RESEARCH ARTICLE

A converging reputation ranking iteration method via the eigenvector

Xiao-Lu Liu[1]*, **Chong Zhao**[2]*

1 School of Management Science and Engineering, Shandong University of Finance and Economics, Jinan, PR China, **2** School of Mathematics, Shandong University, Jinan, PR China

* liuxiaoluliuxiaolu@163.com (XLL); chong.zhao@sdu.edu.cn (CZ)

Abstract

Ranking user reputation and object quality in online rating systems is of great significance for the construction of reputation systems. In this paper we put forward an iterative algorithm for ranking reputation and quality in terms of eigenvector, named EigenRank algorithm, where the user reputation and object quality interact and the user reputation converges to the eigenvector associated to the greatest eigenvalue of a certain matrix. In addition, we prove the convergence of EigenRank algorithm, and analyse the speed of convergence. Meanwhile, the experimental results for the synthetic networks show that the AUC values and Kendall's τ of the EigenRank algorithm are greater than the ones from the IBeta method and Vote Aggregation method with different proportions of random/malicious ratings. The results for the empirical networks show that the EigenRank algorithm performs better in accuracy and robustness compared to the IBeta method and Vote Aggregation method in the random and malicious rating attack cases. This work provides an expectable ranking algorithm for the online user reputation identification.

OPEN ACCESS

Citation: Liu X-L, Zhao C (2022) A converging reputation ranking iteration method via the eigenvector. PLoS ONE 17(10): e0274567. https://doi.org/10.1371/journal.pone.0274567

Editor: Tatsuro Kawamoto, National Institute of Advanced Industrial Science and Technology, JAPAN

Received: October 12, 2021

Accepted: August 30, 2022

Published: October 3, 2022

Copyright: © 2022 Liu, Zhao. This is an open access article distributed under the terms of the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: The data underlying the results presented in the study are available from GroupLens Research Group (http://www.grouplens.org) and Netflix Inc (http://www.netflixprize.com).

Funding: This work is supported by the National Natural Science Foundation of China (Grant Nos. 12071253, 72101135), the Science Foundation of Ministry of Education of China (Grant No. 18YJC630102), the Natural Science Foundation of Shandong Province of China (Grant No. ZR2020QG008) and the Young Scholars Program

Introduction

User reputation measures the user ability of rating accurately to various objects. In the interpretation of online rating systems, one of the central problems is how to construct the personal reputation systems [1–3], which could have an impact on e-commerce [4, 5], recommender systems [6, 7], rumor spreading [8, 9], etc. In the past several years, online rating systems of many platforms provide user channels to express their preferences to different objects. Nevertheless, not every user gives reasonable/accurate ratings due to his/her dishonesty or non-familiarity [2, 10]. Inaccurate ratings affect the normal sales of online businesses, endangering the healthy development of the social economy. Therefore, measuring the online user reputation according to their rating behaviors is crucial for the maintenance of a good market order and construction of strong cyberpower [11–14].

In the existing works, the iteration-oriented mechanisms have been widely explored, Pagerank [15] and HITS algorithms [16] as representatives. In recent years, quality-based ranking methods are introduced, in which each object is assumed to have an inherent quality and the user reputation is characterized by the relations between his/her rating vector and the

of Shandong University. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

corresponding objects' calculated quality vector. User reputation and object quality are interdependent and are updated iteratively until they become stable. Laureti *et al.* [17] raised an iterative refinement (abbr. IR) method by analyzing the difference between user rating and object quality vectors. Similarly, there is improved IR method [18]. Zhou *et al.* [19] proposed a correlation-based ranking (abbr. CR) method in terms of Pearson correlation coefficient between vectors of user rating and object quality. Similarly, there are IARR method [20], IRUA method [21], etc. [22]. Besides, Allahbakhsh *et al.* [23] proposed a Vote Aggregation method by computing the grade of credibility, and assigning each user a degree of consistence with the community sentiment. Along this line, there are group-based ranking method [24] and so on [25–27]. In addition, Liu *et al.* [28] designed a reputation ranking algorithm via the beta distribution (RBPD for short), in which the user reputation is determined by the probability that the user is to rate fairly. And IBeta method [29] is proposed by introducing an iterative reputation-allocation process based on the RBPD method.

Among the existing reputation ranking methods, the iterative algorithms could measure the user reputation in good accuracy. However, many iterative methods do not converge, or omit the proof of theoretical convergence [19, 20, 29]. It is expected that iterative algorithm with definite convergence can lead to more robust reputation lists [26, 30]. Inspired by this thought, in this paper we develop an iterative reputation and quality ranking algorithm based on eigenvector analysis, called the EigenRank algorithm, where the user reputation and object quality are interdependent and the user reputation vector series converges to the eigenvector corresponding to the largest eigenvalue of a certain matrix. In addition, we give the proof of the convergence of EigenRank algorithm, and analyse the speed of convergence. Meanwhile, we explore the ranking performance of the EigenRank algorithm for the synthetic and empirical networks compared with the IBeta method and Vote Aggregation method. Our results for the synthetic networks imply that the AUC values and Kendall's τ of the EigenRank algorithm exceed their counterparts obtained from the IBeta method and Vote Aggregation method with different proportions of random/malicious ratings. The results for the empirical networks show that the EigenRank algorithm has a better performance in accuracy and robustness than the IBeta method and Vote Aggregation method in the random and malicious rating attack cases.

We organize this paper as follows. The EigenRank algorithm is presented in section II. In sections III and IV, we show the experimental results for the synthetic networks and empirical networks, respectively. Finally, section V gives the conclusion and discussions. The analysis of convergence for the EigenRank algorithm is put in the Supporting Information.

The EigenRank algorithm

We use a weighted bipartite network $G = \{U, O, E\}$ to represent the rating system, where U is the user set, O is the object set and E is the rating set. For the sake of clarity, Latin and Greek letters are used to represent attributes of users and objects, respectively. We use $r_{i\alpha}$ to denote the rating given by user i to object α, and list the ratings into a matrix $\mathbf{A}$, called the rating matrix. It is noted that, in the case the range of ratings varies with object, the ratings should be linearly normalized into a fixed interval, for instance [0, 1], before constructing the matrix $\mathbf{A}$. The letters used in this paper are shown in Table 1.

Firstly, we normalize the ratings as follows. Because different users have different rating criteria, part of them tend to give high ratings and others give low ones, therefore ratings do not reflect the real opinion of the users to objects precisely. To balance this deviation, we need to

Table 1. Descriptor table of letters used in this paper.

