J Supercomput (2014) 70:1051–1074
DOI 10.1007/s11227-014-1116-y

PerContRep: a practical reputation system for pervasive content services

Zheng Yan · Yu Chen · Yue Shen

Published online: 5 February 2014

Abstract Social network has extended its popularity from the Internet to mobile domain. Personal mobile devices can be self-organized and communicate with each other for instant social activities at any time and in any places to achieve pervasive social networking (PSN). In such a network, various content information flows. To which extent should mobile users trust it, whilst user privacy can also be preserved? Existing work has not yet seriously considered trust and reputation management, although trust plays an important role in PSN. In this paper, we propose PerContRep, a practical reputation system for pervasive content services that can assist trustworthy content selection and consumption in a pervasive manner. We develop a hybrid trust and reputation management model to evaluate node recommendation trust and content reputation in the context of frequent change of node pseudonyms. Simulations show the advantages of PerContRep in assisting user decisions and its effectiveness with regard to unfair rating attack, collaborative unfair rating attack, on-off attack and conflict behavior attack. A prototype system achieves positive user feedback on its usability and social acceptance.

Z. Yan
The State Key Laboratory of ISN, Xidian University, No. 2 South Taibai Road, Xi'an 710071, China

Z. Yan (✉) · Y. Shen
Department of Communications and Networking, Aalto University, Otakaari 5, 02150 Espoo, Finland
e-mail: zyan@xidian.edu.cn; zheng.yan@aalto.fi

Y. Shen
e-mail: yue.shen@aalto.fi

Y. Chen
Human Computer Interaction Group, EPFL, Lausanne, Switzerland
e-mail: yu.chen@epfl.ch

Keywords Trust · Reputation system · Recommendation · Social networking · Mobile ad hoc networks

1 Introduction

Social network has extended its popularity from the Internet to mobile domain. Personal mobile devices can be self-organized and communicate with each other for instant social activities at any time and in any places to achieve pervasive social networking (PSN). In such a network, various content information flows. For example, a user could query people in vicinity using his/her mobile device about which shop is on sale, which movie is recommended to watch, or which mobile application should be installed for entertainment. The neighbors of the user could respond these queries by providing their recommendations via PSN, e.g., based on mobile ad hoc networks (MANET). This kind of pervasive content services is very valuable for mobile users, especially when fixed networks (e.g., the Internet) or cellular networks are temporarily unavailable or costly to access in present locations of the users (e.g., on a flight) or in urgent situations (e.g., a disaster). Thus, it becomes an essential complement of the Internet on-line social services.

There are quite a number of vivid research activities related to social networking and computing. Recent efforts have started to study social communications in the mobile domain. A number of research groups in academia have focused on social activities based on MANET [1–5]. In industry, several companies, such as Microsoft, Nokia and Intel have conducted researches in the area of PSN [6–9]. However, existing projects have not yet seriously considered trust and reputation management, although trust plays an important role in PSN. It helps people overcome uncertainty, make a correct decision and avoid potential risks. In the literature, trust and reputation mechanisms have been widely studied in various fields of distributed systems, such as MANET, peer-to-peer (P2P) systems, Grid computing, pervasive computing and e-commerce [10]. But the literature still lacks an effective solution that can be practically applied into pervasive content services to solve such a concrete issue as: to which extent should mobile users trust the information provided in the PSN, especially when pseudonyms are applied in information query and response? Moreover, unfair ratings could artificially inflate or deflate reputations in PSN [11]. The reputation systems are vulnerable to a number of potential attacks, such as Sybil attack, on-off attack, independent/collaborative bad mouthing attack, and conflict behavior attack [12,13]. The usage of pseudonyms makes hard to trace malicious behaviors and also badly influences the accuracy of reputation evaluation. Pervasive social networking introduces additional challenges to trust and reputation management with regard to assisting user decisions and tracking malicious social behaviors in practice.

In this paper, we propose PerContRep to solve and overcome the above issues and challenges. It is a reputation system for pervasive content services that can assist users to select trustworthy contents and track malicious social behaviors through trust and reputation management. We design a novel hybrid trust model to evaluate both mobile node recommendation trust (in short node trust) and content reputation (i.e., the public trust in content). It concerns both historical node recommendation behaviors accumu-

lated by a trusted server (TS) who knows the real identifiers of nodes and ephemeral experiences collected by individual nodes using pseudonyms. On the basis of the hybrid trust model, we apply credible content reputation evaluation by weighting the votes of a node on contents with its credibility. The credibility is generated based on the node trust and the similarity of node social interests. Particularly, the node trust is certified by the TS based on historical node recommendation behaviors. It is further evolved according to the quality of recent PSN social activities (e.g., recommending a content by voting) at each individual node. This process is iterative in PerContRep. In particular, PerContRep is designed to provide sound robustness and preserve user privacy. It conducts credible content reputation evaluation to overcome unfair rating attack and collaborative unfair rating attack [11,13]. We further apply suitable decay on the contributions of past experiences and punish malicious recommendations to prevent on-off attack and conflict behavior attack [11]. In addition, PerContRep enhances node privacy through system design. Specifically, the contribution of this paper can be summarized as below:

1. We motivate trust and reputation management for pervasive content services in PSN by proposing an appropriate reputation system architecture that can support node privacy preservation.
2. To the best of our knowledge, PerContRep is one of the first solutions to aid content selection and consumption for mobile users in the context of PSN. It achieves trust and reputation management in a hybrid manner.
3. We prove the effectiveness and robustness of PerContRep through simulations and show its sound usability and social acceptance through a prototype-based user study.

The rest of the paper is organized as follows. Section 2 gives a brief overview of related work. Section 3 introduces the system and thread models of PerContRep and its design goals. The PerContRep system structure, trust model and the algorithms used for content reputation generation and node trust evaluation are described in Sect. 4, followed by experimental simulation results and user study results in Sect. 5. We further discuss the privacy issues and system design considerations in Sect. 6. Finally, conclusions and future work are presented in the last section.

2 Related work

Reputation system architecture is generally classified into two main types: centralized and distributed [14]. The system architecture determines how ratings and reputation scores are communicated between participants in a reputation system. In the literature, distributed trust evaluations have been studied in MANET, seldom the solutions support node privacy [15–17]. This could cause such potential attacks as bad mouthing attack or unfair rating attack targeting at a specific node [13]. Most existing systems maintain a statistical representation of reputation by borrowing tools from the realms of game theory [18–20,47], Bayesian analytics [21] and other theories [44]. These systems try to counter any arbitrary or selfish routing misbehavior of nodes by enforcing nodes to cooperate with each other. However, little work has paid attention to the content reputation issue in PSN with node privacy as a main concern. On the other

hand, practical reputation systems generally apply a centralized server to collect feedback for reputation generation (e.g., eBay [22] and Yahoo auctions [23]). However, many existing systems (e.g., Amazon and eBay) lack considerations on the credibility of user rating. This greatly influences the quality of produced reputations. The usage of pseudonym and the ease of its change additionally complicate the picture by allowing participants to effectively erase their prior history. PerContRep adopts a hybrid reputation system architecture, where node trust and content reputation are evaluated in a distributed way, but with the support of a centralized trusted server.

In the literature, trust and reputation mechanisms have been widely studied in various fields of distributed systems [10]. Many mechanisms have been developed for supporting trusted communications and collaborations among computing nodes [15, 16,25,26,46]. Examples are FuzzyTrust system [27], the eBay user feedback system [22], PeerTrust [19], an objective trust management framework (OTMF) for MANET [26], an application-independent and distributed trust evaluation model for wireless medical sensor networks [46] and Credence—a robust and decentralized system for evaluating the reputation of files in a P2P system [28]. Some work evaluates trust based on social relationships [29]. In these researches, trust can be modeled, calculated and thus expressed using a value. Despite the availability of various trust models and mechanisms, their fundamental criteria are still not well understood. Without sufficiently addressing this issue, the design of trust models is still at an empirical stage [15]. Current work focuses on concrete solutions in specific systems. Additional examination is required before applying an existing solution into another domain. Since none of the above studies considered how to support privacy, it is hard to directly adopt them in PerContRep.

