Egyptian Informatics Journal 24 (2023) 257–263

Contents lists available at ScienceDirect

Egyptian Informatics Journal

journal homepage: www.sciencedirect.com

Study on trust evaluation and service selection for Service-Oriented E-Commerce systems in IoT environments

Xu Wu [*,a,b], Junbin Liang [a]

[a] School of Computer, Electronics and Information, Guangxi University, University East Road, Nanning 530000, China
[b] School of information science and technology, Hainan Normal University, Longkun South Road, Haikou 571158, China

ARTICLE INFO

Article history:
Received 3 April 2022
Revised 10 March 2023
Accepted 19 March 2023
Available online 1 April 2023

Keywords:
Trust computing
Service selection
Internet of Things
E-Commerce systems

ABSTRACT

Trust-based service management in IoT systems, a promising research direction provides an intelligent solution for the identification of appropriate and trustworthy service providers from a huge of number smart nodes providing similar services. An efficient trust-based service ranking scheme (TBSRS) is proposed. The novelty of our design lies in providing accurate trust evaluation and service ranking in the absence of assessment data or presence of untrustworthiness recommendations. TBSRS also addresses the issue about the trustworthiness evaluation of a new service. Service providers are ranking based on the binary fruit fly optimization algorithm in TBSRS. The experimental results demonstrate desirable convergence, accuracy, and resiliency properties of TBSRS, and also demonstrate that TBSRS outperforms Adaptive IoT Trust and IoT-HiTrust in terms of the precision of service selection and other a few performance metrics.

1. Introduction

As there are many E-Commerce Systems which can provide the similar services, it is hard for the nodes to identify the appropriate and trustworthy service providers from these similar service alternatives. Applying a trust model to the study of service selection has been proved to be an effective approach. Trustworthiness of nodes is considered as a service quality metric which reflects the performance characteristics of the services. The approach helps a service requester to decide whether the service provided is trustworthy or not.

Most of trust based service selection approaches assess the trustworthiness of service providers based on the direct service experiences or indirect recommendations. In general, if a node has not enough interaction experiences with service providers, then recommendations would be needed for decision making. These recommendations may include a number of untrustworthy or unfair users' feedbacks. Various recommendation filtering methods have been proposed in the area. One method is to use trust threshold to filter untrustworthy recommendations [1–3]. However, it neglects the most critical service requirements of requester. This greatly impacts on the results of recommendation. Another method is to compute social similarity to filter untrustworthy recommendations [4–7]. But it is impractical to build an accurate social network graph for all nodes in an IoT network where the number of nodes is too large, and many nodes often join in or leave the network due to their mobility.

How to evaluate the trustworthiness of a new services in Service-Oriented E-Commerce Systems. For a new service, it is difficult to evaluate the trustworthiness due to data sparsity. To the best of our knowledge, one possible solution used by most of trust models is to assign an initial trust value to the new service. However, this leads to another problem how to determine the initial trust value. In this paper, we propose an efficient trust-based service ranking scheme given as TBSRS which can provide accurate trust evaluation and service ranking in the absence of assessment data or presence of untrustworthiness recommendations. The key contributions of this paper are as follows:

1) A service requirement similarity algorithm based on distance correlation coefficient was proposed for filtering out the recommenders who have a low similarity with the service requester; 2) A novel trust evaluation algorithm is proposed based on combining subjective and objective data. The algorithm solves trustworthiness assessment problem of a new service; 3) A service ranking method is proposed based on the binary fruit fly optimization algorithm, which provides the optimal ranking for service providers by

[*] Corresponding author.
E-mail address: xrdz2006@163.com (X. Wu).

https://doi.org/10.1016/j.eij.2023.03.003
1110-8665/© 2023 THE AUTHORS. Published by Elsevier BV on behalf of Faculty of Computers and Artificial Intelligence, Cairo University.

considering both service provider's trustworthiness and service requester's benefit.

The remainder of the paper is arranged as follows: Section II discusses the previous work. The proposed scheme TBSRS is presented in Section III. In Section IV, we conduct extensive experiments to evaluate the proposed scheme. Conclusions and our future work are discussed in Section V.

2. Related work

With the development of IoT technology, more and more E-Commerce applications are running on top of it, such as smart health, smart car, smart home and smart city [9,10]. A large of number smart nodes can request and provide these similar applications on behalf of the owners. In such a scenario, it is crucial to select the appropriate and trustworthy service providers for maximally satisfying the service requester. Trust is viewed as an effective method for selecting trustworthy service providers [11,12,18]. A noteworthy point is that the traditional trust models for P2P networks, social networks and service computing system cannot be directly applied for service selection in IoT systems due a huge number of IoT nodes, varied user requirements and complex context environments [13]. Designing an effective and efficient trust computation method is still an urgent and important issue in Service-Oriented E-Commerce Systems.