Letters	Descriptor
$r_{i\alpha}$	the rating given by user i to object α
d_i	degree of user i
k_α	degree of object α
A	the rating matrix
B	the normalized rating matrix
R	user reputation vector
Q	object quality vector
D	diag$\{d_1, d_2, \ldots, d_{\vert U\vert}\}$
K	diag$\{k_1, k_2, \ldots, k_{\vert O\vert}\}$

https://doi.org/10.1371/journal.pone.0274567.t001

make some normalization, the rating $r_{i\alpha}$ is transformed to the extent of fanciness $r'_{i\alpha}$,

$$r'_{i\alpha} = \begin{cases} 2(r_{i\alpha} - r_i^2)/(r_i^1 - r_i^2) - 1, & r_i^1 \neq r_i^2 \\ 0, & r_i^1 = r_i^2 \end{cases}, \tag{1}$$

where r_i^1 and r_i^2 are the maximum and minimum ratings user i gives, respectively. Thus, all the ratings given by a specific user are normalized to [-1, 1], so that the extreme ratings are set 1 and -1, respectively. Particularly, the normalized ratings are all 0 if and only if the user gives constant rating. Positive $r'_{i\alpha}$ reflects positive opinion and negative $r'_{i\alpha}$ represents negative opinion. The normalized rating are arranged in a matrix **B**, whose (i, α)-th entry is $r'_{i\alpha}$.

Initially, the user reputation $\mathbf{R}^{(0)}$ is assigned in terms of the user degree, $R_i = k_i/|O|$ (where $|O|$ is the number of objects). Object quality is proportional to the received average normalized rating, weighted by user reputation.

We tempt to express the object quality as

$$\mathbf{Q} = c_1 f(\mathbf{K})\mathbf{B}^{\mathrm{T}}\mathbf{R}, \tag{2}$$

where f is a continuous monotone function on $[0, \infty)$, and $c_1 > 0$ is an adjustable parameter that does not influence the relative rank of object quality **Q**. To determine this function f, imagine that if each user has a brother, whom gives the same rating as i did to each object. In this case, the relative rank of the objects should not be disturbed, and the rating matrix becomes $\begin{pmatrix} \mathbf{B} \\ \mathbf{B} \end{pmatrix}$ instead of **B**, and **R** is replaced by $\begin{pmatrix} \mathbf{R} \\ \mathbf{R} \end{pmatrix}$. Then formula (2) becomes

$$\mathbf{Q} = c'_1 f(2\mathbf{K})\mathbf{B}^{\mathrm{T}}\mathbf{R}, \tag{3}$$

for some new constant c'_1. Comparing (3) with (2) conclude

$$f(2k_\alpha) = cf(k_\alpha), \forall \alpha \in O \tag{4}$$

for some constant c independant on α. Suppose $c = 2^{u_1}$, then (4) can be reformed as

$$(2k_\alpha)^{-u_1} f(2k_\alpha) = k_\alpha^{-u_1} f(k_\alpha), \forall \alpha \in O. \tag{5}$$

Suppose f satisfies $(2x)^{-u_1} f(2x) = x^{-u_1} f(x), x \in [0, \infty)$ and $v = \lim_{x\to 0+} x^{-u_1} f(x)$ exists, then for each $x > 0$ we have $x^{-u_1} f(x) = \lim_{n\to\infty} (2^{-l}x)^{-u_1} f(2^{-l}x) = v$, therefore $x^{-u_1} f(x) \equiv v$, i.e.

$f(x) = vx^{u_1}, \forall x$. Consequently, the object quality can be expressed as

$$\mathbf{Q} = c_1 \mathbf{K}^{u_1} \mathbf{B}^{\mathrm{T}} \mathbf{R}. \tag{6}$$

User reputation is measured by the relationship between the user's normalized rating and object quality. When users have a positive view to the object, the larger is the object quality, the higher is the user reputation; Conversely, when the user has a negative view of the object, the smaller is the object quality, the higher is the user reputation. From this viewpoint, the user reputation can be expressed by

$$\mathbf{R} = c_2 \mathbf{D}^{u_2} \mathbf{B} \mathbf{Q}, \tag{7}$$

where c_2, u_2 are adjustable parameters, and $c_2 > 0$ does not change the relative rank of user reputation **R**.

When $u_2 = 0$, the opinions of inactive users will be covered up by those of active ones; meanwhile, When $u_2 = -1$, the opinions of active and inactive users have the same weight. These two choices are too extreme, hence we make a compromised choice, $u_2 = -\frac{1}{2}$ in our algorithm. Similarly, u_1 is also set $-\frac{1}{2}$.

At each step, the vectors **R** and **Q** will be updated according to formulae (6) and (7). The reputation vector after the n-th iteration step is recorded as $\mathbf{R}^{(n)}$. As $n \to \infty$, $\mathbf{R}^{(n)}$ converges to the unit eigenvector associated to the greatest eigenvalue of a certain matrix, and we put its proof in the S1 Appendix. The iteration process will not stop until the reputation vector becomes stable, i.e. the distance $||\mathbf{R}^{(n)} - \mathbf{R}^{(n-1)}||_2^2$ between the reputation vectors decays to below the threshold $\delta = 10^{-10}$, where

$$||\mathbf{R}^{(n)} - \mathbf{R}^{(n-1)}||_2^2 = \frac{1}{|U|} \sum_{i \in U} (\mathbf{R}_i^{(n)} - \mathbf{R}_i^{(n-1)})^2. \tag{8}$$

The numerical convergence of EigenRank algorithm is theoretically guaranteed, of which the proof is put in the S1 Appendix. We discuss the parameters c_1 and c_2. Substituting formula (6) into formula (7), we get

$$\mathbf{R} = c_1 c_2 \mathbf{D}^{-\frac{1}{2}} \mathbf{B} \mathbf{K}^{-\frac{1}{2}} \mathbf{B}^{\mathrm{T}} \mathbf{R}. \tag{9}$$

One can find that **R** is an eigenvector associated to the largest eigenvalue of the matrix $\mathbf{D}^{-\frac{1}{2}} \mathbf{B} \mathbf{K}^{-\frac{1}{2}} \mathbf{B}^{\mathrm{T}}$. Let λ_m denote the largest eigenvalue of matrix $\mathbf{D}^{-\frac{1}{2}} \mathbf{B} \mathbf{K}^{-\frac{1}{2}} \mathbf{B}^{\mathrm{T}}$, then $\mathbf{D}^{-\frac{1}{2}} \mathbf{B} \mathbf{K}^{-\frac{1}{2}} \mathbf{B}^{\mathrm{T}} \mathbf{R} = \lambda_m \mathbf{R}$, we find that the parameters c_1 and c_2 satisfy,

$$c_1 c_2 = \frac{1}{\lambda_m}. \tag{10}$$

As mentioned before, the values of c_1, c_2 do not influence the relative rank of reputation **R**. In fact, what we need is the direction of the vector **R**, which determines the relative ratios of its coordinates, rather than its length. In each step of iteration, we normalize $\mathbf{R}^{(n)}$ in the 2-norm, which do not impact on its direction, and could make the sequence $\mathbf{R}^{(n)}$ converge. The procedure of EigenRank Algorithm is summarized in Algorithm 1.