Recently, a number of reputation systems have been proposed in the context of digital contents and ad hoc networks. For example, Adler and Alfaro proposed a content-driven reputation system for Wikipedia authors solely on the basis of content evolution; but not on user-to-user comments or ratings [30]. The concept of data centric trust in volatile environments, such as ad hoc networks, was introduced in [17] to evaluate node trust through the data reported by it. Gupta et al. proposed a partially distributed reputation system for P2P systems by introducing a reputation computation agent (RCA). Its system structure is similar to PerContRep. But this system did not consider the challenges caused by privacy enhancement. In addition, the RCA is applied only for calculating peer reputation based on its contributions to the system. In [31], trust in the evaluator indexes its impact on the rating system. The trust value is dynamically adjusted based on past estimation performance. In PerContRep, we apply the TS to evaluate both node trust and content reputation on the basis of long-term historical recommendation behaviors. The node trust is further evolved based on its content recommendation performance in PSN. Both the node trust and content reputation are, respectively, evaluated by each individual node and at the TS according to ephemeral and historical experiences and knowledge.

Maintaining and disseminating indirect reputation information incur overhead in MANET. In most reputation systems, the reputation of a node should be globally shared in the network. The purpose is to make the reputation of a node known to all other nodes and decrease the detection time. But this causes communication overhead at both the individual node and the network. OCEAN [32] discounts second-hand reputation

exchange and only utilizes local reputation based on direct observations to achieve a reasonable performance. PerContRep concerns both local-aware and global-aware trust and reputation by aggregating local experiences and global experiences together. By deploying the TS, the overhead of reputation maintenance and dissemination is eliminated among PSN nodes.

Inconsistent reputation problem (i.e., different nodes may have different reputation values for the same node) often occurs in the ad hoc networks due to subjective reason and/or different local experiences. This makes it hard to distinguish correct reputation ratings from reputation voting messages. Locally aware reputation system (LARS) was proposed to deal with selfish behaviors and malicious behaviors (e.g., packet dropping and unfair rating) [33]. In LARS, the reputation of a node is derived from direct observation and the exchange of second-hand reputation information is disallowed. In PerContRep, we apply the TS to unify node recommendation trust and content reputation based on local experiences reported by nodes. This trust/reputation information is issued to the node by the TS. Serving as the initial value of trust or an important input to content reputation generation, it is further evolved based on newly accumulated experiences at the individual node. In addition, the above process is iterated. Thereby, we avoid the inconsistent reputation problem and eliminate node trust inaccuracy caused by multi-hop reputation dissemination. Trust or reputation is evaluated based on first-hand experiences no matter at the TS or the node.

Nowadays, reputation systems still face a number of problems. They could be vulnerable to a number of potential attacks, such as unfair ratings that artificially inflate or deflate reputations [11,22,23,34], Sybil attack, on-off attack, independent/collaborative bad mouthing attack, and conflict behavior attack [12,36]. The usage of pseudonyms makes it hard to trace malicious behaviors, thus badly influences the accuracy of trust evaluation and reputation generation. Sun et al proposed a number of schemes to overcome some of the above attacks, but they did not consider the additional challenges caused by privacy preservation [13,37]. ReTrust is an attack-resistant and lightweight trust management scheme for wireless medical sensor networks that can efficiently detect malicious/faulty behaviors such as bad mouthing and on-off attacks [45]. But its two-tier architecture is not suitable for PSN. Jin et al surveyed exiting solutions for overcoming social spam and Sybil attacks in on-line social networks [48]. This survey indicated that privacy is a vital problem suffered in current on-line social networks. PerContRep aims to counter the above potential attacks in the scenario of PSN.

A number of research groups have focused on social activities based on mobile ad hoc networks. Stanford MobiSocial Group has developed Junction, a mobile ad hoc and multiparty platform for MANET applications [1]. Micro-blog [2], developed by SyNRG in Duke University, helps users to post micro-blogs tagged by locations. AdSocial [3], introduced by ETHz Systems Group, provides a pervasive social communication platform. Floating content concept was analyzed based on a theoretical framework to study the fundamental quantities of an ephemeral content sharing service in an opportunistic network [4]. In a proposed floating content system, content is only shared within an anchor zone in a best-effort manner, i.e., copies are kept available within that zone while they are deleted outside the anchor zone [5]. In industry, Microsoft Research Asia developed EZSetup system in order to make a mobile

user find services provided by his/her neighbors [6]. The Nokia Instant Community (NIC) developed by the Nokia Research Center provides an instant social networking platform to allow people in vicinity to communicate, get to know, and share information with each other [7,8]. Similarly, Intel Berkeley Lab ran a project named Familiar Stranger based on mobile devices to extend our feelings and relationships with strangers that we regularly observe but do not interact with in public places [9]. However, trust and reputation aspects in social networking are not considered in these projects. Traditional centralized social networking systems (e.g., facebook) have not taken user privacy into serious concern. They cannot support pervasive social networking demands, especially when users do not have Internet connection, but with location proximity.

In our previous work, we developed PerChatRep, a reputation system for pervasive social chatting based on the result of a need assessment survey [38]. The node trust evaluation is designed by considering the trust influencing factors studied in the user need assessment survey. PerChatRep is specific for a pervasive social chatting scenario, which cannot be applied into pervasive content services since it does not consider how to generate content reputation and the node trust is evaluated based on the factors related to social chatting although the system architecture is similar to PerContRep. In a preliminary version of this paper [35], we studied a hybrid trust model for trustworthy pervasive content services. The serious investigation on PerContRep's effectiveness, robustness and user acceptance, as well as system deployment will be presented in the rest of this paper.

3 Problem statement

3.1 System and threat model

We consider a PSN system involving two different kinds of entities: the PSN nodes that interact with each other for instant social communications; a trusted server (TS) that has functions and capability that the PSN nodes do not have and is trusted to provide identity and key management. It can collect information from the PSN nodes about social behaviors and communications to conduct trust and reputation evaluation. To save computation resources and release processing burdens, PSN nodes may resort to the TS through the mobile Internet to manage identities, keys and maintain trust relationships for secure PSN communications in various situations. To be able to provide integrity and privacy in PSN communications, nodes should be able to authenticate with each other using pseudonyms.

We assume that the TS is reliable and trustworthy for preserving the private data of nodes by deploying secure data protection technologies [39,40]. This assumption is based on existence of business incentives. The TS is assumed to have abundant storage capacity and computation power. It is available for a PSN node to register itself into the system although its availability is not essential during PSN. The communications among PSN nodes and between the nodes and the TS are secured by applying an existing security protocol. Each node registers at the TS with a unique identifier and

the TS can map it with its pseudonyms used in PSN. The nodes may not trust with each other. Delegation is not allowed among PSN nodes since they are mostly strangers.

Obviously, PerContRep could be attacked by a number of attacks. First, malicious nodes can provide dishonest recommendations on contents to frame good ones and/or boost bad ones. This attack, referred to as a bad mouthing or unfair rating attack is the most straightforward attack. More seriously, some nodes could collaboratively perform such an unfair rating attack together. Second, malicious nodes could also behave well and badly alternatively, hoping that they can remain undetected while causing damage. This attack is called as an on-off attack. Third, some nodes could intentionally recommend different contents in a contradictory way, by behaving honestly for one content and dishonestly for another. This is a type of a conflict behavior attack. PerContRep aims at countering the above threats.

PerContRep is designed based on Nokia Instant Community platform over MANET [7], originally named Awareness Networking [8]. This platform had been tested in several trials with hundreds of participants and achieved very good feedback on its connectivity and security [49]. We also used this platform to develop a pervasive chatting reputation system with sound user usage and study results regarding its user acceptance [38]. Thus, in this paper, we assume that the MANET communications are stable and reliable in PerContRep.

3.2 An example application scenario

We consider the following scenario as an example of our target pervasive content services. A PSN node queries its neighbors about a mobile application for photo tagging in a football auditorium by promising to share a valuable picture taken from a good spot. Its neighbors could further distribute this query to other nodes. Suppose that a set of nodes respond the query by recommending some contents by voting, the query node processes all collected responses and calculates the reputation values of recommended contents in order to assist user decision. After consuming the content (i.e., the mobile application), the node could provide feedback to the TS by rating the content and reporting collected recommendations. Thus the TS can evaluate the reputation of involved contents based on all collected data. It can also evaluate all recommender node trust according to recommendation performance by aggregating and analyzing all information about the same node although a PSN node could use multiple pseudonyms. Periodically or by request, the TS issues a new pseudonym with a valid node trust token to each node. The trust token contains the node's latest trust value evaluated by the TS and a time stamp to indicate its validity period. A more interesting scenario is that a node could view the trust values of other nodes in vicinity with the aid of the TS (e.g., via a location-based service). Thus, trust and reputation evaluation and visualization promote honest and secure social networking.

