Decision Support for Personalized Cloud Service Selection through Multi-Attribute Trustworthiness Evaluation

Shuai Ding1,2*, Chen-Yi Xia3*, Kai-Le Zhou1,2, Shan-Lin Yang1,2, Jennifer S. Shang4
1 School of Management, Hefei University of Technology, Hefei, P.R. China, 2 Key Laboratory of Process Optimization and Intelligent Decision-Making, Ministry of Education, Hefei, P.R. China, 3 Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology and Key Laboratory of Computer Vision and System (Ministry of Education), Tianjin University of Technology, Tianjin, P.R. China, 4 The Joseph M. Katz Graduate School of Business, University of Pittsburgh, Pittsburgh, Pennsylvania, United States of America

Abstract

Facing a customer market with rising demands for cloud service dependability and security, trustworthiness evaluation techniques are becoming essential to cloud service selection. But these methods are out of the reach to most customers as they require considerable expertise. Additionally, since the cloud service evaluation is often a costly and time-consuming process, it is not practical to measure trustworthy attributes of all candidates for each customer. Many existing models cannot easily deal with cloud services that have very few historical records. In this paper, we propose a novel service selection approach in which the missing value prediction and the multi-attribute trustworthiness evaluation are commonly taken into account. By simply collecting limited historical records, the current approach is able to support the personalized trustworthy service selection. The experimental results also show that our approach performs much better than other competing ones with respect to the customer preference and expectation in trustworthiness assessment.

Introduction

Cloud computing has become the driver for innovation in the recent years, from startups (e.g. Dropbox, Instagram) to established enterprises (Samsung). They are all using cloud computing to better serve their customers around the world [1]. Cloud service is also gaining wide acceptance and becoming popular to individuals as it reduces hardware and licensing costs, and it is scalable and allows users to work from any computer anywhere.

Several leading IT enterprises including Google, IBM, Microsoft, and Amazon have started to offer cloud services to their customers [2–4]. While many small and medium-sized enterprises (SMEs) and individual customers prefer to apply cloud services to build their business system or personal applications, they are often facing two major challenges at the selection time: (1) multiple cloud services are often available by different vendors providing similar functional properties (i.e., “functionally-equivalent”); Customers usually lack appropriate, qualified, sufficient information and benchmarks to assess cloud services with regard to individual preferences and market dynamics [5]; (2) although cloud service vendors are struggling to improve service quality and performance, cloud computing are not necessarily trustworthy – unhandled exceptions and crashes may cause cloud service to deviate dramatically from the expectation [6,7]. Therefore, there is an increasing demand to help the non-expert customers with the selection of trustworthy cloud service.

The trustworthiness of cloud service affects customers' perception towards service quality, which has significant bearing on customer satisfaction and royalty. The trustworthy attributes include reliability, scalability, availability, safety, security, etc [8–10]. Designing a general and comprehensive analytical model for trustworthiness evaluation is challenging, as the model needs the assessor to achieve, in reasonable time, useful results to determine the best service option. Due to their commercial value (similar to online recommendation system), several evaluation models [11–14] have been proposed by academia and industry lately. These models focus on quantitative analysis and evaluate trustworthiness through a collectively exhaustive dataset.

Except for some specific cases, the assessment dataset remains very sparse due to the costly and time-consuming nature of cloud service invocation. Intuitively, without sufficient data, fair review of cloud services cannot be achieved by existing evaluation methods [9,15]. Fortunately, cloud vendors can collect historical records (QoS values, customer ratings, etc) from different cloud applications in cloud computing environment. With the vast amount of collaborative filtering (CF) technologies available in the field of online recommendation system, we believe there is a strong...
theoretical foundation to derive a generic trustworthiness model to support the evaluation of cloud service.

There have been some attempts to improve the accuracy of cloud service assessment by a CF process. However, very little attention is paid to the trustworthiness of cloud service, and no interest is given to the case when significant attribute values are missing. The lack of general and formal methodology can be attributed to the large process gap between the cloud service recommenders and trustworthiness researchers. To deal with this challenge, we propose a new CF approach to make use of hidden information (i.e. experience usability, value distribution) to measure the similarity between different services. Moreover, to support personalized selection of cloud services, we also provide a natural treatment for multi-attribute aggregation taking into account customer’s preference and expectation.

### Background

In the current market, multiple cloud services of similar functions are often available for specific domains. For example, in cloud storage service (e.g. data service, online file system, online backup plan), over 100 functionally-equivalent cloud services are offered by vendors. Some typical examples can be found in Table 1. Given the lack of cloud computing experience of non-expert customers, it is tedious to manually select an appropriate candidate from a set of functionally-equivalent services. Therefore, cloud service evaluation through quality analysis has gained much attraction among service-oriented computing and cloud computing communities over the past two decades.