The convergence analysis of EigenRank algorithm can be seen in the S1 Appendix. We find that the iteration stops when the number of steps $n > \log_\eta \frac{x_1^{(0)} \delta}{16||\mathbf{D}^{-\frac{1}{4}}|| \cdot ||\mathbf{D}^{\frac{1}{4}}|| \cdot ||x^{(0)}||_2} + 1$. In addition, the total complexity of the EigenRank algorithm is $O(|U||O|\log_\eta \delta)$.

Algorithm 1 The iterative EigenRank Algorithm

```
Input: Rating matrix A;
```

```
Output: User reputation matrix R and object quality matrix Q;
Make normalization of ratings;
Initialize $\mathbf{R}^{(0)}$ in terms of user degree;
while $||\mathbf{R}^{(n)} - \mathbf{R}^{(n-1)}||_2^2 < \delta = 10^{-10}$ is not met do
    $\mathbf{Q} \leftarrow \mathbf{K}^{-\frac{1}{2}}\mathbf{B}^{\mathrm{T}}\mathbf{R}$;
    $\mathbf{R} \leftarrow \mathbf{D}^{-\frac{1}{2}}\mathbf{BQ}/||\mathbf{D}^{-\frac{1}{2}}\mathbf{BQ}||_2$;
end
return R and Q
```

Results for synthetic networks

Moreover, we analyze the ranking performance of the EigenRank algorithm compared to the IBeta method and Vote Aggregation method for the synthetic networks, in which we firstly add the weighted links by employing the preferential attachment mechanism [31] and then insert different number of distorted ratings.

When we generate the synthetic networks, the numbers of users and objects are set $|U|$ = 6000 and $|O|$ = 6000, respectively. The network sparsity η is set to 0.02, 0.03, respectively. We add the weighted links (ratings) one after one until the total cardinality of the ratings $|E|$ reach $\eta \times |U||O| = 7.2 \times 10^5$, 1.08×10^6, respectively. The nodes (user and object) of links are selected in terms of the node degree preferentially. The possibility of selecting user i and object α at each time step t are formulated as

$$p_i(t) = \frac{k_i(t) + 1}{\Sigma_{j \in U}(k_j(t) + 1)}, \tag{11}$$

$$p_\alpha(t) = \frac{k_\alpha(t) + 1}{\Sigma_{\beta \in O}(k_\beta(t) + 1)}, \tag{12}$$

where $k_i(t)$ is the degree of user i at the time step t and $k_\alpha(t)$ is the degree of object α at the time step t.

Two ingredients contribute to the rating $r_{i\alpha}$, the link weight from user i to object α: the object inherent quality Q'_α, and the rating error $\Delta\delta_{i\alpha}$. Here Q'_α subjects to uniform distribution $U(1, 5)$ and $\Delta\delta_{i\alpha}$ is drawn from the normal distribution $N(0, \Delta\delta_i^2)$, in which $\Delta\delta_i$ is the rating error of user i, subjecting to $U(1, 5)$. The rating $r_{i\alpha}$ is calculated by

$$r_{i\alpha} = [Q'_\alpha + \Delta\delta_{i\alpha}], \tag{13}$$

where [ ] represents the closest integer to $Q'_\alpha + \Delta\delta_{i\alpha}$. The rating $r_{i\alpha}$ is limited to the set {1, 2, 3, 4, 5}, and we will truncate the ratings beyond {1, 2, 3, 4, 5}.

Then, we replace part of the links by the distorted ratings. In the synthetic networks, we assume two kinds of distorted ratings simultaneously exist in real online rating systems, those are, random ratings and malicious ones. The former are given by users who rate objects totally randomly in the set {1, 2, 3, 4, 5}, while the latter arise when some of users always give extreme ratings to push up or down certain target objects. In the synthetic networks, we replace ρ proportion of the original links by the distorted ratings (random/malicious case). Larger parameter ρ reflects more noisy information. $\rho = 0$ represents there is all true information and $\rho = 1$ means totally chaos. In our synthetic experiments, the proportion ρ is set to 0.025, 0.05, . . ., 0.5, successively.

To characterize the ranking performance of the EigenRank algorithm, we introduce two indices: AUC curve [32] (the area under a receiver operating characteristic curve) and Kendall's τ [33]. When calculating the AUC value, we need to divide the objects into two subsets, i.e. the benchmark objects and non-benchmark ones, according to their qualities. After n times of independent comparisons, there are n_1 times that the benchmark one has a higher quality than its opponent, and n_2 times they reach a draw. The AUC value is calculated by

$$AUC_s = \frac{2n_1 + n_2}{2n}, \tag{14}$$

where the parameter $n = 1 \times 10^9$ in the experiments. When $AUC_s = 1$, the selected benchmark object has a higher rank than its opponent in all the comparisons. While $AUC_s = 0.5$ implies all the objects are randomly ranked. In the synthetic networks, we select 20% highest-quality objects as benchmark objects based on their inherent qualities Q'_α. The higher is the AUC value, the more accurate is the ranking of object quality.

Another index Kendall's τ computes the rank correlation between the inherent quality Q' and the obtained object quality Q, which is formulated by

$$\tau = \frac{2}{|O|(|O| - 1)} \sum_{\mu<v} sgn[(Q'_\mu - Q'_v)(Q_\mu - Q_v)], \tag{15}$$

where $sgn(x)$ is the sign function. $(Q'_\mu - Q'_v)(Q_\mu - Q_v) > 0$ means Q' and Q is concordant and negative indicates discordant. τ always lies in $[-1, 1]$ and higher Kendall's τ indicates higher ranking accuracy.

We compare the ranking performance of the EigenRank algorithm to IBeta method and Vote Aggregation method. Considering Vote Aggregation method has parameters, in our experiments we set the parameters the same as in the examples from [23].

Fig 1 shows the AUC values AUC_s and Kendall's τ of the EigenRank algorithm, IBeta method and Vote Aggregation method via different proportions of distorted ratings (random/malicious case) for the synthetic networks. From Fig 1(a) and 1(b) one sees that both AUC_s and Kendall's τ of the EigenRank algorithm are larger than their counterparts in the IBeta method and Vote Aggregation method for different parameter ρ in the random rating attack case when $|E| = 7.2 \times 10^5$. For example, when $\rho = 0.1$, the AUC_s of the EigenRank algorithm, IBeta method and Vote Aggregation method could reach 0.906, 0.888 and 0.869, respectively, and the corresponding τ could reach 0.662, 0.615 and 0.574, respectively. From Fig 1(c) and 1(d) we observe that both AUC_s and τ of the EigenRank algorithm are greater than the ones from the IBeta method and Vote Aggregation method for different ρ in the malicious rating attack case when $|E| = 7.2 \times 10^5$. For instance, when $\rho = 0.1$, the AUC_s of the EigenRank algorithm, IBeta method and Vote Aggregation method reach 0.902, 0.883 and 0.862, respectively, and the corresponding τ reach 0.651, 0.602 and 0.559, respectively. One also finds that both AUC_s and τ of the EigenRank algorithm exceed their counterparts in the IBeta method and Vote Aggregation method with different ρ, not only in the random rating attack case Fig 1(e) and 1(f) but also in the malicious rating attack case Fig 1(g) and 1(h) when $|E| = 1.08 \times 10^6$. For example, when $\rho = 0.2$ in the random rating attack case, the AUC_s of the EigenRank algorithm, IBeta method and Vote Aggregation method reach 0.910, 0.897 and 0.879, respectively. When $\rho = 0.2$ in the malicious rating attack case, the τ of the EigenRank algorithm, IBeta method and Vote Aggregation method could reach 0.656, 0.610 and 0.561, respectively. The results imply that the EigenRank algorithm measures the user reputation and object quality more accurately than the IBeta method and Vote Aggregation method.