3.3 Design goals

To provide trustworthy pervasive content services in PSN under the aforementioned model, PerContRep design should achieve the following performance goals: (1) Flex-

ibility and comprehension: the trust and reputation management should be flexible no matter if the TS is available or not during pervasive social networking. The system should support trust and reputation evaluation in either a centralized or distributed manner; (2) Security and robustness: PerContRep can effectively counter the attacks as described in Sect. 3.1; (3) Usability: PerContRep provides sound usability and can be easily accepted by mobile users; (4) Privacy preservation: PerContRep allows and supports anonymous identifiers of nodes (i.e., pseudonyms) applied in PSN.

4 The PerContRep

4.1 System structure

We attempt to utilize the advantages of both distributed and centralized trust and reputation management in PerContRep. Figure 1 illustrates PerContRep structure. It is a hybrid trust/reputation management infrastructure. At each node, a network observer records recommendations. A content observer monitors the usage history of the contents to generate real usage behavior-based trust cue. A content voter provides a user interface for the node user to recommend and vote contents. A content query provides the node user an interface to send a content recommendation request (i.e., a query) in PSN. A trust information disseminator reports the recommendation records to the TS. A trust evaluator evaluates the recommendation trust of the query responder nodes and the reputation of recommended contents. A reputation extractor receives the trust tokens issued by the TS and a list of reputation values of different contents. A Trust Dataset stores all data related to the above functional blocks in the node.

At the TS, a reputation generator calculates node trust values and content reputation values; meanwhile it identifies malicious nodes. A reputation distributor distributes the trust tokens, as well as content reputation values to each node periodically or by request. An ID manager handles node registration and issues new node pseudonyms

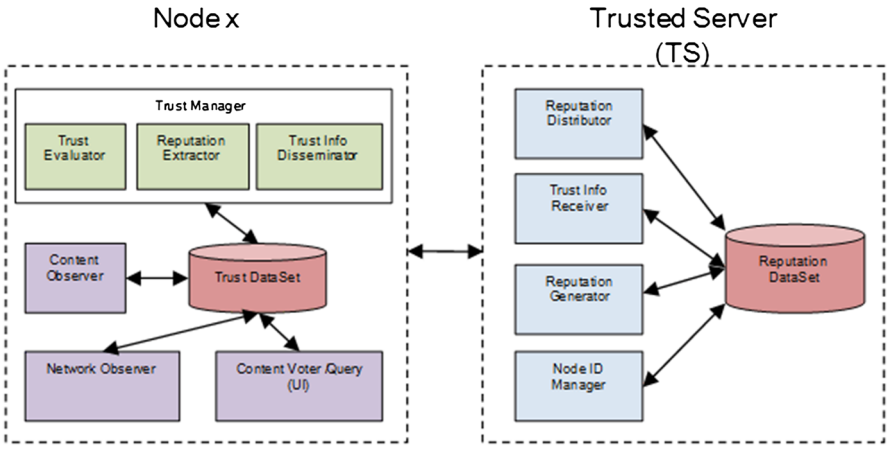

Fig. 1 PerContRep system structure

(by request or periodically). A trust information receiver collects the records reported by the nodes and saves them into a Reputation Database, which also saves the content reputation information, the trust token of each node and the real identifiers of the nodes and their pseudonyms.

Applying the hybrid trust and reputation management and introducing the centralized TS have a number of practical advantages: (a) avoid inconsistent reputation problem that often occurs in MANET due to subjective reasons and/or different local experiences; (b) support accurate node trust evaluation based on unique node identifiers even though the node pseudonyms could be frequently changed, thus overcome the Sybil attack and support user privacy in PSN; (c) save the communication cost among nodes by reducing the communications for trust/reputation information dissemination in MANET; (d) provide an economic approach to collect useful data from PSN that can further support other promising services (e.g., location-based content recommendation services) and new business models. More importantly, PerContRep can generate reputation and evaluate trust in either a distributed or centralized manner. When the TS is not available, it can also work in a distributed way based on the node pseudonyms. On the other hand, we apply the TS to issue the blacklist of malicious nodes and the favorite list of honest and good nodes. It can generate the reputation of various contents based on the feedback and recommendation records reported by the query nodes. A content recommendation service based on content reputation can be provided. With the rapid growth of mobile internet, each mobile device has many chances to connect to the Internet, such as at home or working offices or even public places. The mobile Internet works together with the self-organized ad hoc networks to offer advanced and promising services. Therefore, applying the above system structure is appropriate and has practical significance.

4.2 PerContRep trust model

The way to calculate trust is called the trust model [11]. Based on the above system design, we propose a hybrid PerContRep trust model to generate node trust and content reputation at both the PSN node and the TS, shown in Fig. 2. The query node processes its query response about content recommendations. The trust evaluator calculates the content reputation based on the votes provided by recommender nodes, the popularity of contents and the trust values of recommender nodes. The node trust issued by the TS is further evolved based on recommendation performance at the query node. The evaluated content reputation will be displayed to help the node user make a selection decision. The query node will report its pervasive social activities to the TS later on when the TS can be connected. Thus, the TS can generate a content reputation by aggregating all content recommendations and precisely evaluate the node trust based on its real identifier. The TS can issue the trust token that contains the node trust and update the content reputation periodically or by request. Herein, the trust value of a recommender node partially serves as the credibility of its recommendation. This trust value is generated by considering two sources of information: one is provided by the TS by assessing historical node recommendation behaviors; the other is the query node evaluation on the recommender node based on locally accumulated experiences. The local

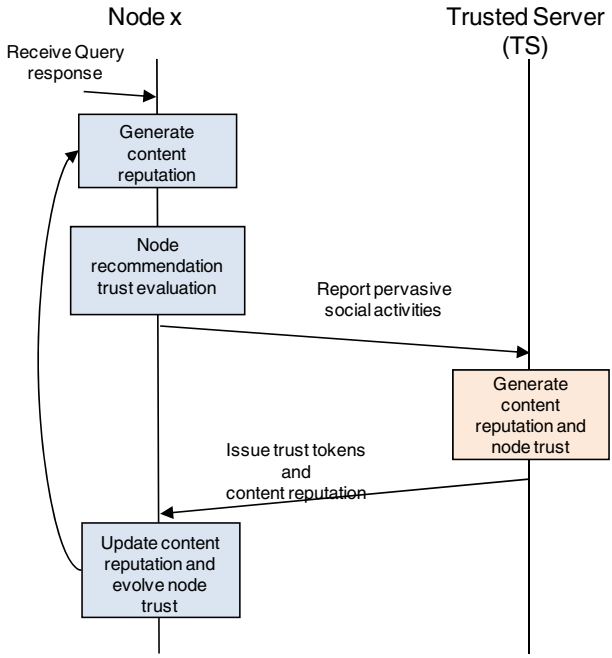

Fig. 2 PerContRep trust model

evaluation is evolved on the basis of the node trust value issued by the TS. Recommendation credibility is applied for generating a reliable reputation value since credibility provides a reason to trust. The evaluation of node trust and content reputation is iterative at both the node and the TS based on newly accumulated experiences and information. The above-described trust model fits into the PerContRep system structure.

To preserve privacy, PerContRep applies the TS to issue and manage the pseudonyms of nodes. The TS can identify the real identifiers of nodes. It could frequently or periodically issue a new pseudonym to the node. Thus, the local experiences are accumulated only based on valid pseudonyms. For example, Node A would assume Node B as a different node once Node B uses a new pseudonym, even though they had interaction with each other before. Historical evaluation on the node trust can only be conducted at the TS by considering all recommendation behaviors related to a concrete node who possibly have applied multiple pseudonyms. Notably, a new trust token is always issued when the pseudonym is changed. However, the node may only request updating the trust value by issuing a new trust token without changing its pseudonym.

4.3 Algorithms

4.3.1 Trust/reputation evaluation at a query node

We consider a number of factors in the calculation of content reputation and node trust at the query node. Based on one of our previous user studies [41], we found

that the more number and/or percentage of the recommendations, the more reputable the content is. This finding was gained from a two-site user study and interviews conducted at both Finland and China with a total of 175 participants to explore the effects of visualizing trust information on mobile users [41]. Thus, we introduce the content popularity that is reflected by the number of votes and its percentage with regard to the total votes in the generation of content reputation. Obviously, the votes of recommender nodes directly show user opinions on a content, thus should be considered in the content reputation generation. But to avoid the bad influence of malicious voting, the credibility of the recommender nodes should be applied to tailor the contribution of each voting. Particularly, the content reputation value issued by the TS (if any) implies content quality and trust assessed based on past public opinions, experiences and knowledge. Thus, it should be considered in the evolution of content reputation even though new information is further accumulated in PSN.