Given the intricate interactions among QoS (Quality of Service) attributes, customer preferences and market dynamics that jointly influence the perceived quality of cloud services, developing a market-relevant analytical model is crucial to cloud service selection [16–18]. Due to their commercial value and the associated research challenges, many researchers and practitioners have studied the topics. Two types of service selection models are widely examined: evaluation-focused service selection models and prediction-focused service selection models.

By achieving market-relevant evaluations, customers can identify risks and benefits of each cloud service application and choose the best for adoption. The most employed evaluation models include: AHP-based cloud service ranking [19], reputation-aware service rating [20], trust-aware service selection [21], brokerage-based selection [22], SLA-based cloud trustworthiness estimation [11], trustworthy service selection [23]. Although these techniques can accurately and exhaustively estimate service quality, their implementation is time-consuming and costly.

Instead of real-world cloud service invocations, the prediction-focused service selection models can produce QoS values or service ranking using collaborative filtering (CF). The CF approaches for cloud service selection can be categorized as: item-based approaches [24], customer-based approaches [25], their fusion approaches [26], model-based approaches [27], and ranking-oriented approaches [28], where the first three categories are rating-oriented approaches. These approaches help assessors predict the missing attribute values by exploiting neighbors’ usage experiences. Several collaborative filtering approaches for cloud service selection have been studied, but they did not consider customer preference and expectation in trustworthiness assessment.

In the prediction process, similar neighbors (customers or services) are identified to generate useful collaborative information. Popular choices for similarity estimation include Pearson correlation coefficient (PCC) [29] and vector similarity (VS) [30]. Since these measures only consider the numerical relationship between different ratings, they remain imprecise and confusing for estimating the neighbor similarity to support missing value

### Table 1. Online cloud storage services.

| Vender            | Cloud Service       | Feature          | Pricing                                      |
|-------------------|---------------------|------------------|----------------------------------------------|
| Amazon            | EBS                 | Storage Service  | $0.1 per GB-month, $0.1 per 1 million I/O requests |
| Amazon            | S3 Standard         | Storage Service  | $0.095 per GB-month, $0.005 per 1000 requests |
| Google            | Google Cloud Storage| Storage Service  | $0.085 per GB-month, $0.01 per 1000 ops-month |
| IBM               | SoftLayer Object Storage | Storage Service | $0.1 per GB-month |
| Microsoft         | Azure Data Service  | Storage Service  | $0.095 per GB-month, $0.01 per 100000 I/O requests |
| Apple             | iCloud              | Storage Service  | $20 for 10 GB upgrade                       |
| GoGrid            | GoGrid Cloud Storage| Storage Service  | $0.12 per GB-month                         |
| JustCloud         | JustCloud Cloud Storage | Storage Service | $1.95 per month, unlimited storage          |
| ZipCloud          | ZipCloud Online Storage | Storage Service | $6.95 per month, unlimited storage         |
| AT&T              | Synaptic Storage    | Storage Service  | Unknown                                      |
| LiveDrive         | Livedrive Backup Plan | Backup System | $6 per month, 2 TB storage space           |
| CrashPlan         | CrashPlan Backup Plan | Backup System | $5.99 per month, unlimited storage        |
| Carbonite         | Cloud Backup Services | Backup System | $39.99 per year, unlimited storage        |
| FlexiScale        | FlexiScale Public Cloud | Platform Service | $17 per 1000 unit-hour                        |
| AppNexus          | AppNexus Cloud      | Platform Service  | Unknown                                      |
| Rackspace         | Mosso cloud files   | File System      | $0.75 per GB-month                          |
| HighTail          | HighTail            | File System      | $15.99 per month, unlimited storage         |
| Amazon            | SimpleDB            | Database         | $0.10 per GB-month                          |

doi:10.1371/journal.pone.0097762.t001
prediction. Concerned that PCC may overestimate the similarities of negative services, Zheng et al. [26] propose a significance weight and modify PCC to improve the accuracy of similarity computation in service recommendation. However, the significance weight affects the similarity computation of positive services with more usage experiences. To address this problem, Ding et al. [31] define a convex function (usage structure factor) to reflect the usability of customer experience.

While a great number of researchers have focused on the trust-aware service selection and recommendation, little attention has been devoted to the role of customer preference and expectation in multi-attribute trustworthiness evaluation [32]. In addition, large quantities of works offer some valuable clues to discern between different services, the significances arising from value distribution is seldom considered. Thus, we will here combine evaluation-focused and prediction-focused approaches to propose a novel trustworthiness evaluation method which will fully utilize the information of similar services and customer’s experience, and take into account both the missing attribute value prediction and the multi-attribute trustworthiness evaluation at the same time.