Results for empirical networks

Furthermore, we explore the ranking performance of the EigenRank algorithm compared with the IBeta method and Vote Aggregation method via two empirical data sets which contain

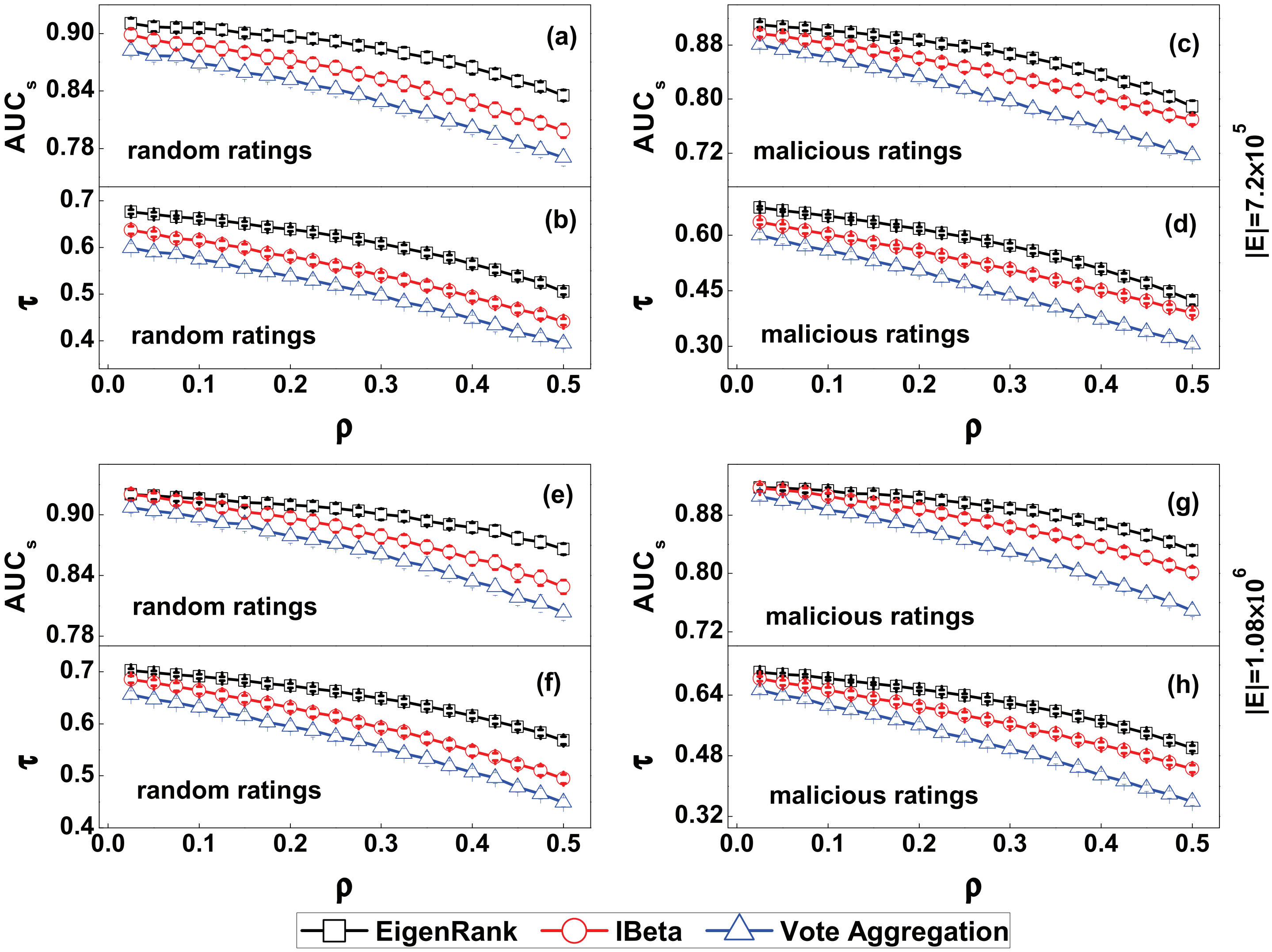

Fig 1. (Color online) AUC_s and τ of the EigenRank algorithm, IBeta method and Vote Aggregation method for the synthetic networks: (a, b) in the random rating attack case when $|E| = 7.2 \times 10^5$; (c, d) in the malicious rating attack case when $|E| = 7.2 \times 10^5$; (e, f) in the random rating attack case when $|E| = 1.08 \times 10^6$; (g, h) in the malicious rating attack case when $|E| = 1.08 \times 10^6$. The parameter ρ denotes the ratio of random/malicious ratings. It can be seen that both the AUC_s and τ of the EigenRank algorithm exceed their counterparts in the IBeta method and Vote Aggregation method with different ρ and $|E|$. The results are averaged over 50 independent realizations. The error bars are the corresponding standard deviations.

https://doi.org/10.1371/journal.pone.0274567.g001

ratings given from users to movies: MovieLens and Netflix. The MovieLens data is downloaded from the GroupLens (http://www.grouplens.org), and the Netflix data is provided from the Netflix Prize (http://www.netflixprize.com). Both of two empirical data sets are extracted from the original data and have 1000209 and 824802 ratings for the MovieLens and Netflix data set, respectively. Each user has at least 20 ratings and the ratings in the two data sets belong to {1, 2, 3, 4, 5}, in which rating 5 represents liking best and rating 1 indicates liking least. Table 2 characterizes some basic statistical properties of the two data sets.

In the empirical networks, distorted ratings (random/malicious case) are also assumed to exist, similar to the synthetic networks. The artificial spammers in two empirical networks are generated with two kinds of attacks separately: random rating attack and malicious rating

Table 2. Basic statistical properties of the empirical data sets used in this paper. $|U|$, $|O|$ and $|E|$ denote the cardinality of users, objects and ratings, respectively. $\langle k_U \rangle$ and $\langle d_O \rangle$ are the average degree of users and objects, respectively. η records the network sparsity.

