	0.192 120.7%	0.255 100.8%
UNN	**0.113** 253.1%	**0.155** 210.0%	**0.227** 160.9%	**0.291** 129.1%
VNN	0.041 28.1%	0.061 22.0%	0.097 11.5%	0.132 3.9%
UGUV +UUV(2-hop)	0.076 137.5%	0.114 128.0%	0.185 112.6%	0.256 101.6%
UGUV+UNN	0.117 265.6%	0.165 230.0%	0.248 185.1%	0.323 154.3%
UGUV+UVUV	0.096 200.0%	0.134 168.0%	0.201 131.0%	0.269 111.8%
UUV(2-hop) +UNN	0.125 290.6%	0.174 248.0%	0.262 201.1%	0.342 169.3%
UUV(2-hop) +UVUV	0.099 209.4%	0.139 178.0%	0.209 140.2%	0.279 119.7%
UNN+UVUV	0.120 275.0%	0.164 228.0%	0.242 178.2%	0.315 148.0%
UGUV+UNN +UUV(2-hop)	0.125 290.6%	0.175 250.0%	0.261 200.0%	0.341 168.5%
UGUV+UVUV +UUV(2-hop)	0.099 209.4%	0.138 176.0%	0.209 140.2%	0.279 119.7%
UUV(2-hop) +UNN+UVUV	**0.127** 296.9%	**0.177** 254.0%	**0.264** 203.4%	**0.345** 171.7%

V. CONCLUSION AND FUTURE WORK

In this paper, we come up with two approaches mainly Matrix Factorization-based and NN-based Recommendation System's for online social voting. By experimenting with real data, we come to know that both social and group affiliation information can drastically improve accuracy of popularity based RSs, mainly for new users or cold users and group affiliation information is dominated in NN-based approaches by social network information. This paper we can improve the recommendation accuracy of cold starters then for heavy users, by taking the valuable information from the social and group information.

This is our first approach towards online social voting recommendation. As further we would like study about how this can implemented for an individual user to develop a customized Recommendation system, provided access to all his/her social neighborhoods and activity of an user.

REFERENCES

[1] R. M. Bond et al., "A 61-million-person experiment in social influence and political mobilization," Nature, vol. 489, pp. 295–298,

[2] Sep. 2012.

[3] G. Adomavicius and A. Tuzhilin, "Toward the next generation of Recommender systems: A survey of the state-of-the-art and possible Extensions," IEEE Trans. Known. Data Eng., vol. 17, no. 6, pp. 734–749, Jun. 2005.

[4] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," Adv. Artif. Intell., vol. 2009, Aug. 2009, Art. no. 421425, doi: 10.1155/2009/421425.

A. Paterek, "Improving regularized singular value decomposition for collaborative filtering," in Proc. KDDCup, 2007, pp. 39–42.

[5] R. Salakhutdinov and A. Mnih, "Probabilistic matrix factorization," in Proc. NIPS, vol. 20. 2008, pp. 1257–1264.

[6] K. Yu, A. Schwaighofer, V. Tresp, X. Xu, and H. P. Kriegel, "Probabilistic memory-based collaborative filtering," IEEE Trans. Known. Data Eng., vol. 16, no. 1, pp. 56–69, Jan. 2004.

[7] R. H. Keshavan, A. Montanari, and S. Oh, "Matrix completion from noisy entries," J. Mach. Learn. Res., vol. 11, pp. 2057–2078, Jul. 2010.

[8] P. Cremonesi, Y. Koren, and R. Turrin, "Performance of recommender algorithms on top-N recommendation tasks," in Proc. ACM RecSys, 2010, pp. 39–46.

[9] Y. Zhang, B. Cao, and D.-Y. Yeung, "Multi-domain collaborative filtering," in Proc. 26th Conf. Uncertainty Artif. Intell. (UAI), Catalina Island, CA, USA, 2010, pp. 725–732.

[10] H. Steck, "Training and testing of recommender systems on data missing not at random," in Proc. ACM KDD, 2010, pp. 713–722.

[11] B. Marlin and R. Zemel, "Collaborative prediction and ranking with non-random missing data," in Proc. ACM RecSys, 2009, pp. 5–12.

[12] X. Yang, H. Steck, Y. Guo, and Y. Liu, “On top-k recommendation using social networks,” in Proc. ACM RecSys, 2012, pp. 67–74.

[13] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, “PathSim: Meta pathbased top-k similarity search in heterogeneous information networks,” Proc. VLDB Endowment, vol. 4, no. 11, pp. 992–1003, 2011.