Ing-Ray Chen et al. [8] propose a distributed IoT trust management protocol given as Adaptive IoT Trust. In order to solve the recommendation collecting issue in Adaptive IoT Trust, Ing-Ray Chen et al. propose a 3-tier cloud-cloudlet device hierarchical trust-based service management protocol called IoT-HiTrust [5], where recommendations are acquired in the cloud. Although IoT-HiTrust is a scalable solution, the trustworthiness computation of service providers still must rely on the existence of enough friendship social relationship data in nature. Yating Wang et al. [1] propose a context-aware trust management model for service-oriented IoT networks. This paper pays attention to predict the probability of making satisfactory service instead of service quality. Al-Hamadi Hamid et al. [6] discuss the trust management problem in a smart service community. The trustworthiness of service providers is assessed based on service requester's own experiences and other witnesses' experiences. Zhiting Lin et al. [2] propose a comprehensive model of trust for social IoT. Relative to [1,4-6,8], the work in [2] focuses on using objective assessment data to evaluate the trustworthiness of service providers. Li et al. [14] propose a trust model to identify trusted service providers in a dynamic IoT environment. The trust of service providers was evaluated based on the objective historical information relating to a service's previous negotiations and its monitored run-time performance. Wang et al. propose a method of Integrating modified cuckoo algorithm and creditability evaluation for QoS-aware service composition [15]. In an open Internet environment, services attribute values will dynamically change along with the different context environments, so the service selection methods based on service attribute values are still have limitations. Binsi et al. [21] propose a trust inference model for service recommendation in IoT. The model incorporated the direct and indirect subjective data for rating predication. Our work is inspired by the above existing research works, different from these methods, our work focuses on providing accurate trust evaluation and service ranking in the absence of assessment data or presence of untrustworthiness recommendations.

3. System model

The section provides the details about the proposed scheme. TBSRS is proposed for Service-Oriented E-Commerce Systems (SOEs) consisting of service providers (SPs) and service requesters (SRs). In this SOE application, TBSRS is based on an assumed scenario, where a SR (service requester) requests a service, and multiple SPs (service providers) with similar service quality are competing for the requested service. The system framework is shown in Fig. 1. The entire framework of TBSRS comprises of three components, namely (i) Filtering component, (ii) Trustworthiness assessment component, and (iii) Ranking component.

A. Filtering component

The filtering progress includes two steps:

Step one: Filtering based on service requirement similarity.

Service requirement is a subject description of user own idea about the most critical metrics of service performance. All features of requirement are be normalized to the interval [0,1]. Suppose $S = \{s_i | i \in \{1, \ldots, m\}\}$ is the set of m services. $N = \{n_j | j \in \{1, \ldots, p\}\}$ is the set of p nodes. $F = \left\{F^k | k \in \{1, \ldots, l\}\right\}$ is the set of l features of service requirement. $F^k_{i,j}$ represents the kth feature value of n_j's service requirement with service s_i, where $i \in \{1, \ldots, m\} j \in \{1, \ldots, p\}$ and $k \in \{1, \ldots, l\}$. According to different features ("benefit" or "cost"), when $F^k_{i,j} \neq$, The normalized feature value $F^k_{i,j}$ is given by:

$$F^k_{i,j} = \begin{cases} \frac{F^k_{i,j} - F^k_{\min}}{F^k_{\max} - F^k_{\min}}, F^k_{i,j} \, is \, benefit \\ \frac{F^k_{\max} - F^k_{i,j}}{F^k_{\max} - F^k_{\min}}, F^k_{i,j} \, is \, \cos t \end{cases} \tag{1}$$

where $F^k_{\min}$ and $F^k_{\max}$ are the minimum and maximum features values, respectively. To support service selection, multiple features are composed on the basis of their weights $\mathrm{F_{i,j}}$ is the composite feature value, and it is defined as:

$$F_{i,j} = \sum_{k=1}^{l} w_k \times F^k_{i,j} \tag{2}$$

where (w_k $w_k \in [0, 1]$ $\sum_{k=1}^{l} w_k = 1$)is the weight of the kth feature.

Pearson Correlation Coefficient (PPC) [16], is used to measure the user similarities. Given the value $F_{i,j}$ and $F_{i,c}$, the similarity $F_{sim(j,c)}$ between n_j and n_c is obtained based on PPC formula:

$$F_{sim(j,c)} = \frac{\sum_{s_i \in S_{j,c}} \left(F_{i,j} - \bar{F_j}\right)\left(F_{i,c} - \bar{F_c}\right)}{\sqrt{\sum\limits_{s_i \in S_{j,c}} \left(F_{i,j} - \bar{F_j}\right)^2} \sqrt{\sum\limits_{s_i \in S_{j,c}} \left(F_{i,c} - \bar{F_c}\right)^2}} \tag{3}$$

where $\bar{F_j}$ and $\bar{F_c}$ are average feature values of n_j and n_c respectively. In this paper, a dynamic threshold δ which differs with respect to each service requester is used. Only when the similarity $F_{sim(j,c)}$ is bigger than the threshold δ, the recommendation from n_c can enter the step two.

Step two: Filtering based on the trustworthiness of recommender.

To filter out untrustworthiness recommendations from step one, a threshold parameter T^{th} is used. Only when n_c's trustworthiness is bigger than the threshold parameter T^{th}, its recommendations will be accepted.

B. Trustworthiness assessment component

The details of trustworthiness assessment are expressed in the following:

1) Subjective evaluation

The subjective trustworthiness of SPs reflects SR's subjective confidence level of service.