Methods and Materials

Based on the fact that the size and rate of growth in customers outweigh the expansion of delivered services in the cloud computing market, we employ item-based CF approach rather than the user-based or their fusion approach to produce the missing attribute values in trustworthiness evaluation. Motivated by the observation that experience usability and value distribution could provide valuable insight and distinctive information in the CF process, we create a new similarity measure for enhancing the prediction performance.

Pearson Correlation Coefficient

To make an accurate prediction, we first estimate the similarity between different cloud services. Given a service selection problem consisting of \( M \) customers and \( N \) services, the customer-service matrix for missing value prediction is denoted as

\[
\begin{bmatrix}
q_{1,1} & \ldots & q_{1,N} \\
\vdots & \ddots & \vdots \\
q_{M,1} & \ldots & q_{M,N}
\end{bmatrix}
\]  

(1)

where the entry \( q_{m,n} \) denotes a historical record (QoS value or customer rating) of cloud service \( cs_n \) made by customer \( u_m \); “\( q_{m,n} = \text{null} \)” states that \( u_m \) didn’t invoke \( cs_n \) yet.

Pearson Correlation Coefficient (PCC). [29] Taking use of numerical distance to estimate the correlation between different services, PCC has been successfully adopted for recommendation system evaluations. Let \( cs_k \) and \( cs_l \) be two services, \( U_{ns} \) be the subset of customers who have invoked both \( cs_k \) and \( cs_l \), then PCC is applied to calculate the similarity between \( cs_k \) and \( cs_l \) by

\[
\text{Sim}(cs_k, cs_l) = \frac{\sum_{m \in U_{ns}} (q_{m,n} - \bar{q}_n)(q_{m,l} - \bar{q}_l)}{\sqrt{\sum_{m \in U_{ns}} (q_{m,n} - \bar{q}_n)^2} \sqrt{\sum_{m \in U_{ns}} (q_{m,l} - \bar{q}_l)^2}}
\]

(2)

where \( \text{Sim}(cs_k, cs_l) \) is in the interval of \([-1, 1]\), \( \bar{q}_n \) and \( \bar{q}_l \) stand for the average values of \( cs_k \) and \( cs_l \) made by different customers. However, as noted in Ref. [26], PCC always overestimate the similarities of negative services, which are actually not similar but happen to have similar usage experience made by few customers. Table 2 shows a simple customer-service matrix which contains six customers (\( u_1 \) to \( u_6 \)) and ten cloud services (\( cs_1 \) to \( cs_{10} \)). When utilizing Eq. (2), we calculate the PCC values between the services, and get the following relation: \( \text{Sim}(cs_1, cs_2) > \text{Sim}(cs_2, cs_1) \geq \text{Sim}(cs_1, cs_3) \), which indicates \( cs_3 \) is more similar to \( cs_1 \) than \( cs_2 \) and \( cs_4 \). It is clearly contrary to the reality due to the limited usage experience. Therefore, it is necessary to reinforce the similarity information in the CF process.

Significance estimation

It seems logical to believe that some cloud services in customer-service matrix may have high significances in making recommendations [33,34]. For instance, a cloud service, which has more useful historical records, may be regarded as more important compared with a negative service. PCC is only related to the numerical distance between different services, but it has nothing to do with the statistical features of historical records. For this case, we introduce two types of significances arising from the experience usability and value distribution of historical records, respectively.

Estimating the experience usability. To determine the significance of neighbors in a CF process, one often assumes a linear relationship between usage experiences and neighbor significances [26,31]. One difference of our work from traditional CF approaches is that we apply a distance measurement method to estimate the experience usability in customer-service matrix. During the distance measurement, Jaccard’s coefficient [35] is frequently employed to estimate the discrimination of asymmetric information on binary variables. Before integrating Jaccard’s coefficient into our similarity measure, we map the original customer-service matrix into a rectangular binary matrix as follows:

\[
\begin{bmatrix}
q_{1,1} & \cdots & q_{1,N} \\
\vdots & \ddots & \vdots \\
q_{M,1} & \cdots & q_{M,N}
\end{bmatrix}
\]

\[
\begin{cases}
H(\text{null} \rightarrow \text{null}) = 0 & \text{if } q_{m,n} = \text{null} \\
H(\text{null} \rightarrow \text{1}) = 1 & \text{if } q_{m,n} = \text{1} \\
H(\text{1} \rightarrow \text{null}) = 1 & \text{if } q_{m,n} = \text{null} \\
H(\text{1} \rightarrow \text{1}) = 0 & \text{if } q_{m,n} = \text{1}
\end{cases}
\]  