Data Sets	$\|U\|$	$\|O\|$	$\|E\|$	$\langle k_U \rangle$	$\langle d_O \rangle$	η
MovieLens	6040	3706	1000209	166	270	0.0447
Netflix	10000	6000	824802	82	137	0.0137

https://doi.org/10.1371/journal.pone.0274567.t002

attack. In both cases, we randomly select some users and assign them distorted ratings: randomly from {1, 2, 3, 4, 5} in the former case; half chance rating 1 and half chance rating 5 in the latter case. The ratio of spammers is q and the activity of spammers is p. For the empirical data sets, we randomly select $q|U|$ users as the spammers, and then assign $p|O|$ distorted ratings to each spammer [24]. If the degree k_i of a selected spammer i is greater than $p|O|$, we randomly distort his/her $p|O|$ ratings, then truncate the remaining $k_i - p|O|$ ratings. Otherwise, we distort all the spammers' ratings, then randomly select $p|O| - k_i$ of his/her non-rated objects and give them distorted ratings.

The AUC curve [32] is again used to characterize the ranking performance of the EigenRank algorithm for the empirical networks. We introduce another index, recall [34]. To calculate the AUC values, we set the spammers and non-spammers as benchmark and non-benchmark, respectively. We conduct n' times independent comparisons of pairs of users: each pair consisting of a spammer and a non-spammer, chosen randomly. Denote n'_1 the number of chances that the spammer has lower reputation than the non-spammer, and n'_2 the number of chances that the two selected users have equal reputation. The AUC value is expressed as

$$AUC_e = \frac{2n'_1 + n'_2}{2n'}, \tag{16}$$

where the parameter $n' = 1 \times 10^9$ in the experiments, as we did in the experiments for synthetic networks. The higher is the AUC value, the more accurate is the ranking of user reputation.

The recall measures to what extent the spammers can be actually identified in the L-lowest reputation users,

$$R_c(L) = \frac{d'(L)}{q|U|}, \tag{17}$$

where $d'(L)$ is the number of identified spammers in the L-lowest reputation users and $L = q|U|$. It's noted that $d'(L) < q|U|$ and $R_c(L) \in [0, 1]$. Higher $R_c(L)$ indicates more accurate reputation ranking list, and vice versa.

Fig 2(a) and 2(d) shows the AUC values AUC_e and recall $R_c(L)$ of the EigenRank algorithm with different (p, q) in the random rating attack case for the MovieLens data set, and Fig 2(b), 2(c), 2(e) and 2(f) shows ΔAUC_e and $\Delta R_c(L)$, the comparison of AUC values and recall between the EigenRank algorithm and IBeta method/Vote Aggregation method. From Fig 2(a) and 2 (d) one sees that AUC_e of the EigenRank algorithm are mainly determined by p, the activity of spammers, while $R_c(L)$ depends mainly on q, the ratio of spammers. Fig 2(b), 2(c), 2(e) and 2 (f) reveals that ΔAUC_e and $\Delta R_c(L)$ are all positive values, implying that the AUC value of the EigenRank algorithm is larger than that of the IBeta method and Vote Aggregation method for each (p, q), and so it is with recall Rc(L). For instance, AUCe = 0.977673, ΔAUCe = 0.05542 (compared to IBeta method) and $R_c(L) = 0.89777$, $\Delta R_c(L) = 0.127881$ (compared to IBeta method) when $(p, q) = (0.1, 0.3)$. The effectiveness of the EigenRank algorithm with different

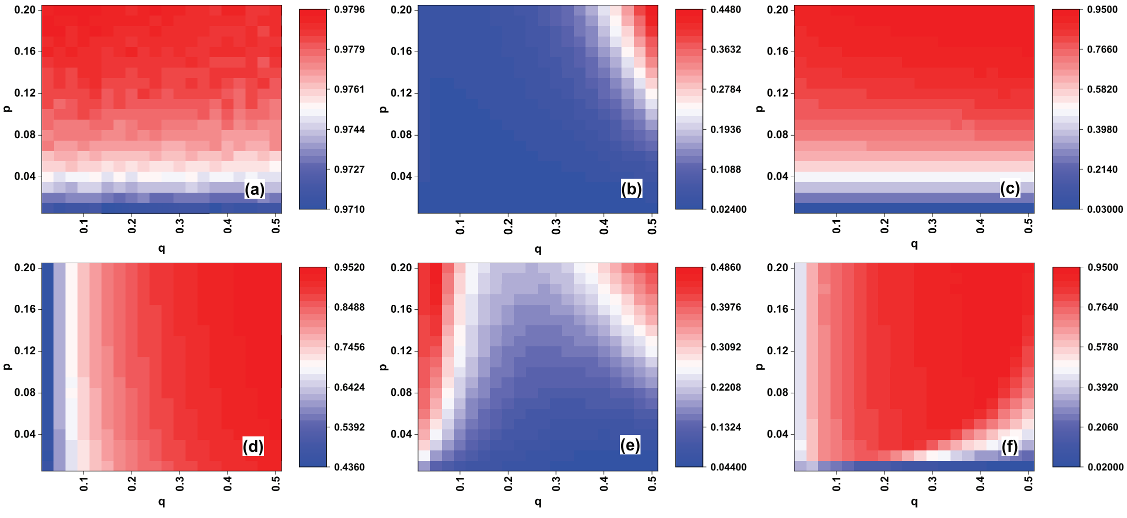

Fig 2. (Color online) The effectiveness of the EigenRank algorithm under random rating attack for the MovieLens data set: (a) the AUC values AUC_e; (b-c) ΔAUC_e, the comparison of AUC_e between EigenRank algorithm and IBeta method (b), Vote Aggregation method (c); (d) the recall $R_c(L)$; (e-f) $\Delta R_c(L)$, the comparison of recall between EigenRank algorithm and IBeta method (e), Vote Aggregation method (f). The parameters q and p denote the ratio and the activity of spammers, respectively. q ranges from 0.025 to 0.5 in increments of 0.025 and p ranges from 0.01 to 0.2 in increments of 0.01. One observes both the AUC_e and $R_c(L)$ of the EigenRank algorithm exceed the ones obtained by the IBeta method and Vote Aggregation method for different (p, q).

https://doi.org/10.1371/journal.pone.0274567.g002

(p, q) in the malicious rating attack case for the MovieLens data set are shown in Fig 3. The results are similar to those in Fig 2.