It is assumed that a service s_i provided by n_q has n indirect recommendations from filtering component, which is denoted as

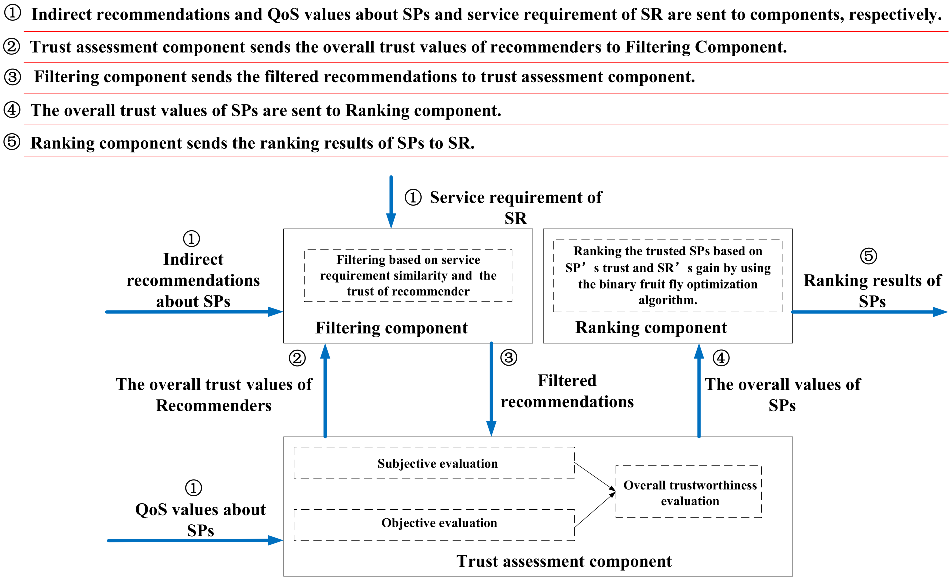

① Indirect recommendations and QoS values about SPs and service requirement of SR are sent to components, respectively.
② Trust assessment component sends the overall trust values of recommenders to Filtering Component.
③ Filtering component sends the filtered recommendations to trust assessment component.
④ The overall trust values of SPs are sent to Ranking component.
⑤ Ranking component sends the ranking results of SPs to SR.
① Service requirement of SR
① Indirect recommendations about SPs
Filtering based on service requirement similarity and the trust of recommender
Filtering component
Ranking the trusted SPs based on SP's trust and SR's gain by using the binary fruit fly optimization algorithm.
Ranking component
⑤ Ranking results of SPs
② The overall trust values of Recommenders
③ Filtered recommendations
④ The overall values of SPs
① QoS values about SPs
Subjective evaluation
Objective evaluation
Overall trustworthiness evaluation
Trust assessment component

Fig. 1. The system framework of TBSRS.

$R_1, R_2, ..., R_n (R_h \in [0,1])$. Bayesian estimation of the trust value of a service is used to evaluate the subjective trustworthiness of n_q

$$T^{subjective}_{q,i} = \frac{\sum_{h=1}^{n} R_h + n\psi^{j}_{q,i}}{2n} \tag{4}$$

where $\psi^{j}_{q,i}$ denotes the prior direct service experiences of requester n_j about s_i provided by n_q . If n_j has no prior service experiences, $\psi^{j}_{q,i}$ is set to 0, that is to say, the subjective trustworthiness value $T^{subjective}_{q,i}$ is obtained only based on the recommendations.

2) Objective evaluation

The objective trustworthiness is to predict the probability of completing its commitment of a service provider by comparing the difference between actual provisioned value and the promised value.

It is assumed that $v \in \{1, ..., V\}$. Let $\mathrm{Q}^{\mathrm{v}}_{\mathrm{qi}}$ and $\mathrm{Q}^{\mathrm{v}}_{\mathrm{qi}}$ present n_q's actual provisioned value and the promised value about the v^{th} attribute $Q^{v}_{q,i}$ for service s_i. $Y^{v}_{q,i}$ is denoted as n_q's service completing extent of $Q^{v}_{q,i}$. The bigger $Y^{v}_{q,i}$ is, the better service completing extent. $Y^{v}_{q,i}$ is calculated as follows:

$$Y^{v}_{q,i} = \begin{cases} \frac{Q^{v'}_{q,i} - Q^{v'}_{q,i}}{Q^{v'}_{q,i}}, q \, is \, benefit \\ \frac{Q^{v'}_{q,i} - Q^{v''}_{q,i}}{Q^{v'}_{q,i}}, q \, is \, \cos t \end{cases} \tag{5}$$

where there are two types of service attributes: benefit and cost. Based on $Y^{v}_{q,i}$, the objective trustworthiness level for the v^{th} attribute is defined as five levels [19]:

$$L^{v}_{q,i} = \begin{cases} 1\,(excellent) \quad if \; 0.5 \leqslant Y^{v}_{q,i} < 1 \\ 2\,(good) \quad if \; 0 < Y^{v}_{q,i} < 0.5 \\ 3\,(satisfactory) \quad if \; Y^{v}_{q,i} = 0 \\ 4\,(bad) \quad if \; -0.5 \leqslant Y^{v}_{q,i} < 0 \\ 5\,(very\,bad) \quad if \; -1 \leqslant Y^{v}_{q,i} < -0.5 \end{cases} \tag{6}$$

Here, Bayes learning theory is applied to measure the objective trustworthiness value of $Q^{v}_{q,i}$. Let $\overrightarrow{Y^{v}_{q,i}} = \left\{ Y^{v1}_{q,i}, Y^{v2}_{q,i}, ..., Y^{vX}_{q,i} \right\}$ represent the service completing extent information for the attribute $Q^{v}_{q,i}$, where X is the number of levels. $Y^{vX}_{q,i}$ represents the times of $Q^{v}_{q,i}$ appearing in the x^{th} level. Similar to the literature [19], we choose a Dirichlet prior over ε, i.e., $p\left(\overrightarrow{\varepsilon}\right) = Dir(\beta_1, ..., \beta_X)$. According to the property of the Dirichlet distribution, the posterior assessment ε becomes

$$\mathrm{E}(\varepsilon_x) = \frac{Y^{vx}_{q,i} + \beta_x}{\sum_{x=1}^{X} \beta_x + \sum_{x=1}^{X+1} Y^{vx}_{q,i}} \tag{7}$$