Suppose K nodes respond node q's query and there are a number of K_m nodes ($K_m \leq K$) recommending content C_m. Their votes are $V^{C_m} = \left\{V_1^{C_m}, V_2^{C_m}, \ldots\ldots, V_{K_m}^{C_m}\right\}$. A total of M contents are recommended in the query response. Obviously, the node recommendation trust shows the honesty of node voting. In addition, the interest similarity between the query node and recommender node implies the possibility of recommendation acceptance. Thereby, we calculate the credibility of voting (W_q^k) for node k at node q as below according to interest similarities and k's recommendation trust s_q^k:

$$W_q^k = \text{Cred}\left(\alpha, s_q^k\right) = \frac{\alpha * s_q^k}{Z}, \tag{1}$$

where α is a community factor. The credibility is high if two nodes fall into the same community with similar interests. If q and k are in the same community, $\alpha = \omega_s$, else $\alpha = \omega_d$; and $\omega_s > \omega_d$. s_q^k is the trust value of node q in k, which is upgraded based on the trust token issued by the TS and fine tuned according to the local experiences of node q. Parameter Z is a normalization factor to make $\sum_{k=1}^{K_m} W_q^k = 1$. Note that if the content reputation is available from the TS, we treat it as the vote of the TS and set high credibility for it.

In summary, we calculate the reputation of content C_m as below by considering the popularity of content, the votes and their credibility.

$$T_{C_m} = f\left(K_m\right) \sum_{k=1}^{K_m} W_q^k V_k^{C_m}, \tag{2}$$

where

$$f(K_m) = \left\{1 - \exp\left(\frac{-K_m^2}{2(\sigma + \varepsilon)^2}\right)\right\}. \tag{3}$$

Herein, we apply a revised Rayleigh cumulative distribution function $f(K_m) = \left\{1 - \exp(\frac{-K_m^2}{2(\sigma+\varepsilon)^2})\right\}$ to model the impact of K_m on content reputation T_{C_m} to model

the content popularity. This function can well reflect the impact of an integer (e.g., the number of recommenders), which is exhibited in a scale from 0 to 1. Thus, it is selected to model the influence of content popularity (represented by the number of recommenders) on content reputation. In this function, parameter σ ($\sigma > 0$) is applied to inversely control how fast the number of recommenders could impact T_{C_m}, which increases as K_m increases. The parameter σ can be set from 0 to theoretically ∞ to capture the characteristics of different scenarios, e.g., the size of pervasive content services and the number of PerContRep participants. Parameter ε ($\varepsilon = -K_m/K$) is the percentage of recommendations regarding the total number of system users, which is a factor that implies the preference of the public on content C_m. We apply the parameter ε to further adjust the influence of K_m on the generation of content reputation.

The evaluation on node k's recommendation trust by node q is on the basis of k's trust s^k issued by the TS and evolved according to its voting quality (i.e., recommendation performance). s_q^k is reset to s^k each time when a new trust token is available since the evaluation of the TS on node trust is more accurate. Then, it is further evolved according to its recommendation performance experienced by node q. The evaluation is conducted at each time when the valid responding time of a query is over. We adjust s_q^k based on the deviation between node k's recommendation $V_k^{C_m}$ on content C_m and C_m's current reputation T_{C_m}. s_q^k is decreased if the deviation is bigger than the half of the maximum voting deviation ρ ($\rho = \frac{1}{2}\left\{\max(V_k^{C_m}) - \min(V_k^{C_m})\right\}$); otherwise, s_q^k is increased. $\max(V_k^{C_m})$ is the maximum voting value, whilst $\min(V_k^{C_m})$ the minimum one. Herein, we introduce a warning flag γ to record the number of bad recommendations (i.e., $\rho - \left|T_{C_m} - V_k^{C_m}\right| < \varphi$ ($\varphi = 0$), φ is a parameter to decide a bad vote) to detect an on-off attack or a conflict behavior attack. The initial value of γ is 0. It is increased by 1 each time when a bad vote happens (i.e., $\gamma + +$ if $\rho - \left|T_{C_m} - V_k^{C_m}\right| < \varphi$ ($\varphi = 0$)). We further introduce a parameter thr to be the threshold of possible occurrence of these attacks. It can be ascertained as the maximum number of bad recommendations that can be tolerated by an evaluating party before conducting a punishment on the node trust. The value of *thr* can be intuitively set according to the number of collected recommendations. The more the recommendations, the bigger *thr* can be set in practice. In addition, we apply a parameter μ ($\mu > 0$) to control the deduction of s_q^k (i.e., a punishment) caused by the bad votes. Thus, we have Formula (3) to adjust the node recommendation trust as below:

$$s_q^k = \begin{cases} s_q^k + \delta x \ (\gamma < thr) \\ s_q^k + \delta x - \mu\gamma \ (\gamma \geq thr) \end{cases} = \begin{cases} 1 \ \left(s_q^k > 1\right) \\ 0 \ \left(s_q^k < 0\right) \end{cases}, \ \left(x = \rho - \left|T_{C_m}^{-} V_k^{C_m}\right|\right), \quad (4)$$

where $\delta > 0$ is the margin of s_q^k update. We apply Algorithm 1 to calculate content reputation and node trust at a query node.

Algorithm 1: Trust/Reputation Evaluation at a Query Node q

1. Input:
2. - $\{V^{C_m}\}$ $V^{C_m} = \{V_1^{C_m}, V_2^{C_m}, \ldots\ldots, V_{K_m}^{C_m}\}$: a number of K_m votes on content C_m; $m = (1, \ldots M)$
3. - K: the total number of query responders;
4. - α: a community factor;
5. - M: the number of recommended contents regarding a query;
6. For each recommender node k, do
7. Calculate the credibility of voting W_q^k based on (1);
8. For each recommended content, do
9. Generate content reputation based on (2)(3);
10. Adjust s_q^k for each node k, do
11. For each recommended content by node k, do
12. Update s_q^k based on (4);
13. Output: T_{C_m} $(m = 1, \ldots, M)$; s_q^k $(k = 1, \ldots, K)$.

4.3.2 Trust/reputation evaluation at TS

Similarly, the content reputation is generated at the TS by considering all collected node recommendation information, the trust values of nodes, and the popularity of the content. Suppose a total number of L nodes register into the PerContRep system $N = \{n_1, n_2, \ldots\ldots n_L\}$ and there are M' contents recommended by nodes $C = \{C_1, C_2, \ldots\ldots, C_{M'}\}$. A number of $K_m^{'}$ nodes recommend content C_m with votes $V^{C_m} = \left\{\left\{V_1^{C_m}\right\}, \left\{V_2^{C_m}\right\}, \ldots\ldots, \left\{V_{K_m^{'}}^{C_m}\right\}\right\}$. The total number of content recommenders is K' $(L \geq K' \geq K_m^{'})$. In practice, node k could vote content C_m many times at different time t_i : $\left\{V_k^{C_m}\right\} = \left\{V_k^{C_m(t_i)}\right\}$, $V_k^{C_m(t_i)}$ is node k's vote on C_m at time t_i. We generally pay more attention to the recent votes to cope with the on-off attack. This rule has been applied in many existing works. Symbol $\overline{V_k^{C_m}}$ denotes the aggregated votes of node k on content C_m. It can be calculated in Formula (5).

$$\overline{V_k^{C_m}} = \frac{1}{O} \sum_k V_k^{C_m(t_i)} e^{-\frac{|t-t_i|^2}{\beta}}, \tag{5}$$

where $O = \sum_{n_i} e^{-\frac{|t-t_i|^2}{\beta}}$, t is the content reputation generation time, β is a parameter to control time decaying.

Similar to (1), we generate the voting credibility W^k of node k by considering its recommendation trust s^k. Differently, we also consider local evidence r^k (observed

by the content observer in Fig. 1) as one impact factor of W^k. For example, r^k is generated based on the content usage statistics in order to indicate the preference of node user on the content [24]. In this case, W^k is content dependent (denoted $W^k_{C_m}$ as shown in Line 9 of Algorithm 2). Thereby, we have:

$$W^k = \Psi\left(s^k, r^k\right) = \frac{s^k * r^k}{Z'}, \tag{6}$$

where Z' is a normalization factor to make $\sum_{k=1}^{K} W^k = 1$.