For the Netflix data set, Figs 4 and 5 show the effectiveness of the EigenRank algorithm in the random and malicious rating attack cases, respectively. One also observes that both the

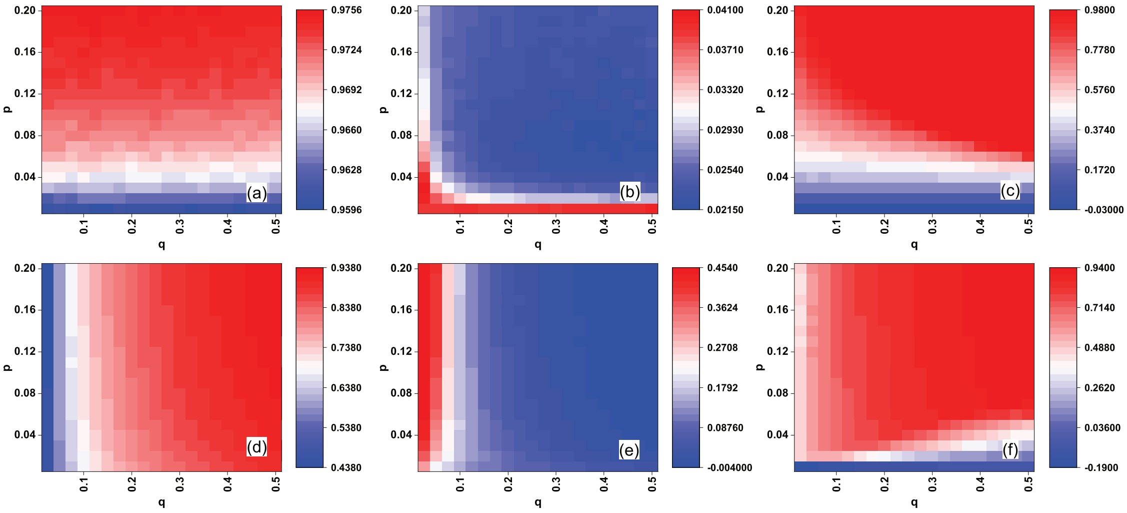

Fig 3. (Color online) The effectiveness of the EigenRank algorithm under malicious rating attack for the MovieLens data set: (a) the AUC values AUC_e; (b-c) ΔAUC_e, the comparison of AUC_e between EigenRank algorithm and IBeta method (b), Vote Aggregation method (c); (d) the recall $R_c(L)$; (e-f) $\Delta R_c(L)$, the comparison of recall between EigenRank algorithm and IBeta method (e), Vote Aggregation method (f). The parameters q and p denote the ratio and the activity of spammers, respectively. q ranges from 0.025 to 0.5 in increments of 0.025 and p ranges from 0.01 to 0.2 in increments of 0.01. One observes that both the AUC_e and $R_c(L)$ of the EigenRank algorithm exceed their counterparts in the IBeta method and Vote Aggregation method for different (p, q).

https://doi.org/10.1371/journal.pone.0274567.g003

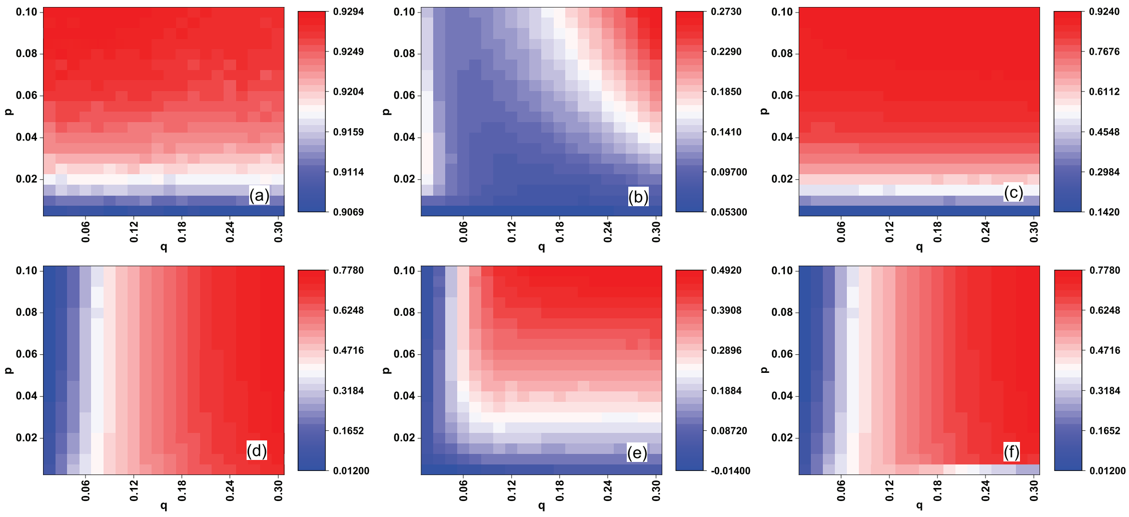

Fig 4. (Color online) The effectiveness of the EigenRank algorithm under random rating attack for the Netflix data set: (a) the AUC values AUC_e; (b-c) ΔAUC_e, the comparison of AUC_e between EigenRank algorithm and IBeta method (b), Vote Aggregation method (c); (d) the recall $R_c(L)$; (e-f) $\Delta R_c(L)$, the comparison of recall between EigenRank algorithm and IBeta method (e), Vote Aggregation method (f). The parameters q and p denote the ratio and the activity of spammers, respectively and $L = q|U|$. q ranges from 0.015 to 0.3 in increments of 0.015 and p ranges from 0.005 to 0.1 in increments of 0.005. One observes that both the AUC_e and $R_c(L)$ of the EigenRank algorithm exceed those of the IBeta method and Vote Aggregation method for different (p, q).

https://doi.org/10.1371/journal.pone.0274567.g004

AUC values and recall of the EigenRank algorithm are larger than their counterparts in the IBeta method and Vote Aggregation method with different pair (p, q) under the random/malicious rating attack case. For example, $AUC_e = 0.908$, $\Delta AUC_e = 0.144$ (compared to Vote Aggregation method) and $R_c(L) = 0.724$, $\Delta R_c(L) = 0.274$ (compared to Vote Aggregation method)