Equation (8) shows the probability of a certain attribute appears in the x^{th} trust level. The objective trustworthiness value of $Q^{v}_{q,i}$ is denoted as the overall probability of excellent, good, and satisfactory level which is given by:

$$T^{objective}_{Q^{v}_{q,i}} = \sum_{x=1}^{3} \mathrm{E}(\varepsilon_x) \tag{8}$$

$T^{objective}_{q,i}$ is the composite trustworthiness of n_q's multiple attributes for service s_i, and it is defined as the objective trustworthiness value of service provider n_q for service s_i:

$$T^{objective}_{q,i} = \sum_{v=1}^{V} \gamma_v T^{objective}_{Q^{v}_{q,i}} \tag{9}$$

where γ_v $(\gamma_v \in [0,1] \sum_{v=1}^{V} \gamma_v = 1X)$ is the weight of $\mathrm{Q}^{v}_{q,i}$.

3) Overall trustworthiness evaluation

The overall trustworthiness of a SP can be evaluated by aggregating both subjective trustworthiness and objective trustworthiness:

$$T_{q,i} = \kappa T^{subjective}_{q,i} + (1-\kappa) T^{objective}_{q,i} \tag{10}$$

The tunable parameter κ is used to balance objective and subjective trustworthiness.

If s_i is a new service, $T^{subjective}_{q,i}$ and $T^{objective}_{q,i}$ calculated by the above Equations will be zero due to the data sparsity problem. It is obviously unfair for a new service. Therefore, to overcome the data sparsity problem, our TBSRS will infer the trustworthiness $T_{q,i}$ based on the previous services provided by n_q. It is assumed that a service s_i provided by n_q has V attributes, and $Q^{v}_{q,i}$ is the v^{th} attribute,$v \in \{1, ..., V\}$. If the attribute $Q^{v}_{q,i}$ is included in the previous service s_e and is the g^{th} attribute of s_e, that is to say $Q^{v}_{q,i}$ and $Q^{g}_{q,e}$

is the same attribute. The objective trustworthiness value of $Q^{v}_{q,i}$ can be inferred as $T^{objective}_{Q^{g}_{q,e}}$. If the attribute $Q^{v}_{q,4}$ is not included in the previous services, the objective trustworthiness value of $Q^{v}_{q,4}$ can be given as 0.5. For example, service s_4 is a new service, which has three attributes:$Q^{1}_{q,4}$, $Q^{2}_{q,4}$ and $Q^{3}_{q,4}$. Their weights are 0.2, 0.6 and 0.2, respectively. $Q^{1}_{q,4}$ is included in the previous experienced service s_1 and trustworthiness in service s_1 is 0.6. $Q^{2}_{q,4}$ is included in the previous experienced service s_2 and trustworthiness in service s_1 is 0.8. $Q^{3}_{q,4}$ isn't included in the previous experienced services, so its trustworthiness is set to 0.5. Based on the above inferring method, the trustworthiness $T_{q,4}$ is: $0.6 \times 0.2 + 0.8 \times 0.6 + 0.5 \times 0.2 = 0.7$.

C. Ranking component

To rank the trustworthy service providers, our scheme uses the binary fruit fly optimization algorithm [20] shown in Fig. 2. The process includes two steps: Initialization and Iteration.

Step one:

The algorithm firstly initializes population size (pop_{size}), maximum number of generations ($gen_{\max}$), local best smell concentration ($smell_{best}$), global best smell concentration ($gsmell_{best}$), best position (pos_{best}), global best position ($gpos_{best}$), fitness value ($fitness$) and random position of "m" fruit flies with appropriate values, shown from line 1 to line 12. The position of a certain fruit fly is denoted as "z" tuple vectors. "z" is the number of trustworthy service providers. If the binary position of fruit fly j_s is $(0,0,1,1,0,0)$, it shows n_3, n_4 are present and n_1, n_2, n_5, n_6 are absent on the position. Global best position ($gpos_{best}$) stores the position of the fruit fly with high fitness value and the position is a seed in the next iteration. The fitness of each fruit fly is calculated with Equation (11).