Similar to Formula (2), the reputation value $T^{'}_{C_m}$ of content C_m can be calculated at the TS as below:

$$T^{'}_{C_m} = f(K^{'}_m) \sum_{k=1}^{K^{'}_m} W^k \overline{V^{C_m}_k}, \tag{7}$$

where $f(K^{'}_m) = \left\{1 - \exp\left(\frac{-K^{'2}_m}{2(\sigma'+\varepsilon')^2}\right)\right\}$ $(\varepsilon' = -K^{'}_m/K')$ is also the Rayleigh cumulative distribution function to indicate the popularity of the content. Parameter σ' has the same meaning as σ.

Algorithm 2: Trust/Reputation Evaluation at the Trusted Server (TS)

1. Input:
2. - $C = \{C_1, C_2, \ldots\ldots, C_M\}$, $N = \{n_1, n_2, \ldots\ldots n_L\}$; s^k $(k = 1, \ldots, L)$;
3. - $\{V^{C_m}\}$, $V^{C_m} = \{\{V^{C_m}_1\}, \{V^{C_m}_2\}, \ldots\ldots, \{V^{C_m}_{K^{'}_m}\}\}$: a number of $K^{'}_m$ votes on content C_m;
4. - K': the total number of content recommenders;
5. Organize votes based on real node identifiers n_l $(l = 1, \ldots, L)$;
6. For each node k, do
7. For each recommended content C_m by node k, do
8. Aggregate node k's votes on content C_m based on (5);
9. Calculate W^k (or $W^k_{C_m}$) based on (6);
10. For each content C_m, do
11. Generate content reputation based on (7);
12. Adjust s^k for each node k, do
13. For each recommended content by node k, do
14. Update s^k based on (8);
15. Output: $T^{'}_{C_m}$ $(m = 1, \ldots, M)$; s^k $(k = 1, \ldots, L)$.

At the node registration time, s^k is set as an initial value (e.g., 0.5 in our simulations). Then, it is further evolved based on the recommendation performance of node k on all involved contents. Similar to Formula (4), we design Formula (8) to adjust the node trust value s^k at the TS.

$$s^k = \begin{cases} s^k + \delta y \ (\gamma < thr) \\ s^k + \delta y - \mu\gamma \ (\gamma \geq thr) \end{cases} = \begin{cases} 1 \ (s^k > 1) \\ 0 \ (s^k < 0) \end{cases}, \left(y = \rho - \left| T'_{C_m} - \overline{V_k^{C_m}} \right| \right), \tag{8}$$

where parameters δ, μ, thr, γ, φ, $\min\left(V_k^{C_m}\right)$ and $\max\left(V_k^{C_m}\right)$ have the same meanings as described in Sect. 4.3.1. γ is initially set as 0. It is increased by one each time when a bad recommendation occurs, i.e., if $y < \varphi \ (\varphi = 0)$, $\gamma + +$. Algorithm 2 is applied to calculate content reputation and node trust at the TS.

5 Evaluation

We design a number of simulations to first show the advantages of PerContRep in assisting user decision, and then demonstrate the effects of unfair rating attack, collaborative unfair rating attack, on-off attack and conflict behavior attack on PerContRep. Since PSN supports instant social communications among a limited number of nodes in vicinity, we set a total of $\mathrm{L} = 50$ nodes in our simulations [35]. We randomly select a query node (from good nodes) and a random number of nodes to respond each query. We consider three contents ($M = 3$) with low quality ($T_{C_1} = 0.1$), medium quality ($T_{C_2} = 0.5$) and high quality ($T_{C_3} = 0.9$), respectively. For simplifying the simulations, we assume that ad hoc routing trust can be ensured. The initial node trust value is set to 0.5. Table 1 provides the simulation settings of other system parameters.

5.1 Effects of PerContRep

We tested the performance of Algorithm 1 when the server connection is not available in four scenarios: (1) each query responding node randomly selects one content to vote honestly; (2) each responding node votes all contents honestly; (3) 5 attackers in the network randomly vote one content dishonestly (e.g., by deflating C_3 and inflating C_1 and C_2), whilst other responding nodes randomly vote one content honestly; (4) 5 attackers vote all contents dishonestly, whilst other responding nodes vote all contents honestly. In all of above scenarios, we fix the query node and suppose that all node pseudonyms are not changed. Figure 3 shows this simulation result. We observe that (1) Algorithm 1 performs well when there is no attack in the system, as shown in

Table 1 Simulation settings of system parameters

Symbol	Settings	Symbol	Settings
α	1	δ	0.05
σ	1.01	φ	0
μ	0.1	β	2
r^k	1	σ'	5
thr	5 at node	thr	30 at TS

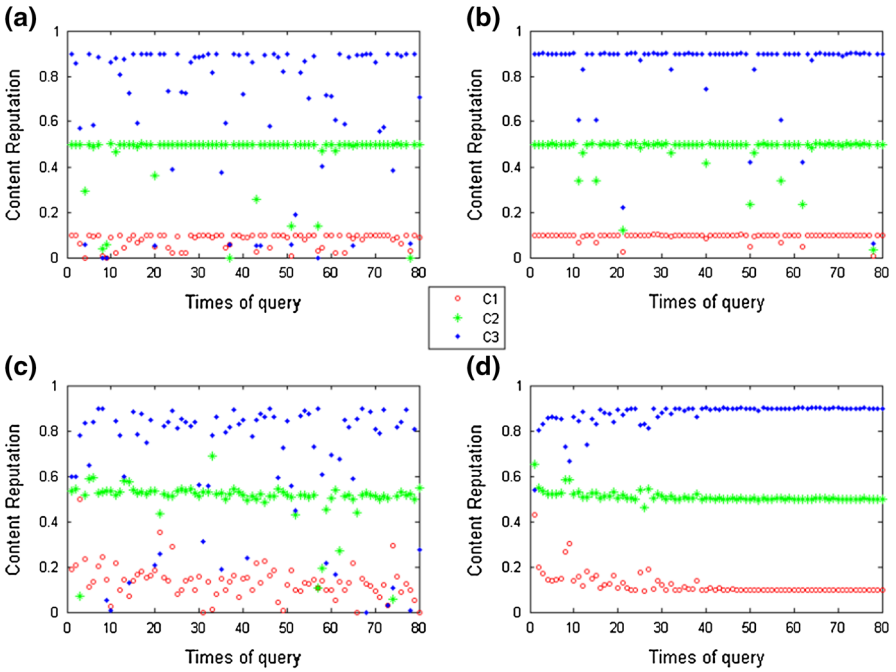

Fig. 3 Content reputations at a query node: **a** each recommender node votes one content honestly; **b** each recommender node votes all contents honestly; **c** 5 malicious nodes vote one content dishonestly; **d** 5 malicious nodes vote all contents dishonestly

Fig. 3a, b; (2) it can adjust the node recommendation trust to improve the content reputation evaluation, as shown in Fig. 3c, d; (3) Algorithm 1 performs better if more recommendation information is collected locally at the query node (comparing Fig. 3a to b, c to d). Note that some reputation values of C_3 in Fig. 3a, c are very low, this is caused by a small number of recommendations since the number of query responders was randomly generated in this simulation.

We further test the performance of both Algorithms 1 and 2 when the server connection is periodically available in the same four scenarios. In this simulation, we randomly select a query node in each query. The query node reports the query results each time when it connects the server after a query. The node pseudonym is changed if it has responded 3 queries. During the change, the server issues a new trust token attached to the new pseudonym. The simulation result is shown in Fig. 4. Four facts are observed. First, PerContRep performs very well at the TS when there is no attack in the system (see Fig. 4 (a) and (b)). Second, PerContRep can adjust the node recommendation trust to identify the malicious node and improve the content reputation evaluation efficiently even though the node pseudonyms could be changed frequently (see Fig. 4c.2, c.3, d.2, d.3). Third, PerContRep performs better if more recommendation information is accumulated by the TS (comparing Fig. 4a to b and c to d). Finally, the TS can evaluate node trust and content reputation more accurately and efficiently than the nodes (comparing Figs. 3 to 4), thus it is significant to introduce the TS in the PerContRep system.