(3)

where \( H(\text{null} \rightarrow \text{null}) = 1 \) denotes the customer \( u_m \) has invoked the service \( cs_n \) previously, whereas \( H(\text{null} \rightarrow \text{1}) = 0 \) denotes that \( u_m \) didn’t invoke \( cs_n \). Let \( |U_n| \) be the number of customers who has invoked \( cs_n \) before, and \( |U_{ns}| \) be the number of customers who invoked both \( cs_n \) and \( cs_l \). We use the Jaccard’s coefficient \( J_{ns} \) to reflect the rise of significance due to the experience usability, which can be expressed mathematically as:

\[
J_{ns} = \frac{|U_n|}{|U_n| + |U_{ns}| - |U_{ns}|} = \frac{\sum_{m=1}^{M} (b_{m,n} \cap b_{m,l})}{\sum_{m=1}^{M} (b_{m,n} \cup b_{m,l})}
\]

(4)

where \( J_{ns} \) is in the interval of [0, 1], and a larger \( J_{ns} \) value indicates that the historical records made over \( cs_n \) is more useful in the CF process. \( J_{ns} = J_{ls} \) holds for all services, which is consistent with the intuition that the similarity between \( cs_n \) and \( cs_l \) is only related to the subset of historical records made by the customers who have invoked both \( cs_n \) and \( cs_l \).

Based on the customer-service matrix in Table 2, we get the significances arising from the experience usability for each service, as shown in Table 3. The values shown in grey are calculated for the negative service \( cs_3 \). As observed from Table 2, \( cs_3 \) has only been invoked twice. Consequently, his experience usability values
are significantly lower than other services (e.g., $J_{1.3} < J_{1.2}$). We can infer that integrating $J_{n}$ into similarity measure will notably reduce the influence of negative service. It is worth noting that neither PCC nor $J_{n}$ can distinguish between $cs_2$ and $cs_4$, since they do not have sufficient power to detect the crucial difference in value distributions.

**Estimating the value distribution.** The neighbors which have the same PCC similarity may have different values in the customer-service matrix for significance estimation. For this case, we propose a method to discriminate neighbors’ significances arising from their unique value distributions. In practice, the customer-service matrix is very sparse due to limited usage experiences. Therefore, we will ignore the historical records made over $cs_n$ and $cs^\prime$, since they do not have sufficient power to detect the crucial difference in value distributions.

$V_{D_{n,y}} = 1 - \frac{H_{n,y}}{H_{\text{max}}}$, where $H_{n,y} = \sum p_k^y \log_2(p_k^y)$ denotes the information entropy of $D_{n,y}$ and $H_{\text{max}}$ denotes the maximal entropy in customer-service matrix, respectively. $V_{D_{n,y}}$ is a linear function defined in $[0, 1]$. From the maximum entropy principle [36], we have $H_{\text{max}} = \log_2(3)$. Thus, Eq.(8) can be rewritten as:

$$V_{D_{n,y}} = 1 - \sum p_k^y \log_2(p_k^y) \cdot \frac{\log_2(3)}{\log_2(3)}$$

Since $D_{n}$ is a finite discrete dataset, the probability of each category can be computed as:

$$\{ p_1 = \frac{|D_n|}{|D_y|}, p_2 = \frac{|D_{n}^-|}{|D_y|}, p_3 = \frac{|D_{n}^\prime|}{|D_y|} \}$$

Table 2. A simple customer-service matrix.

| $cs_1$ | $cs_2$ | $cs_3$ | $cs_4$ | $cs_5$ | $cs_6$ | $cs_7$ | $cs_8$ | $cs_9$ | $cs_{10}$ |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| $u_1$  | 0.9    | 0.7    | null   | 1      | 0.8    | 0.4    | 0.7    | null   | null   |
| $u_2$  | 0.8    | 0.7    | 0.8    | 0.4    | 0.6    | 0.7    | null   | 0.9    | 0.8    |
| $u_3$  | 0.9    | 0.8    | null   | 0.6    | 0.5    | 0.5    | null   | 0.7    | 0.6    |
| $u_4$  | 0.8    | 0.9    | null   | 1      | null   | 0.8    | 0.6    | 0.7    | 0.6    |
| $u_5$  | 0.7    | 0.6    | null   | 0.5    | 0.7    | null   | 0.8    | null   | 0.4    |
| $u_6$  | null   | 0.8    | 0.9    | 0.6    | null   | null   | 0.9    | 0.8    | 0.7    |

Table 3. Significances arising from the experience usability.

| $cs_1$ | $cs_2$ | $cs_3$ | $cs_4$ | $cs_5$ | $cs_6$ | $cs_7$ | $cs_8$ | $cs_9$ |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 0.833  | 0.167  | 0.333  | 0.833  | 0.8    | 0.667  | 0.667  | 0.667  | 0.333  |
| 0.833  | 0.333  | 0.25   | 0.333  | 0.333  | 0.333  | 0.333  | 0.667  | 0.4    |
| 0.5    | 0.667  | 0.5    | 0.667  | 0.6    | 0.667  | 0.6    | 0.667  | 0.167  |
| 0.5    | 0.667  | 0.5    | 0.667  | 0.6    | 0.667  | 0.6    | 0.667  | 0.167  |
Table 4. Significances arising from the value distribution.