$$fitness = \sigma T_{p,i} + (1-\sigma)G_{j,i} \tag{11}$$

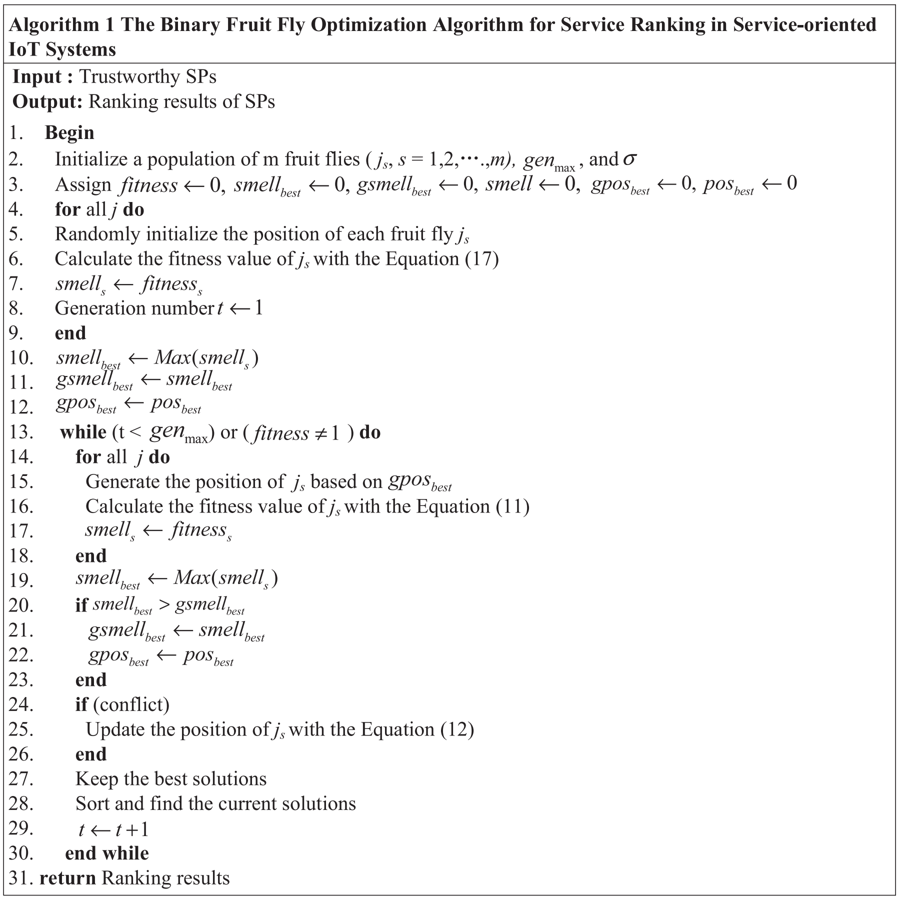
where $T_{p,i}$ denotes the trustworthiness of service provider n_q for the service s_i. σ is a weight value. $G_{j,i}$ denotes the gain of a service requester n_j using the service provided by n_q. In experimental section, the gain of a service requester is randomly set as a value of [0,1].

Step two:

The random position in the current iteration is generated in the initialization step. Then, the fitness value of each fruit fly is evaluated based on current position. $smell_{best}$ and pos_{best} are updated with the fitness value and position of the fruit fly with high fitness value respectively. The new $smell_{best}$ is compared with the last $gsmell_{best}$. If $smell_{best}$>$gsmell_{best}$, then $gsmell_{best}\leftarrow$ $smell_{best}$; $gpos_{best}\leftarrow pos_{best}$. Subsequently, the new $gsmell_{best}$ and $gpos_{best}$ are set as a seed for the next generation. In order to obtain the optimal service ranking result, the algorithm will check whether the value of fitness is 1 or not, shown from line 13 to line 23. In addition, if the current global best solution is the same with the solution from the last iteration, the Equation (12) [12] will be used to update the position of fruit flies, shown from line 24 to line 26.

$$p(t+1) = Mutation(p(t)) \tag{12}$$

The iteration process is completed until one of the two conditions ($fitness = 1$ or $gen_{\max}$) is met.



Algorithm 1 The Binary Fruit Fly Optimization Algorithm for Service Ranking in Service-oriented IoT Systems

Input : Trustworthy SPs
Output: Ranking results of SPs

```
1.  Begin
2.    Initialize a population of m fruit flies ( $j_s$, $s = 1,2,\cdots,m$), $gen_{\max}$ , and $\sigma$
3.    Assign $fitness \leftarrow 0$, $smell_{best} \leftarrow 0$, $gsmell_{best} \leftarrow 0$, $smell \leftarrow 0$, $gpos_{best} \leftarrow 0$, $pos_{best} \leftarrow 0$
4.    for all $j$ do
5.    Randomly initialize the position of each fruit fly $j_s$
6.    Calculate the fitness value of $j_s$ with the Equation (17)
7.    $smell_s \leftarrow fitness_s$
8.    Generation number $t \leftarrow 1$
9.    end
10.   $smell_{best} \leftarrow Max(smell_s)$
11.   $gsmell_{best} \leftarrow smell_{best}$
12.   $gpos_{best} \leftarrow pos_{best}$
13.   while (t < $gen_{\max}$) or ( $fitness \neq 1$ ) do
14.     for all $j$ do
15.       Generate the position of $j_s$ based on $gpos_{best}$
16.       Calculate the fitness value of $j_s$ with the Equation (11)
17.       $smell_s \leftarrow fitness_s$
18.     end
19.     $smell_{best} \leftarrow Max(smell_s)$
20.     if $smell_{best} > gsmell_{best}$
21.       $gsmell_{best} \leftarrow smell_{best}$
22.       $gpos_{best} \leftarrow pos_{best}$
23.     end
24.     if (conflict)
25.       Update the position of $j_s$ with the Equation (12)
26.     end
27.     Keep the best solutions
28.     Sort and find the current solutions
29.     $t \leftarrow t+1$
30.   end while
31. return Ranking results
```

Fig. 2. The binary fruit fly optimization algorithm.

4. Experimental analysis

4.1. Experiment setup

We simulate a service-oriented network system with N_n nodes. The experimental data come from realistic dataset (Epinions Trust Network Datasets). We slight modify them in the experiments. Table 1 lists the parameters and their values/ ranges used in the performance validation.

4.2. Performance metrics

The main performance metrics are shown as follows:

1) **The difference between predicted trust value and actual trust value:** The smaller the difference, the higher accuracy of trustworthiness computation.