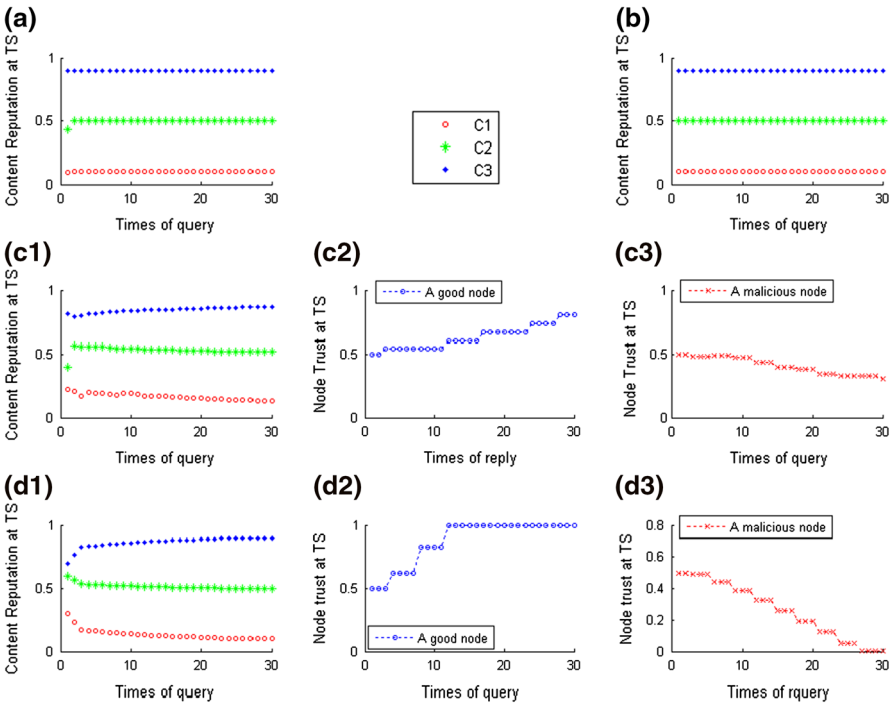

Fig. 4 Content reputations at the TS: **a** each recommender node votes one content honestly; **b** each recommender node votes all contents honestly; **c** 5 malicious nodes vote one content dishonestly; **d** 5 malicious nodes vote all contents dishonestly

5.2 Robustness of PerContRep

We adopt commonly used metrics in information retrieval, Recall (R), Precision (P) and F measure (F) to describe the malicious node detection performance. For each good node k, the number of nodes that belong to Malicious Node (MN) and indeed detected as MN by k, denoted as $x(k)$; the number of nodes that do not belong to MN but are added to MN by k, denoted as $y(k)$; the number of nodes that belong to MN but are not detected as MN by k, denoted as $z(k)$. With these data we do a precision–recall evaluation. We define:

$$R(k) = \frac{x(k)}{x(k) + z(k)} \tag{9}$$

$$P(k) = \frac{x(k)}{x(k) + y(k)} \tag{10}$$

$$F(k) = \frac{2P(k)R(k)}{P(k) + R(k)} \tag{11}$$

The node average F measure of the whole network is:
$F = \sum_{k=1}^{|G|} F(k)/|G|$, where $|G|$ is the number of good nodes in the network.

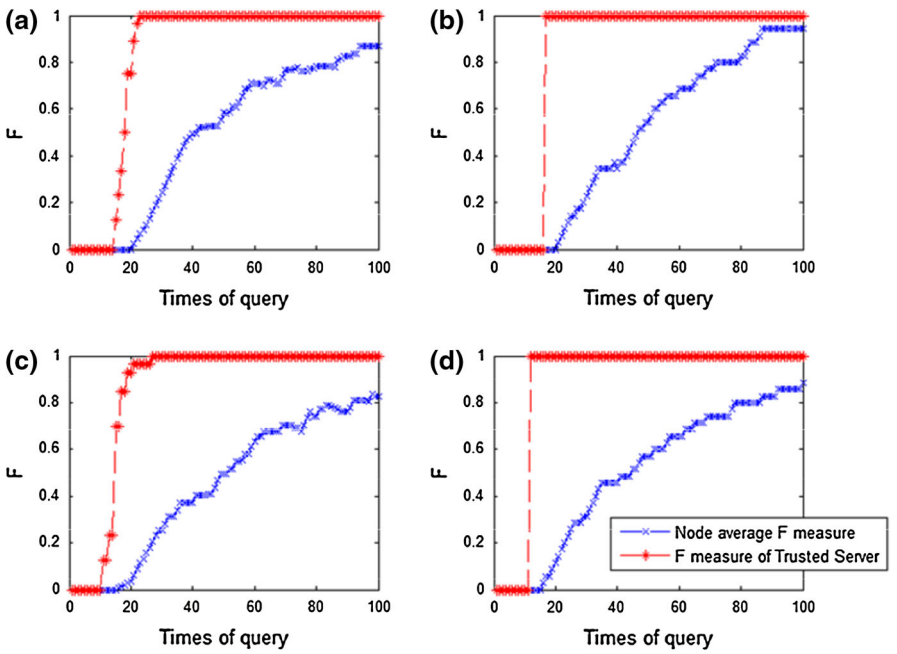

Fig. 5 F measure of nodes and TS regarding malicious node detection performance under 15 (30 % nodes) fixed attackers: **a** unfair rating attack; **b** collaborative unfair rating attack; **c** on-off attack; **d** conflict behavior attack

Obviously, $R, P, F \in [0, 1]$. The higher the R, P, and F are, the more desirable the measures are for good system performance. Particularly, R indicates the performance of false negative detection (i.e., malicious nodes go unnoticed). P indicates the performance of false positive detection (i.e., innocent nodes are blamed). Good system performance requests both high recall R and high precision P. Thus, we make use of F measure to indicate the system performance. Obviously, High F measure is desirable for a good system performance.

We designed simulations to test the effectiveness of PerContRep with regard to unfair rating attack, collaborative unfair rating attack, on-off attack and conflict behavior attack. In the simulations, we did not let the TS issue the blacklist of malicious nodes to each node to compare the performance of trust and reputation generation at the node with the TS.

The impacts of four types of attacks are demonstrated in Fig. 5. We tested four scenarios: (1) unfair rating attack: 15 (30 % nodes) fixed attackers in the network randomly vote one content dishonestly, whilst other responding nodes randomly vote one content honestly; (2) collaborative unfair rating attack: 15 (30 % nodes) fixed attackers in the network vote the same one content (e.g., content C_3, $T_{C_3} = 0.9$) dishonestly, whilst other responding nodes randomly vote one content honestly; (3) on-off attack: 15 (30 % nodes) fixed attackers randomly vote one content with honest and dishonest recommendations alternatively, whilst other responding nodes randomly vote one content honestly; (4) conflict behavior attack: 15 (30 % nodes) fixed attackers

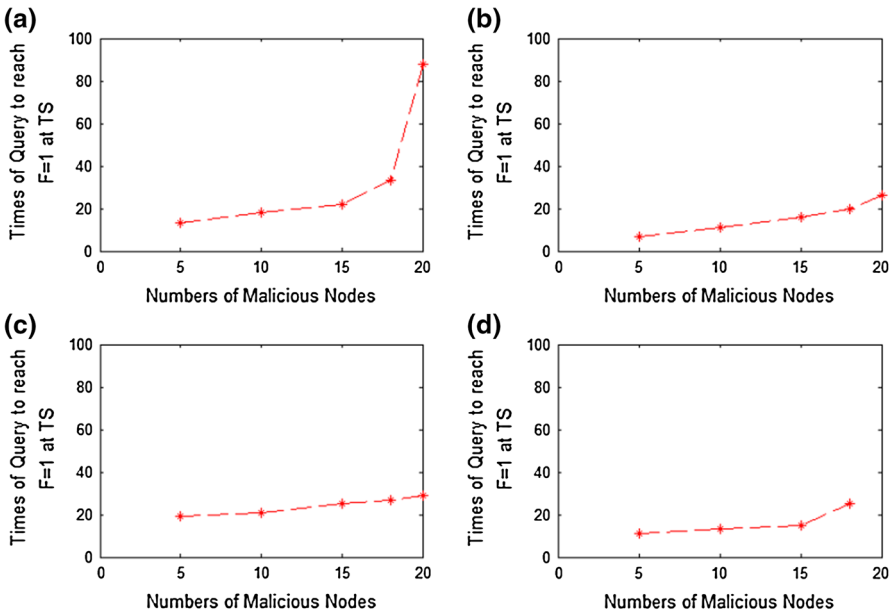

Fig. 6 The robustness of PerContRep in the case of **a** unfair rating attack; **b** collaborative unfair rating attack; **c** on-off attack; **d** conflict behavior attack

vote one content C_1 ($T_{C_1} = 0.1$) honestly and another one C_3 ($T_{C_3} = 0.9$) dishonestly in a consistent way (i.e., they always vote C_1 honestly and C_3 dishonestly), whilst other responding nodes vote these two contents honestly.