|   | CS1 | CS2 | CS3 | CS4 | CS5 | CS6 | CS7 | CS8 | CS9 | CS10 |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
| CS1 | 1   | 1   | 1   | 0.369 | 0.369 | 0.387 | 1   | 1   | 1    | 0.421 |
| CS2 | 0.545 | 0.685 | 1   | 1   | 0.488 | 0.421 | 1   | 1   | 0.421 | 0.488 |
| CS3 | 0.387 | 0.369 | 1   | 1   | 0.369 | 0.39  | 1   | 1   | 1    | 0.369 |
| CS4 | 0.369 | 0.488 | 1   | 1   | 0.369 | 0.421 | 1   | 1   | 1    | 0.421 |
| CS5 | 0.488 | 0.369 | 1   | 1   | 0.369 | 0.421 | 1   | 1   | 1    | 0.421 |
| CS6 | 0.387 | 0.421 | 0.369 | 1   | 0.488 | 0.421 | 1   | 1   | 0.369 | 1    |
| CS7 | 0.421 | 0.488 | 1   | 1   | 1    | 0.369 | 0.421 | 0.369 | 0.421 |
| CS8 | 0.421 | 0.421 | 1   | 1   | 1    | 0.369 | 0.421 | 0.369 | 0.421 |

Table 2. Table 4 shows the values of these significances. The values shown in grey are calculated for the cloud services CS2 and CS4.

Similarity measurement adopting significance. After we have defined the two types of significance for each service, we can then create the similarity measure, \( \text{Sim}(CS_n, CS_3) \), which takes into account the significance previously defined. To estimate the significance as accurately as possible, we identify the significance of \( CS_n \) with respect to \( CS_3 \) as a linear combination of \( J_{n,y} \) and \( VD_{n,y} \), such that:

\[
\text{SIG}_{n,y} = x \times J_{n,y} + (1-x) \times VD_{n,y},
\]

where \( x \) is defined to determine how much our significance relies on experience usability and value distribution. If \( x = 0 \), we only extract the experience usability for conducting significance estimation, and if \( x = 1 \), we consider only the value distribution. Hence, the similarity measure can be written in standard form:

\[
\text{Sim}(CS_n, CS_3) = \text{SIG}(CS_n, CS_3) \times \text{Sim}(CS_3, CS_3)
\]

where \( \text{SIG}_{n,y} \) denotes the significance of \( CS_n \) with respect to \( CS_3 \), and \( \text{Sim}(CS_3, CS_3) \) denotes the PCC value between \( CS_3 \) and \( CS_3 \). Different from existing similarity measures, our approach employs not only numerical distance but also usage experience as well as value distribution to determine the similarity between different services. With the definition of similarity measure defined in Eq. (11), for every cloud service in customer-service matrix, we rank their neighbors and select the top-k most similar services to make missing value prediction. Following the top-k similar service defined in [26], we get:

\[
CS^k_n = \{CS_y | CS_y \in CS_n, \text{Sim}^k(CS_n, CS_y) > 0, y \neq n\}, \tag{12}
\]

where \( CS_n \) denotes the neighbor set of \( CS_n \) in customer-service matrix, and \( \text{Sim}^k(CS_n, CS_y) \) denotes the similarity between \( CS_n \) and \( CS_y \). For the customer-service matrix in Table 2, we set \( x \) to 0.8 to obtain the similarity measures between different services (see Table 5). The top 3 neighbors of each service are marked in grey areas as seen in each column.

Missing value prediction. With the exponential growth of cloud service on the Internet, service recommendation techniques like QoS-aware CF approaches have become increasingly important and popular [37]. Based on our similarity measure, we propose an enhanced item-based CF approach (named as JV-PCC) to reinforce the prediction performance. To predict the missing value \( \hat{q}_{mn} \) of service \( CS_n \) for customer \( u_m \), we first determine the objective weight of each similar neighbor:

Table 5. Similarities between different services.