2) **The percentage of malicious nodes selected as SPs for a good SR:** The smaller the number, the higher the capacity of filtering out malicious recommendations. The percentage is expressed as follows:

$$P_{m,sp} = \left(\frac{N_{m,sp}}{N_{sp}}\right) \times 100\% \tag{13}$$

N_{sp} is the number of SPs selected to provide service for a good SR. $N_{m,sp}$ is the number of malicious nodes in N_{sp}.

3) **The precision of service selection:** The bigger the number, the higher the precision of service selection. The precision is expressed as follows:

$$precision = \left(\frac{N_{satisfaction}}{N_{sp}}\right) \times 100\% \tag{14}$$

N_{sp} is the number of SPs selected to provide service for a good SR. $N_{satisfaction}$ is in N_{sp}, and it is the number of SPs providing the satisfied service.

4) **The percentage of good nodes providing new service selected as SPs for a good SR:** The bigger the number, the higher the capacity of assessing a new service. The percentage is expressed as follows:

$$P_{sp,new} = \left(\frac{N_{sp,new}}{N_{new}}\right) \times 100\% \tag{15}$$

N_{new} is the number of good nodes providing new service.$N_{sp,new}$ is the number of good nodes providing new service selected as SPs in N_{new}. The performance of our TBSRS is compared against: Adaptive IoT Trust [17] and IoT-HiTrust [3]. The reason we select Adaptive IoT Trust and IoT-HiTrust is that their SP's trustworthiness evaluation method is similar with our SP's trustworthiness computation method. Two approaches and our TBSRS are both trust-based service selection methods. In addition, the two approaches outperform existing trust-based service selection ones for SOEs.

Table 1
Parameter list for performance evaluation.

Symbols	Meaning	Default	Range
N_n	Number of nodes	100	[50–300]
N_a	Number of service attributes	3	[1–5]
N_s	Number of service types	2	[2–5]
N_f	Number of user requirement's feature	3	[2–5]
$T_{q,i}$	Initial trustworthiness of any node	0.5	[0,1]
$G_{j,i}$	Initial gain of any node	0.5	[0,1]
ι	Service request rate	5 / hr	[2–10]
P_m	Percentage of malicious nodes	10%	[10%-70%]
λ	Similarity weight factor	0.5	[0–1]
κ	Weight factor of objective and subjective trustworthiness	0.5	[0–1]
σ	Weight factor of trustworthiness and gain	0.5	[0–1]
N_{sc}	Number of selected candidate SPs	Top-4%	[Top-1%-Top-13%]

4.3. Performance validation

4.3.1. TBSRS performance

In the first experiment, the trustworthiness of a malicious SP and a good SP is calculated by our TBSRS from the perspective of a good SR as time progresses. Fig. 3 shows the experimental results. When the experiment begins, the trustworthiness of malicious SP and good SP goes down and rises up, respectively. The main reason is that TBSRS effectively filter out untrustworthy assessment data based on the proposed recommendation filtering algorithm.

In the second experiment, the trust difference of a SP's predicted trustworthiness and the actual trustworthiness is measured from the perspective of a good SR as time progresses. Fig. 4 shows the experimental results. One can observe that the trust difference firstly goes down slowly, then goes down quickly, finally converges to 0. The main reason is that when the experiments start, the trustworthiness assessment data is sparse in which case the malicious recommendations would have a great influence on the trustworthiness computation, so the predicted trustworthiness only slowly closes to the actually trustworthiness. As time passes, more trustworthy assessment date is gathered in which case the malicious recommendations would have a little influence on the trustworthiness computation, so the predicted trustworthiness quickly closes to the actually trustworthiness until it is equal to the actual trustworthiness. The third experiment shows the impact of different value of Similarity weight factor λ on the precision of service selection. Top-13% SPs are selected in the experiment. The results are shown in Fig. 5. We are able to see that when λ equals to 0.5, the value of precision of service selection reaches the highest point under the same simulation condition. Thus, we can get the conclusion that the algorithm can achieve better result under the condition of $\lambda = 0.5$.

4.3.2. Comparative analysis

The first experiment compares the three methods in selecting malicious nodes as SPs for a good SR. Fig. 6 shows the percentage of malicious SPs selected by a good SR as time progresses. The pink curve expresses Adaptive IoT Trust. The green curve expresses IoT-HiTrust. The blue curve expresses our TBSRS. The trustworthiness of all SPs is initially set to 0.5, so malicious SRs initially could have the chances selected as SPs to provide the service. Under our TBSRS, the percentage of malicious nodes selected is the lowest in comparison with Adaptive IoT Trust and IoT-HiTrust.

The second experiment compares the three methods in the precision of service selection. The Top-1%, Top-4%, Top-7%, Top-10% and Top-13% SPs are selected respectively and the selection results by three methods will be compared and shown in Fig. 7. One can observe that TBSRS outperforms both Adaptive IoT Trust and IoT-HiTrust in terms of the precision performance metric of service

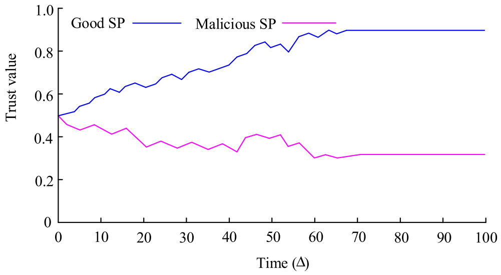


Fig. 3. Trustworthiness of a good SP and a malicious SP vs. time.