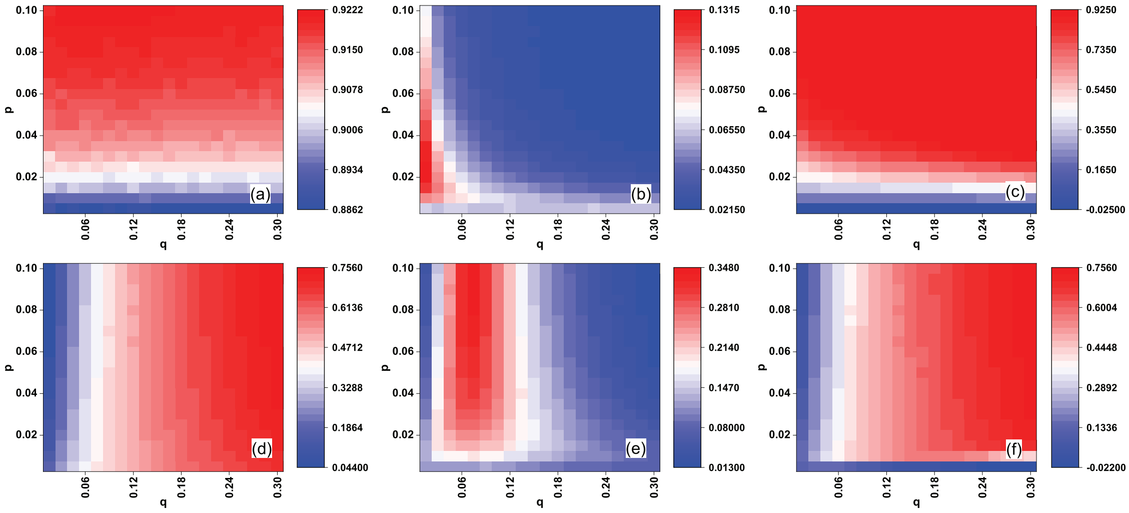

Fig 5. (Color online) The effectiveness of the EigenRank algorithm under malicious rating attack for the Netflix data set: (a) the AUC values AUC_e; (b-c) ΔAUC_e, the comparison of AUC_e between EigenRank algorithm and IBeta method (b), Vote Aggregation method (c); (d) the recall $R_c(L)$; (e-f) $\Delta R_c(L)$, the comparison of recall between EigenRank algorithm and IBeta method (e), Vote Aggregation method (f). The parameters q and p denote the ratio and the activity of spammers, respectively and $L = q|U|$. q ranges from 0.015 to 0.3 in increments of 0.015 and p ranges from 0.005 to 0.1 in increments of 0.005. It can be seen that both the AUC_e and $R_c(L)$ of the EigenRank algorithm are larger than their counterparts in the IBeta method and Vote Aggregation method for different (p, q).

https://doi.org/10.1371/journal.pone.0274567.g005 when $(p, q) = (0.005, 0.3)$ in Fig 4; $AUC_e = 0.915$, $\Delta AUC_e = 0.024$ (compared to IBeta method) and $R_c(L) = 0.658$, $\Delta R_c(L) = 0.081$ (compared to IBeta method) when $(p, q) = (0.05, 0.21)$ in Fig 5; The results for empirical networks (Figs 2–5) indicate that the EigenRank algorithm is of better accuracy and robustness than IBeta method and Vote Aggregation method in the rating attack cases.

Conclusion and discussions

In this paper we present an iterative reputation and quality ranking algorithm via the eigenvector, called the EigenRank algorithm, where the user reputation and object quality are interdependent and the user reputation converges to a unique stationary ranking (the eigenvector corresponding to the largest eigenvalue of a certain matrix). In addition, we prove the convergence of EigenRank algorithm, and analyse its speed of convergence. Moreover, we explore the ranking performance of the EigenRank algorithm for the synthetic and empirical networks compared with the IBeta method and Vote Aggregation method. For the synthetic networks, the results indicate that the AUC values and Kendall's τ of the EigenRank algorithm exceed the ones obtained from the IBeta method and Vote Aggregation method with different proportions of random/malicious ratings. The results for the empirical networks show that the EigenRank algorithm is of better accuracy and robustness than the IBeta method and Vote Aggregation method in the random and malicious rating attack cases. This work provides a ranking without uncertainty for the user reputation identification.

In future, we will examine the convergence of other reputation ranking algorithms. Moreover, the rating time is an important attribute of ratings, and presenting new reputation ranking method considering time factor is worthy of further investigation. Consequently, we will concentrate on searching for more accurate and robust reputation ranking algorithms which are absured to estimate the user reputation.

Supporting information

S1 Appendix.
(PDF)

Acknowledgments

We acknowledge GroupLens Research Group for providing us MovieLens data and the Netflix Inc. for Netflix data.

Author Contributions

Conceptualization: Xiao-Lu Liu.

Data curation: Xiao-Lu Liu.

Formal analysis: Chong Zhao.

Funding acquisition: Xiao-Lu Liu, Chong Zhao.

Investigation: Xiao-Lu Liu.

Methodology: Xiao-Lu Liu, Chong Zhao.

Software: Xiao-Lu Liu.

Validation: Xiao-Lu Liu.

Visualization: Xiao-Lu Liu.

Writing – original draft: Xiao-Lu Liu, Chong Zhao.

Writing – review & editing: Xiao-Lu Liu, Chong Zhao.

References

1. Gao J, Zhang YC, Zhou T. Computational socioeconomics. Phys Rep. 2019; 817: 1–104. Available: https://www.sciencedirect.com/journal/physics-reports/vol/817. https://doi.org/10.1016/j.physrep.2019.05.002
2. Li B, Li RH, King I, Lyu MR, Yu JX. A topic-biased user reputation model in rating systems. Knowl Inf Syst. 2015; 44: 581–607. Available: https://link.springer.com/article/10.1007/s10115-014-0780-9.
3. Liao H, Mariani MS, Medo M, Zhang YC, Zhou MY. Ranking in evolving complex networks. Phys Rep. 2017; 689: 1–54. Available: https://www.sciencedirect.com/science/article/abs/pii/S0370157317300935. https://doi.org/10.1016/j.physrep.2017.05.001
4. Yan SR, Zheng XL, Wang Y, Song WW, and Zhang WY. A graph-based comprehensive reputation model: Exploiting the social context of opinions to enhance trust in social commerce. Inform Sciences. 2015; 318: 51–72. Available: https://www.sciencedirect.com/science/article/pii/S0020025514009487. https://doi.org/10.1016/j.ins.2014.09.036
5. Jøsang A, Ismail R, Boyd C. A survey of trust and reputation systems for online service provision. Decis support syst. 2007; 43(2), 618–644. Available: https://www.sciencedirect.com/science/article/abs/pii/S0167923605000849. https://doi.org/10.1016/j.dss.2005.05.019
6. Zhou T, Kuscsik Z, Liu JG, Medo M, Wakeling JR, Zhang YC. Solving the apparent diversity-accuracy dilemma of recommender systems. Proc Natl Acad Sci USA. 2010; 107(10): 4511–4515. Available: http://www.pnas.org/content/107/10/4511.short. https://doi.org/10.1073/pnas.1000488107 PMID: 20176968
7. Liu JG, Zhou T, Guo Q. Information filtering via biased heat conduction. Phy Rev E. 2011; 84: 037101. Available: http://journals.aps.org/pre/abstract/10.1103/PhysRevE.84.037101. PMID: 22060533
8. Al-Garadi MA, Varathan KD, Ravana SD, Ahmed E, Mujtaba G, Khan MUS, et al. Analysis of online social network connections for identification of influential users: survey and open research issues. ACM Comput Surv. 2018; 51: 16.1–16.37. Available: https://dl.acm.org/doi/abs/10.1145/3155897.
9. Moreno Y, Nekovee M, and Pacheco AF. Dynamics of rumor spreading in complex networks. Phys Rev E. 2004; 69: 066130. Available: https://journals.aps.org/pre/abstract/10.1103/PhysRevE.69.066130. PMID: 15244690
10. Zeng A, Cimini G. Removing spurious interactions in complex networks. Phys Rev E. 2012; 85: 036101. Available: https://journals.aps.org/pre/abstract/10.1103/PhysRevE.85.036101. PMID: 22587143
11. Noorian Z, Zhang J, Liu Y, Marsh S, Fleming M. Trust-oriented buyer strategies for seller reporting and selection in competitive electronic marketplaces. Auton Agent Multi-Ag. 2014; 28: 896–933. Available: https://link.springer.com/article/10.1007/s10458-013-9243-z.
12. Sun JC, Feng L, Xie JR, Ma X, Wang DS, Hu YQ. Revealing the predictability of intrinsic structure in complex networks. Nat Commun. 2020; 11: 574. Available: https://www.nature.com/articles/s41467-020-14418-6. https://doi.org/10.1038/s41467-020-14418-6 PMID: 31996676
13. Sun HL, Liang KP, Liao H, Chen DB. Evaluating user reputation of online rating systems by rating statistical patterns. Knowl-Based Syst. 2021; 219: 106895. Available: https://www.sciencedirect.com/science/article/abs/pii/S0950705121001581. https://doi.org/10.1016/j.knosys.2021.106895
14. Liao H, Wu J, Mao YF, Zhou MY, Vidmer A, Lu KZ. Addressing time bias in bipartite graph ranking for important node identification. Inform Sciences. 2020; 540: 38–50. Available: https://www.sciencedirect.com/science/article/abs/pii/S0020025520305454. https://doi.org/10.1016/j.ins.2020.05.120
15. Brin S, Page L. The anatomy of a large-scale hypertextual Web search engine. Comp Net and ISDN Sys. 1998; 30: 107–117. Available: https://www.sciencedirect.com/science/article/pii/S016975529800110X. https://doi.org/10.1016/S0169-7552(98)00110-X
16. Kleinberg JM. Authoritative sources in a hyperlinked environment. J ACM. 1999; 46: 604–632. Available: https://doi.org/10.1145/324133.324140
17. Laureti P, Moret L, Zhang YC, Yu YK. Information filtering via Iterative Refinement. Europhys Lett. 2006; 75: 1006. Available: https://iopscience.iop.org/article/10.1209/epl/i2006-10204-8/meta. https://doi.org/10.1209/epl/i2006-10204-8
18. Ren J, Zhou T, Zhang YC. Information filtering via self-consistent refinement. Europhys Lett. 2008; 82: 58007. Available: https://iopscience.iop.org/article/10.1209/0295-5075/82/58007/meta. https://doi.org/10.1209/0295-5075/82/58007