We observe that the malicious node detection at the node is much slower than the TS due to the frequent change of node pseudonyms. Thus, introducing the TS can effectively improve the robustness of PerContRep and at the same time enhance the node privacy. Notably, it takes longer time for PerContRep to detect malicious nodes if their percentage is higher. We also find that PerContRep performs more robust and efficient under the collaborative unfair rating attack than the unfair rating attack by comparing Fig. 5b to a. This is because it is easier for PerContRep to detect malicious actions targeting on one content than multiple contents. In addition, PerContRep can achieve good performance with the help of the TS when the on-off attack or the conflict behavior attack happens. Its detection is more accurate and efficient at the TS than the node since the node real identifiers are available and more pieces of evidence are aggregated. Another reason is that the PerContRep algorithm controls the influence of past good or bad behaviors on current content reputation generation.

Figure 6 further shows the robustness of PerContRep under different kinds of attacks. We can see that PerContRep performs well if the number of attackers is less than 40 % of the total number of nodes. It is capable of detecting all malicious nodes under different attacks within a limited number of queries at the TS. If the malicious nodes are below 30 % of the total nodes, the system can detect all of them within 25 times of query. Based on our additional simulation results (e.g., $L = 1{,}000$), mostly 40 % and at least 30 % attack nodes can be tackled. This is

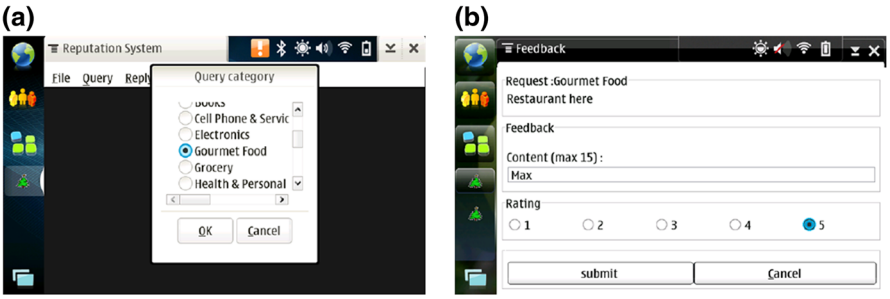

Fig. 7 **a** User interface of node query for content recommendations; **b** user interface of query response for content recommendations

independent of the scale of PerContRep and in accordance with the PerContRep design.

5.3 Implementation of PerContRep

We have implemented a PerContRep prototype system. We developed the PSN nodes with Nokia N810 tablets using Python with GTK binding and the TS on Linux (Ubuntu 9.04) together with Apache, MySql and PHP. Pervasive social networking is implemented via MANET communications based on Wireless LAN chips. In our prototype, there is no guarantee for PSN nodes to connect to the TS. We attempt to achieve efficient power consumption by controlling the message length of node communications within 100 bytes and applying an awareness ad hoc networking platform developed by the Nokia Research Center [8]. This is because the message length of node communications will greatly influence power consumption. The longer the length, the more power will be consumed [8]. The prototype system provides essential security protection on node–server communications with Open SSL and node–node communications by utilizing a community symmetric key. The user interfaces of query and query response are shown in Fig. 7.

5.4 Usability and social acceptance

We further conducted an interview on ten university postgraduates (50 % female) aged between 21 and 27 after feature introduction and PerContRep usage to evaluate its usability and social acceptance in the aspects of perceived usefulness, perceived ease of use, interface, playfulness and user attitude. The participants were asked to express their agreement on the statements listed in Table 2. A 5-point Likert scale was applied. Our interview was designed based on the technology acceptance model (TAM) and its extension, which indicates that usefulness, ease of use and playfulness lead to user acceptance [42,43]. This theory also indicates that good interface leads to perceived usefulness and ease of use; playfulness causes easy acceptance (i.e., the attitude of usage). After the test, each participant was awarded a small gift.

Table 2 PerContRep interview statements

Purpose	Interview statements
Perceived ease of use	Q1: I think it is easy for me to start content information query using PerContRep
	Q2: I think it is easy for me to make a decision during content selection
	Q3: It is easy for me to select contents from a number of recommendations
Perceived usefulness	Q4: PerContRep indicates content reputation
	Q5: PerContRep assists my decision in content selection and consumption
	Q6: PerContRep is a useful and helpful application
Interface	Q7: Reputation visualization during content recommendation is useful
	Q8: PerContRep has a good design on reputation visualization
	Q9: PerContRep has a good design on reputation explanation
	Q10: PerContRep has a good design on user interface
Playfulness	Q11: PerContRep is an interesting application
Attitude	Q12: I like using PerContRep

PerChatRep has satisfactory evaluation scores with regard to perceived ease of use, perceived usefulness, interface, playfulness and user attitude. Their average rating scales were 4.23, 4.33, 3.95, 4.40 and 4.00, respectively. We got fairly high average scores (>4.2) for perceived ease of use, perceived usefulness, and playfulness. In terms of perceived ease of use, we notify that visualizing reputation in pervasive content services made participants easier to select contents from a number of recommendations. The result showed that PerContRep is a very useful and interesting (playful) application with good usability that can aid user decision in pervasive content services. Its UI (e.g., content recommendation query and response) gained good feedback from the participants. They liked using PerContRep. Based on the TAM, we can conclude that PerContRep was well accepted by the participants.

6 Further discussion

In PerContRep, the pseudonyms of node can be generated in different ways. One simple scenario is that the pseudonym is the node ID expressed by its MAC address and/or network layer address that could be frequently changed to enhance node privacy. In this scenario, the system supports that the node pseudonym is selected from a pre-prepared list stored at both the TS and each node during node registration. Thus, the TS can link the pseudonym to the unique identifier of a node for the purpose of node trust evaluation and content reputation generation. The server knows 'who is who' after getting the reports from the nodes, even though the node could change its pseudonym without contacting the server. Another scenario is that the generation of the pseudonym is based on cryptography algorithms, e.g., shared key or public/private key mechanisms. In the case of a shared key, each client has a unique shared key known by the server. The client generates the pseudonym as the hash of the shared key and a random number. When the client sends the pseudonym and the random number to the server, the server can identify the client by calculating the hash with the shared key and the random number and comparing with the received hash. Moreover, advanced public/private key authentication algorithms can be applied in the generation

of pseudonym as well. However, the selection of a suitable mechanism depends on a number of factors such as security, complexity, computation cost, power consumption, efficiency, and so on. The discussion of the algorithms is out of the scope of the paper.

In PerContRep, we evaluate trust and reputation at both individual nodes and the TS by applying two similar but different algorithms. Although node evaluation is efficient, it is ephemeral with limited information due to the change of pseudonyms. Thereby, we adopt the TS to complement this disadvantage by aggregating all related information together for a more accurate and reliable evaluation. The advantage of our design is proved by our simulations. With this way, we overcome the challenges caused by privacy enhancement. Each node can provide its valid trust token issued by the TS to other nodes in PSN if needed. Regarding when the trust token should be issued, we apply the following policies in our prototype system:

- The trust token should be issued to a node when its pseudonym is changed;
- A new trust token should be issued to a node when its old token is expired;
- The server should notify all good nodes by issuing a new trust token of a malicious node once it is detected;
- The trust token is updated by the server upon a node request or periodically or every time the node gains connection to the server.

7 Conclusions

In this paper, we designed and developed PerContRep based on the specified design goals. It can assist user for content selection and consumption in pervasive content services through trust and reputation evaluation on both PSN nodes and contents recommended over PSN. Based on the hybrid trust and reputation management model, PerContRep can flexibly support trust and reputation evaluation in either a centralized or distributed manner no matter if the TS is available or not during pervasive social networking. Simulation based evaluation showed the effectiveness of PerContRep and its robustness with regard to unfair rating attack, collaborative unfair rating attack, on-off attack and conflict behavior attack when the malicious nodes do not exceed 30 % of the total number of system nodes. A PerContRep prototype system based on Nokia N810 tablets achieved good feedback on its usability and social acceptance based on a small scale user study according to the TAM model. In general, the participants liked using PerContRep. Almost all participants reported PerContRep's ease of use, usefulness and playfulness. We further discussed practical solutions for identity privacy preservation and trust token issuing in PerContRep. Current results showed that PerContRep well achieved its design goals with regard to flexibility and comprehension, security and robustness, usability, and privacy preservation. For future work, we are going to further improve and extend PerContRep and apply it to support new business scenarios.

Acknowledgments This work is sponsored by the following grants: the PhD grant (JY0300130104) of Chinese Educational Ministry, the initial grant of Chinese Educational Ministry for researchers from abroad (JY0600132901), and the grant of Shaanxi Province for excellent researchers from abroad (680F1303). A preliminary version of this paper appeared in the Proceedings of the 7th International Conference Ubiquitous Intelligence and Computing 2010 (UIC '10) [35].