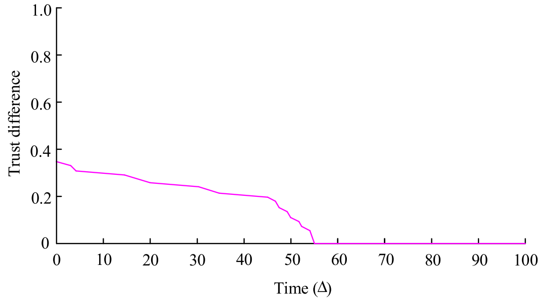


Fig. 4. Trust difference of a SP's predicted trustworthiness and the actual trustworthiness.

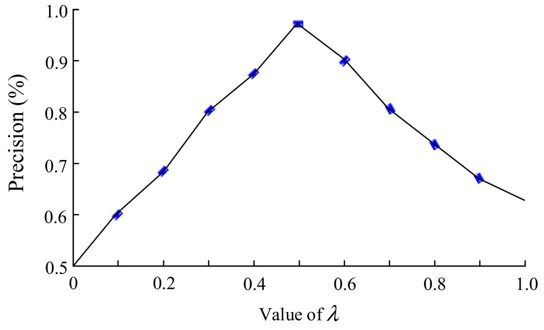


Fig. 5. The influence of λ on precision of service selection.

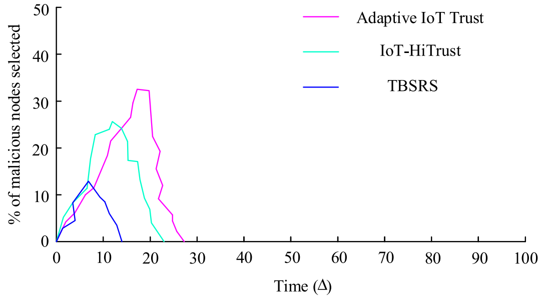


Fig. 6. Percentage of malicious nodes selected for service over time under Adaptive IoT Trust, IoT-HiTrust and TBSRS.

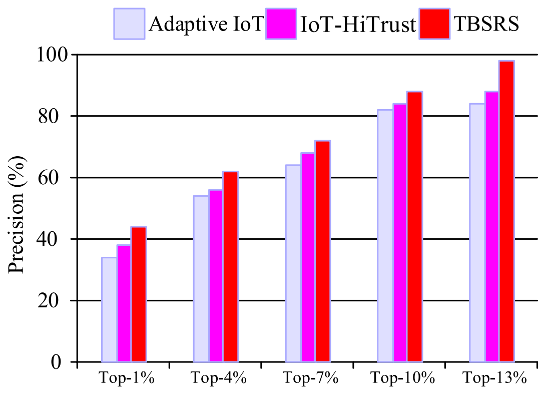


Fig. 7. The precision of service selection under Adaptive IoT Trust, IoT-HiTrust and TBSRS.

selection. In addition, when selecting the top-13% candidate SPs, TBSRS has a very high precision, which is close to 100%. This observation implies that, Top-13% candidate SPs are recommended by us to the SR.

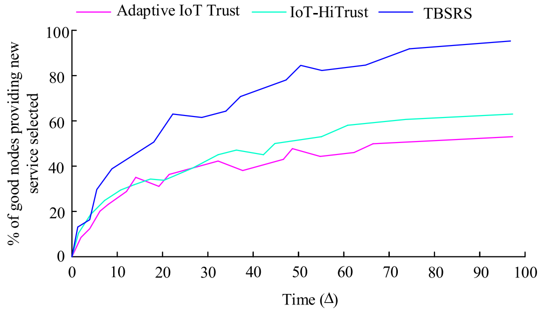


Fig. 8. Percentage of good nodes providing new service selected for service over time under Adaptive IoT Trust, IoT-HiTrust and TBSRS.

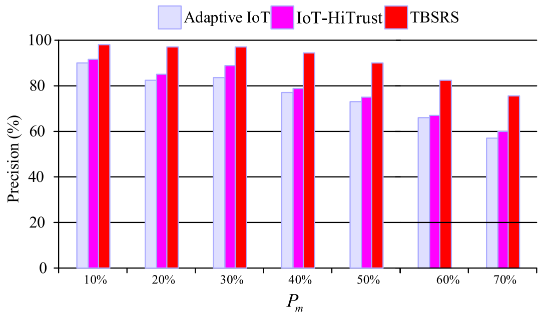


Fig. 9. The influence of P_m on precision of service selection.

The third experiment compares the three methods in the percentage of good nodes providing new service selected as SPs. The experimental results are shown in Fig. 8.

One can observe that TBSRS outperforms both Adaptive IoT Trust and IoT-HiTrust in terms of the percentage of good nodes providing new service selected as SPs. The experiments results show that more new services are selected by SRs due the accurate trustworthiness of new service in TBSRS as time processes. The fourth experiment shows the impact of different value of Percentage of malicious nodes P_m on the precision of service selection. Top-13% SPs are selected in the experiment.

One again observes that TBSRS outperforms Adaptive IoT Trust and IoT-HiTrust. An important observation from Fig. 9 is that, even when malicious recommenders add to 70%, the precision of service selection of TBSRS keep also remarkably more than 0.75. The reason is that our TBSRS provides the optimal ranking for SPs by considering both service provider's trustworthiness and service requester's benefit.