|   | CS1 | CS2 | CS3 | CS4 | CS5 | CS6 | CS7 | CS8 | CS9 | CS10 |
|---|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|
| CS4 | 0.361 | 0.334 | 0.366 | -0.099 | -0.421 | -0.663 | -0.24 | 0.402 | 0.141 |
| CS2 | 0.323 | 0.403 | 0.484 | -0.379 | 0.376 | -0.153 | 0    | 0.213 | -0.081 |
| CS3 | 0.334 | 0.34  | 0.223 | 0.36  | 0.36  | 0.259 | 0.4  | -0.165 | -0.212 |
| CS4 | 0.314 | 0.387 | 0.223 | 0.36  | 0.36  | 0.303 | -0.494 | -0.347 | -0.166 | 0.046 |
| CS5 | -0.096 | 0.326 | 0.36  | 0.453 | 0.324 | -0.099 | -0.334 | -0.464 | -0.432 |
| CS6 | -0.435 | 0.376 | 0.36  | 0.408 | 0.39  | 0.145 | 0.145 | -0.312 | -0.208 | -0.045 |
| CS7 | -0.663 | 0.135 | 0.189 | -0.605 | -0.085 | 0.146 | 0.549 | 0.032 | -0.451 |
| CS8 | -0.24  | 0    | 0.4  | -0.43 | -0.334 | -0.312 | 0.549 | 0.032 | -0.451 |
| CS9 | 0.324 | 0.184 | -0.165 | -0.193 | -0.464 | -0.152 | 0.027 | 0.393 | -0.395 |
| CS10 | 0.14  | -0.079 | -0.268 | 0.04  | -0.432 | -0.405 | -0.544 | -0.334 | -0.432 |

doi:10.1371/journal.pone.0097762.t005
\[ \omega_{n,y} = \frac{\text{Sim}^2(c_{s_n}, c_{s_y})}{\sum_{q_{m,y} \neq \text{null}} \text{Sim}^2(c_{s_n}, c_{s_y})}, \]  

where \( c_{s_y} \in \text{CS}^k \) denotes a similar neighbor of \( c_{s_n} \) while \( \text{Sim}^2(c_{s_n}, c_{s_y}) \) denotes the similarity between \( c_{s_n} \) and \( c_{s_y} \). The objective weights define the relative importance of each similar neighbor in the CF process. Next, we attain a prediction by a classic aggregation function:

\[ \hat{q}_{m,n} = \sum_{q_{m,y} \neq \text{null}} \left( q_{m,y} \times \omega_{n,y} \right), \]

where \( q_{m,y} \) denotes the historical record of \( c_{s_y} \) made by customer \( u_m \). In practice, local runtime environment (e.g. network bandwidth) and customer’s rating style may significantly influence the historical records over delivered services. However, the above function deeds inappropriate as it is calculated through only one customer. To address this problem, JV-PCC predicts the missing attribute value by the following equation:

\[ \hat{q}_{m,n} = q_m^\text{\#} + \left( q_m^\text{\#} - q_m \right) \sum_{c_{s_y} \in \text{CS}^k_{\text{null}}} \omega_{n,y} \times \left( \frac{q_m - q_{m,y}}{q_y^\text{\#} - q_y} \right), \]

where \( \omega_{n,y} \) denotes the objective weight of \( c_{s_y} \) with respect to \( c_{s_n} \), while \( q_y^\text{\#} \) and \( q_y^\text{\#} \) denote the minimum and maximum values of service \( c_{s_y} \), respectively. Table 6 displays the values estimated for the missing records in Table 2. In the experimental examples, both customer-based and service-based neighborhood information were adopted for approximating the missing value.

### Table 6. Predicted attribute values.

| \( cs_1 \) | \( cs_2 \) | \( cs_3 \) | \( cs_4 \) | \( cs_5 \) | \( cs_6 \) | \( cs_7 \) | \( cs_8 \) | \( cs_9 \) | \( cs_{10} \) |
|---|---|---|---|---|---|---|---|---|---|
| \( u_1 \) | 0.843 | | | | | | | | |
| \( u_2 \) | | | | | | | | | |
| \( u_3 \) | 0.865 | | | | | | | | |
| \( u_4 \) | 0.85 | 0.8 | | | | | | | |
| \( u_5 \) | 0.831 | | | | | | | | |
| \( u_6 \) | 0.865 | | | | | | | | |

\[ H_{m,n} = \begin{cases} q_m - q_m^\text{\#}, & q_m \in [0, \text{MAX}(q_{m,n})] \\ q_m^\text{\#} - q_m, & q_m \in [\text{MIN}(q_{m,n}), 0] \end{cases}, \]

where \( q_m^\text{\#} \) and \( q_m^\text{\#} \) denote the minimum and maximum attribute values for customer \( u_m \) and they are subject to the following constraints:

\[ \{ q_m^\text{-} = \text{MIN}(q_{m,n} | n = 1, \ldots, N), \quad q_m^\text{\#} = \text{MAX}(q_{m,n} | n = 1, \ldots, N) \}. \]

The attribute utility \( H_{m,n} \) is in the range of \([0, 1]\), where a larger \( H_{m,n} \) indicates that customer \( u_m \) is more satisfied with the service \( c_{s_n} \).