References

1. Junction, Stanford MobiSocial Group. http://openjunction.org/. Accessed 20 March 2013
2. MicroBlog. http://synrg.ee.duke.edu/microblog.html. Accessed 20 March 2013
3. Sarigöl E, Riva O, Stuedi P, Alonso G (2009) Enabling social networking in ad hoc networks of mobile phones. Proc VLDB Endow 9(2):1634–1637
4. Hyytia E, Virtamo J, Lassila P, Kangasharju J, Ott J (2011) When does content float? characterizing availability of anchored information in opportunistic content sharing. In: Proceedings of IEEE INFOCOM, pp 3137–3145
5. Ott J, Hyytiä E, Lassila PE, Kangasharju J, Santra S (2011) Floating content for probabilistic information sharing. Pervasive Mob Comput 7(6):671–689
6. EZSetup. http://research.microsoft.com/en-us/groups/wn/mssn.aspx. Accessed 20 March 2013
7. Nokia Instant Community. http://conversations.nokia.com/2010/05/25/nokia-instant-community-gets-you-social/. Accessed 20 March 2013
8. Ahtiainen A, Kalliojarvi K, Kasslin M, Leppanen K, Richter A, Ruuska P, Wijting C (2009) Awareness networking in wireless environments: means of exchanging information. IEEE Vehicular Technol Mag 4(3):48–54
9. Familiar Stranger. http://www.paulos.net/research/intel/familiarstranger/index.htm. Accessed 20 March 2013
10. Yan Z, Holtmanns S (2008) Trust modeling and management: from social trust to digital trust. In: Subramanian R (ed) Computer security, privacy and politics: current issues challenges and solutions. Idea Group Inc., Hershey, USA, pp 290–323
11. Yang Y, Sun Y, Kay S, Yang Q (2009) Defending online reputation systems against collaborative unfair raters through signal modeling and trust. Proc ACM SAC 2009:1308–1315
12. Douceur JR (2002) The sybil attack. In: Proceedings of IPTPS, LNCS, vol 2429, pp 251–260
13. Sun Y, Han Z, Liu KJR (2008) Defense of trust management vulnerabilities in distributed networks. IEEE Commun Mag 46(2):112–119
14. Jøsang A, Ismail R, Boyd C (2007) A survey of trust and reputation systems for online service provision. Decision support systems. Elsevier, Netherlands, pp 618–644
15. Sun Y, Yu W, Han Z, Liu KJR (2006) Information theoretic framework of trust modeling and evaluation for ad hoc networks. IEEE J Sel Area Commun 24(2):305–317
16. Theodorakopoulos G, Baras JS (2006) On trust models and trust evaluation metrics for ad hoc networks. IEEE J Sel Areas Commun 24(2):318–328
17. Raya M, Papadimitratos P, Gligory VD, Hubaux JP (2008) On data-centric trust establishment in ephemeral ad hoc networks. IEEE INFOCOM, pp 1912–1920
18. Michiardi P, Molva R (2002) Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. Adv Commun Multimed Secur LNCS 2828:107–121
19. Xiong L, Liu L (2003) A reputation-based trust model for peer-to-peer ecommerce communities. In: Proceedings of the IEEE conference on E-Commerce, pp 228–229
20. Buchegger S, Boudec JL (2002) Performance analysis of the CONFIDANT protocol. In: Proeedings of the ACM international symposium on mobile ad hoc networking and computing (MobiHoc), pp 226–236
21. Buchegger S, Boudec JYL (2003) The effect of rumor spreading in reputation systems for mobile ad-hoc networks. In: Proceedings of WiOpt modeling and optimization in mobile, ad hoc and wireless networks, Sophia-Antipolis
22. Resnick P, Zeckhauser R (2002) Trust among strangers in internet transactions: empirical analysis of eBay's reputation system. In: Baye M (ed) Advances in applied microeconomics: the economics of the internet and e-commerce. Elsevier, Netherlands
23. Resnick P, Kuwabara K, Zeckhauser R, Friedman E (2000) Reputation systems. Commun ACM 43(12):45–48
24. Yan Z, Zhang P, Deng RH (2012) TruBeRepec: a trust-behavior-based reputation and recommender system for mobile applications. J Pers Ubiquitous Comput 16(5):485–506
25. Lin C, Varadharajan V, Wang Y, Pruthi V (2004) Enhancing grid security with trust management. In: Proceedings of IEEE international conference on services, computing, pp 303–310
26. Li J, Li R, Kato J (2008) Future trust management framework for mobile ad hoc networks. IEEE Commun Mag 46(4):108–115

27. Song S, Hwang K, Zhou R, Kwok YK (2005) Trusted P2P transactions with fuzzy reputation aggregation. IEEE Internet Comput 9(6):24–34
28. Walsh K, Sirer EG (2005) Fighting peer-to-peer spam and decoys with object reputation. In: Proceedings of P2PECON, pp 138–143
29. Trifunovic S, Legendre F, Anastasiades C (2010) Social trust in opportunistic networks. In: Proceedings of IEEE INFOCOM Workshops, pp 1–6
30. Adler BT, Alfaro L (2007) A content driven reputation system for the Wikipedia. WWW07, pp 261–270
31. Kujimura K, Nishihara T (2003) Reputation rating system based on past behavior of evaluators. In: Proceedings of the 4th ACM conference on electronic commerce, pp 246–247
32. Bansal S, Baker M (2003) Observation-based cooperation enforcement in ad hoc networks. Technical Report, Stanford University, NI/0307012
33. Hu J, Burmester M (2006) LARS: a locally aware reputation system for mobile ad hoc networks. In: Proceedings of the 44th ACM annual Southeast Regional Conference, pp 119–123
34. Corritore CL, Kracher B, Wiedenbeck S (2003) On-line trust: concepts, evolving themes: a model. Int J Hum Comput Stud 58(6):737–758
35. Yan Z, Chen Y (2010) AdContRep: a privacy enhanced reputation system for MANET content services. LNCS 6407:414–429
36. Liu Z, Yau SS, Peng D, Yin Y (2008) A flexible trust model for distributed service infrastructures. In: Proceedings of 11th IEEE symposium on object oriented real-time, distributed computing, pp 108–115
37. Sun Y, Han Z, Yu W, Liu KJR (2006) A trust evaluation framework in distributed networks: vulnerability analysis and defense against attacks. INFOCOM, pp 1–13
38. Yan Z, Chen Y, Shen Y (2012) A practical reputation system for pervasive social chatting. J Comput Syst Sci. doi:10.1016/j.jcss.2012.11.003
39. Yu S, Wang C, Ren K, Lou W (2010) Achieving secure, scalable, and fine-grained data access control in cloud computing. In: Proceedings of IEEE INFOCOM, pp 534–542
40. Wan Z, Liu J, Deng RH (2012) HASBE: a hierarchical attribute-based solution for flexible and scalable access control in cloud computing. IEEE Trans Info Forensics Secur 7(2):743–754
41. Yan Z, Liu C, Niemi V, Yu G (2013) Exploring the impact of trust information visualization on mobile application usage. J Pers Ubiquitous Comput. doi:10.1007/s00779-013-0636-4
42. Davis FD (1989) Perceived usefulness, perceived ease of use, and user acceptance of information technology. MIS Q 13(3):319–340
43. Venkatesh V, Bala H (2008) Technology acceptance model 3 and a research agenda on interventions. Decis Sci 39(2):273–315
44. Yan Z (2010) Trust modeling and management in digital environments: from social concept to system development, IGI Global
45. He D, Chen C, Chan S, Bu J, Vasilakos AV (2012) ReTrust: attack-resistant and lightweight trust management for medical sensor networks. IEEE Trans Info Technol Biomed 16(4):623–632
46. He D, Chen C, Chan S, Bu J, Vasilakos AV (2012) A distributed trust evaluation model and its application scenarios for medical sensor networks. IEEE Trans Info Technol Biomed 16(6):1164–1175
47. Wang Y, Nakao A, Vasilakos AV, Ma J (2011) P2P soft security: on evolutionary dynamics of P2P incentive mechanism. Comput Commun 34(3):241–249
48. Jin L, Chen Y, Wang T, Hui P, Vasilakos, AV (2013) Understanding user behavior in online social networks: a survey. IEEE Commun Mag
49. Yan Z, Niemi V, Chen Y, Zhang P, Kantola R (2013) Mobile social networking: an innovative approach. In: Chin A, Zhang D (eds) Towards trustworthy mobile social networking. Springer, Germany, pp 195–235