5. Conclusion

In the paper, an efficient trust-based service ranking scheme (TBSRS) is proposed for providing accurate trust evaluation and service ranking in the absence of assessment data or presence of untrustworthiness recommendations. The entire framework of TBSRS comprises of three components, namely (i) Filtering component (ii) Trustworthiness assessment component (iii) Ranking component. The convergence, accuracy and resiliency properties of TBSRS is analyzed and validated via simulation. A comparative analysis of TBSRS is also performed against Adaptive IoT Trust and IoT-HiTrust. The experimental results validate by simulation demonstrate that TBSRS outperforms these existing approaches in terms of the precision of service selection and other a few performance metrics. TBSRS will be applied to mobile social IoT applications, where the mobility of nodes is considered. In service ranking component, how to measure the gain of a service requester is not discussed in our paper, so it will be furtherly researched. Lastly, TBSRS will be further tested in complicated multi-domain scenarios, and will extend our work in terms of dealing with more sophisticated attack behaviors such as opportunistic, collusion, and insidious attacks.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

The work in this paper has been supported by National Natural Science Foundation of China (62062006, 71501156, 62062008, 62067001, 61762008), Natural Science Basis Research Plan in Guangxi Province of China (2018JJA170028, 2019JJA170045, 2018JJA170194), Special Funds of Guangxi Bagui Scholars and National Key Research and Development Program of China (2018YFB1404404).

References

[1] Yating Wang, Ing-Ray Chen, Jin-Hee Cho, et al., "CATrust: Context-Aware Trust Management for Service-Oriented Ad Hoc Networks, IEEE Trans Services Comput 2018; 11(6): 908–221.

[2] Lin Z, Dong L. Clarifying trust in social internet of things. IEEE Trans Knowl Data Eng 2018;30(2):234–48.

[3] Cho JH, Swami A, Chen I-R. Modeling and analysis of trust management with trust chain optimization in mobile Ad hoc networks. J Netw Comput Appl 2012;35(3):1001–12.

[4] Nitti M, Girau R, Atzori L. Trustworthiness management in the social Internet of Things. IEEE Trans Knowl Data Manag 2014;26(5):1253–66.

[5] Chen I-R, Guo J, et al. Trust-based service management for mobile cloud IoT systems. IEEE Trans Netw Service Manage 2019;16(1):246–63.

[6] Hamid A-H, Chen I-R, Cho JH. Trust management of smart service communities. IEEE Access 2019. doi: https://doi.org/10.1109/ACCESS.2019.2901023.

[7] Wu X, Cheng B, Chen J. Collaborative filtering service recommendation based on a novel similarity computation method. IEEE Trans Services Comput 2017;10(3):352–65.

[8] Chen IR, Guo J, Bao F. Trust management for SOA-based IoT and its application to service composition. IEEE Trans Services Comput 2016;9(3):482–95.

[9] Guo J, Chen I-R, Tsai JJP. A survey of trust computation models for Internet of Things systems. Comput Commun 2017;97:1–14.

[10] Wang Y, Chen IR, Cho JH, Swami A, Chan K. Trust-based service composi- tion and binding with multiple objective optimization in service-oriented mo- bile Ad Hoc networks. IEEE Trans Serv Comput 2016. doi: https://doi.org/10.1109/TSC.2015.2491285. in press.

[11] Li YY, Huang YS, et al. Service selection mechanisms in the Internet of Things (IoT): a systematic and comprehensive study. Clust Comput 2020;23:1163–83.

[12] Adewuyi AA, Cheng H, Shi Q, Cao J, Wang X, Zhou B. SC-TRUST: a dynamic model for trustworthy service composition in the internet of things. IEEE Internet Things J 2022;9(5):3298–312.

[13] Xia H, Li Z, Zheng Y, Liu A, Choi Y, Sekiya H. A novel lightweight subjective trust inference framework in MANETs. IEEE Trans Sustain Comput doi: 10.1109/TSUSC.2018.2817219.

[14] Fan L, Gary W, et al. A trust model for SLA negotiation candidates selection in a dynamic IoT environment. IEEE Trans Services Comput 2022;15(5):2565–78.

[15] Wang H, Yang D, et al. Integrating modified cuckoo algorithm and creditability evaluation for QoS-aware service composition. Knowl-Based Syst 2019;140:64–81.

[16] Adomavicius G, Zan H, Tuzhilin A. Personalization and recommender systems. INFORMS Tutorials Operations Res 2014;14(1):55–107.

[17] Ding S, Xia CY, Zhou KL, Yang SL, Shang JS. Decision support for personalized cloud service selection through multi-attribute trustworthiness evaluation. PLoS One 2014;9(6):e97762. doi: https://doi.org/10.1371/journal.pone.0097762. PMID: 24972237.

[18] Viriyasitavat W, Xu LD, Bi ZM. User-oriented selections of validators for trust of internet-of thing services. IEEE Trans Ind Inform 2022;18:4859–67.

[19] Wang HB, Yang DR, Yu Q, Ta Y. Integrating modified cuckoo algorithm and creditability evaluation for QoS-aware service composition. Knowl-Based Syst 2018;140:64–81.

[20] Shen L, Chen H, Yu Z, Kang W, Zhang B, Li H, Yang B, Liu D. Evolving support vector machines using fruit fly optimization for medical data classification. Knowl-Based Syst 2016;96:61–75. doi: https://doi.org/10.1016/j.knosys.2016.01.002.

[21] Cai B, Li XY, et al. A reliable and lightweight trust inference model for service recommendation in SIoT. IEEE Internet Things J 2022;9(13):10988–1003.