#### Customer satisfaction estimation.

From influential theory in marketing science, we consider that the perception of cloud service trustworthiness is a customer satisfaction function, which includes customer preference and expectation attributes. In general, customer satisfaction function should exhibit two characteristics: (1) given the same expectation, a trustworthy cloud service is weighed much more heavily than an untrustworthy service. This effect is reflected in the derivation of attribute utility; (2) customer satisfaction slightly increases when attribute utility surpasses a certain value (expectation), and significantly decreases when attribute utility falls below expectation [39]. We formalize this interaction as a piecewise linear function:

\[ C_{m,n} = \begin{cases} H_{m,n}, & H_{m,n} \geq H^{\text{exp}}, \\ H_{m,n} - (H^{\text{exp}} + 1)^\delta, & H_{m,n} < H^{\text{exp}}, \end{cases} \]

where \( C_{m,n} \) is constrained to \( 0 \leq C_{m,n} \leq 1 \); the parameter \( \delta \) regulates the impact of customer preference on perceived trustworthiness; and \( H^{\text{exp}} \) denotes the customer expectation with regard to selecting trustworthy cloud service. As shown in Fig. 1, \( C_{m,n} \) is continuous (i.e. the piecewise function converges at \( H_{m,n} = H^{\text{exp}} \)).
The severity and rate of satisfaction (controlled by $H^{exp}$ and $\delta$, respectively) reflect different customer’s tolerance to untrustworthy candidates. Let $H^{exp} = 0.7$, and $\delta = 2$. Table 7 shows the customer satisfaction for each attribute value (historical record or predicted value), which corresponds to the original customer-service matrix satisfaction for each attribute value (historical record or predicted aggregating trustworthy attributes. Let $u_i \in [0, 1]$, the degree of trustworthiness (alias “trust value” [40]) of candidates. Let $\forall j = \{1, 2, \ldots, J\}$ reflect different customer’s tolerance to untrustworthy candidates.

**Decision support for personalized service selection.** Multi-attribute trustworthiness evaluation is an important step for making accurate service selection. We suppose that $u_n$ is the active customer, who requires trustworthy cloud service. While the evaluation results have arrived, a set of appropriate service candidates can be identified for $u_n$ by:

$$CS^m = \{ c_{i,m} | trust_n > \theta_m, n = 1, \ldots, N \},$$

where $trust_n$ denotes the trust value of $c_{i,m}$, $\theta_m$ denotes the selection threshold determined by $u_m$. We aim to remedy the shortcomings of evaluation-focused selection methods by avoiding the costly and time-consuming real-world service invocations. Note that when $CS^m = \emptyset$ the service selection for the active customer $u_m$ needs to be degrade by decreasing the parameter $\theta_m$.

Let $\varepsilon_1 = \ldots = \varepsilon_6 = 0.85$, a set of trustworthy cloud services can be recommended for $u_1 \ldots u_6$ as

$$CS^1 = \{c_{s_1}, c_{s_4}, c_{s_5}\}, \quad CS^2 = \{c_{s_8}, c_{s_9}\}, \quad CS^3 = \{c_{s_3}, c_{s_6}, c_{s_10}\}, \quad CS^4 = \{c_{s_2}, c_{s_4}, c_{s_5}, c_{s_6}\}, \quad CS^5 = \{c_{s_7}\}, \quad CS^6 = \{c_{s_7}, c_{s_8}\},$$

where the customer satisfaction for each attribute value is presented in Table 7. In practice, our approach makes it possible to deal with various types of trustworthiness-aware cloud service selections by combing the evaluation-focused and the prediction-focused methods. Note that if trustworthiness is not the only issue that affects customer’s decision making, it is necessary to extend the selection process of our approach, e.g., price-oriented service filtering, into other attributes or indexes.

### Results

In this section, abundance of experiments are conducted to show how to recommend trustworthy cloud service in the context

---

**Table 7.** Customer satisfaction.

| $u_1$ | $u_2$ | $u_3$ | $u_4$ | $u_5$ | $u_6$ |
|---|---|---|---|---|---|
| $c_1$ | 1 & 0.134 & 0.229 & 1 & 1 & 0.134 & 0 & 0.215 & 1 |
| $c_2$ | 0.32 & 0.134 & 0 & 0 & 0.134 & 0.134 & 0.466 & 1 & 1 & 0.134 |
| $c_3$ | 1 & 0.623 & 0.587 & 0.134 & 0 & 0 & 0.05 & 0.166 & 0.884 & 1 |
| $c_4$ | 0.32 & 1 & 0.32 & 1 & 1 & 1 & 0.196 & 0 & 0.196 & 0.531 |
| $c_5$ | 0 & 0 & 0.115 & 0.036 & 0.623 & 0.115 & 0.8 & 0.407 & 0 & 0.623 |
| $c_6$ | 0.825 & 0.623 & 1 & 0.134 & 0.645 & 0.83 & 1 & 0.926 & 0.8 & 0 |

---

Figure 1. Customer satisfaction function $C_{m,n}$ (a) and (b) depict the distributions of customer satisfaction as recorded at the fixed expectation $H^{exp} = 0.7$ and $H^{exp} = 0.9$, where the parameter $\delta$ is varied from 2 to 6 in increment of 2. It can be observed that the rate of change in customer satisfaction differs significantly when $H_{m,n}$ falls below and exceeds the expectation.

doi:10.1371/journal.pone.0097762.t001
of large sparse assessment dataset, and to verify the efficiency of our CF approach.