[14] Y. Sun, B. Norick, J. Han, X. Yan, P. S. Yu, and X. Yu, “Integrating meta-path selection with user-guided object clustering in heterogeneous information networks,” in Proc. ACM KDD, 2012, pp. 1348–1356.

[15] R. Pan et al., “One-class collaborative filtering,” in Proc. ICDM, 2008, pp. 502–511.

[16] Y. Li, J. Hu, C. Zhai, and Y. Chen, “Improving one-class collaborative filtering by incorporating rich user information,” in Proc. ACM CIKM, 2010, pp. 959–968.

[17] Sindhwani, S. S. Bucak, J. Hu, and A. Mojsilovic, “One-class matrix completion with low-density factorizations,” in Proc. IEEE ICDM, Dec. 2010, pp. 1055–1060.

[18] Weibo. [Online]. Available: http://weibo.com/

[19] By the Numbers: 58 Amazing Weibo Statistics. (May 2016).[Online].Available:http://expandedramblings.com/index.php/weibouser-statistics/

[20] Sina Vote, accessed on 2013. [Online]. Available:http://vote.weibo.com/

[21] Twitter, accessed on 2017. [Online]. Available: http://twitter.com

[22] Facebook, accessed on 2013. [Online]. Available:

[23] http://www.facebook.com

[24] M. Richardson and P. Domingos, “Mining knowledge-sharing sites forviral marketing,” in Proc. ACM KDD, 2002, pp. 61–70.

[25] M. Jamali and M. Ester, “Using a trust network to improve top-Nrecommendation,” in Proc. ACM RecSys, 2009, pp. 181–188.

[26] K. Järvelin and J. Kekäläinen, “Cumulated gain-based evaluation of IRtechniques,” ACM Trans. Inf. Syst., vol. 20, no. 4, pp. 422–446, 2002.

[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

[28] networks,” in Proc. WWW, 2011, pp. 537–546. [29] Y. Shi, A. Karatzoglou, L. Baltrunas, M. Larson, N. Oliver, and Hanjalic, “CLiMF: Learning to maximize reciprocal rank with collaborative less-is-more filtering,” in Proc. ACM RecSys, 2012, pp. 139–146.

[29] X. Yang, H. Steck, and Y. Liu, “Circle-based recommendation in online social networks,” in Proc. ACM KDD, 2012, pp. 1267–1275.

[30] M. Jamali and M. Ester, “A matrix factorization technique with trust propagation for recommendation in social networks,” in Proc. ACM RecSys, 2010, pp. 135–142.

[31] H. Ma, H. Yang, M. R. Lyu, and I. King, “SoRec: Social recommendationusing probabilistic matrix factorization,” in Proc. ACM CIKM,2008, pp. 931–940.

[32] H. Ma, I. King, and M. R. Lyu, “Learning to recommend with social trust ensemble,” in Proc. SIGIR, 2009, pp. 203–210.

[33] H. Ma, D. Zhou, C. Liu, M. R. Lyu, and I. King, “Recommender systems with social regularization,” in Proc. ACM WSDM, 2011, pp. 287–296.

[34] J. Noel et al., “New objective functions for social collaborativefiltering,” in Proc. 21st Int. Conf. World Wide Web (WWW), 2012, pp. 859–868.

[35] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, “BPR: Bayesian personalized ranking from implicit feedback,” in Proc. 25th Conf. Uncertainty Artif. Intell. (UAI), 2009, pp. 452–461.

[36] R. Jäschke, L. Marinho, A. Hotho, L. Schmidt-Thieme, and G. Stumme, “Tag recommendations in folksonomies,” in Proc. 11th Eur. Conf. Principles Practice Known. Discovery Databases (PKDD), 2007,pp. 506–514.

[37] M. Jiang, P. Cui, N. J. Yuan, X. Xie, and S. Yang, “Little is much: Bridging cross-platform behaviors through overlapped crowds,” in Proc. 13th AAAI Conf. Artif. Intell., 2016, pp. 13–19.

[38] M. Jiang, P. Cui, X. Chen, F. Wang, W. Zhu, and S. Yang, “Social recommendation with cross-domain transferable knowledge,” IEEE Trans. Known. Data Eng., vol. 27, no. 11, pp. 3084–3097, Nov. 2015.

[39] Y. Sun, N. J. Yuan, X. Xie, K. McDonald, and R. Zhang, “Collaborative nowcasting for contextual recommendation,” in Proc. 25th Int. World Wide Web Conf. (WWW), 2016, pp. 1407–1418.

[40] H. Gao, J. Tang, X. Hu, and H. Liu, “Content-aware point of interest recommendation on location-based social networks,” in Proc. 29th AAAI Conf. Artif. Intell. (AAAI), 2015, pp. 1721–1727.