19. Zhou YB, Lei T, Zhou T. A robust ranking algorithm to spamming. Europhys Lett. 2011; 94: 48002. Available: http://iopscience.iop.org/0295-5075/94/4/48002. https://doi.org/10.1209/0295-5075/94/48002

20. Liao H, Zeng A, Xiao R, Ren ZM, Chen DB, Zhang YC. Ranking reputation and quality in online rating systems. PLoS ONE. 2014; 9(5): e97146. Available: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0097146. PMID: 24819119

21. Liu XL, Guo Q, Hou L, Cheng C, Liu JG. Ranking online quality and reputation via the user activity. Physica A. 2015; 436: 629–636. Available: http://www.sciencedirect.com/science/article/pii/S0378437115004586. https://doi.org/10.1016/j.physa.2015.05.043

22. Li RH, Yu JX, Huang X, Cheng H. Robust reputation-based ranking on bipartite rating networks. Proceedings of the 2012 SIAM International Conference on Data Mining. 2012; Apr: 612–623. Available: https://epubs.siam.org/doi/abs/10.1137/1.9781611972825.53.

23. Allahbakhsh M, Ignjatovic A. An iterative method for calculating robust rating scores. IEEE Trans Parallel and Distrib Syst. 2015; 26: 340–350. Available: https://ieeexplore.ieee.org/abstract/document/6762988. https://doi.org/10.1109/TPDS.2013.215

24. Gao J, Dong YW, Shang MS, Cai SM, Zhou T. Group-based ranking method for online rating systems with spamming attacks. Europhys Lett. 2015; 110: 28003. Available: http://iopscience.iop.org/0295-5075/110/2/28003. https://doi.org/10.1209/0295-5075/110/28003

25. Gao J, Zhou T. Evaluating user reputation in online rating systems via an iterative group-based ranking method. Physica A. 2017; 473: 546–560. Available: https://www.sciencedirect.com/science/article/pii/S0378437117300559. https://doi.org/10.1016/j.physa.2017.01.055

26. Allahbakhsh M, Ignjatovic A, Motahari-Nezhad HR, Benatallah B. Robust evaluation of products and reviewers in social rating systems. World Wide Web. 2015; 18(1): 73–109. Available: https://link.springer.com/article/10.1007/s11280-013-0242-4.

27. Dai L, Guo Q, Liu XL, Liu JG, Zhang YC. Identifying online user reputation in terms of user preference. Physica A. 2018; 494: 403–409. Available: https://www.sciencedirect.com/science/article/pii/S0378437117312669. https://doi.org/10.1016/j.physa.2017.12.020

28. Liu XL, Liu JG, Yang K, Guo Q, Han JT. Identifying online user reputation of user-object bipartite networks. Physica A. 2017; 467: 508–516. Available: https://www.sciencedirect.com/science/article/pii/S037843711630718X. https://doi.org/10.1016/j.physa.2016.10.031

29. Liu XL, Jia SW. An iterative reputation ranking method via the beta probability distribution. IEEE Access. 2019; 7: 540–547. Available: https://ieeexplore.ieee.org/abstract/document/8567893. https://doi.org/10.1109/ACCESS.2018.2885551

30. He X, Gao M, Kan MY, Wang D. Birank: Towards ranking on bipartite graphs. IEEE T Knowl Data En. 2017; 29(1): 57–71. Available: https://ieeexplore.ieee.org/abstract/document/7572089. https://doi.org/10.1109/TKDE.2016.2611584

31. Barabási AL, Albert R. Emergence of scaling in random networks. Science. 1999; 286(5439): 509–512. Available: https://science.sciencemag.org/content/286/5439/509.abstract. https://doi.org/10.1126/science.286.5439.509 PMID: 10521342

32. Hanley JA, McNeil BJ. The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology. 1982; 143(1): 29–36. Available: https://pubs.rsna.org/doi/abs/10.1148/radiology.143.1.7063747.PMID: 7063747

33. Kendall MG. A new measure of rank correlation. Biometrika. 1938; 30: 81–93. Available: https://sci-hub.se//https://www.jstor.org/stable/2332226. https://doi.org/10.1093/biomet/30.1-2.81

34. Herlocker JL, Konstan JA, Terveen LG, Riedl JT. Evaluating collaborative filtering recommender systems. ACM Trans Inf Syst. 2004; 22(1): 5–53. Available: https://dl.acm.org/doi/abs/10.1145/963770.963772.