Prototype implementation and results

To demonstrate the effectiveness of the proposed service selection approach, we use Microsoft C# .NET to develop a prototype system. Based on literature [8, 41, 42], we find Availability and Performance are two commonly used trustworthy attributes. We utilize them to conduct trustworthiness-aware service selection, by including two types of historical records: response-time and throughput. Their evaluation styles and weights are summarized as Table 8.

We employ an open QoS research dataset [43] to simulate the historical records of Performance and Availability in cloud service market. The QoS values for response-time and throughput were collected from 339 users over 5825 web services in a real-world environment. Since it is impractical to discover and distinguish all functionally-equivalent services at the selection time, we randomly select 100 services’ QoS records, and construct two 339x6100 customer-service matrices for our experiment. Figure 2 shows the value distribution of response-time and throughput in user-service matrix. We cannot simply utilize these QoS records to analyze and rank the cloud services since these customer-service matrices are sparse assessment datasets, and cannot accurately interpret the trustworthiness status of all services. Suppose $u_{339}$ is the active customer. The historical records made by $u_{339}$ contains 9 and 7 missing values (on response-time and throughput, respectively) which will potentially affect his cloud service selection decision. Therefore, the proposed CF approach is employed to predict the missing attribute values. At this simulation experiment, the similarity parameter $\alpha$ is set to 0.8 and remains so until the trust values for $u_{339}$ are reported. Once the prototype system obtains the customer satisfactions by utilizing Eqs.(12)–(14), where the parameter $\delta = 2$ and the expectation $H^{exp} = 0.7$, the active customer will receive the trust values of each service. We vary the selection parameter $e_{339}$ from 0 to 1 in increment of 0.1, and count the cloud services whose trust values surpass $e_{339}$ (the number of recommended services for $u_{339}$, i.e. $|CS_{339}|$). The experiment results are shown in Figure 3. Although we only study two trustworthy attributes in the

Figure 2. QoS value distributions. (a) and (b) depict the value distributions of response-time and throughput in our customer-service matrices, where “−1” indicates that the service invocation failed due to an http error. The ranges of response-time and throughput are 0–16.053 seconds and 0–541.546 kbps, respectively.

Figure 3. The number of recommended services for $u_{339}$. Results are presented for the proposed cloud service selection approach, where the parameter $\varepsilon_{339}$ is varied from 0 to 1 in increment of 0.1.

Figure 4. Impact of preference and expectation. (a) and (b) depict the experimental results of preference parameter $\delta$ and expectation $H^{exp}$, respectively. They indicates that $\delta$ regulates the elimination rate of untrustworthy cloud services, whereas $H^{exp}$ controls the degree of customer’s tolerance to untrustworthy service.
experiment, the proposed approach can be easily extended to other trustworthiness-aware service selection problems. When selecting the optimal trustworthy services from a set of functionally-equivalent candidates, the entry data of our approach are the corresponding historical records (i.e., QoS values or customer ratings), the active customer’s preference and expectation towards service trustworthiness, and the selection parameter.

Impact of \( \delta \) and \( \exp \)

Different customers have different preference and expectation in trustworthy service selections. Instead of risk-neutral attribute utility, we use the customer satisfaction \( C_{m,n} \) to identify the perceived trustworthiness of delivered services. To evaluate the impact of customer’s preference and expectation, we have conduct additional experiments with variable parameters \( \delta \) and \( \exp \). In these experiments, we first vary \( \delta \) from 2 to 6 in increment of 2, where the expectation \( \exp \) is fixed at 0.7 first. Later, we set \( \delta \) to 2, and vary \( \exp \) from 0.7 to 0.9 in increment of 0.1. Figure 4 (a) shows the experimental results of preference parameter \( \delta \) and Figure 4 (b) shows the experimental results of expectation \( \exp \). The parameters \( \delta \) and \( \exp \) jointly determine how to derive the customer satisfaction from attribute utility to approximate the active customer’s attitude towards profit and risk.

Performance comparison of CF approaches

In this work, we present an enhanced item-based CF approach (i.e., JV-PCC) to predict the missing attribute values for cloud service selection. Our approach engages the significances \( J_{n,y} \) and \( VD_{n,y} \) to improve the accuracy of similarity estimation. To study the prediction performance, we compare JV-PCC with two existing item-based approaches: Item-based CF adopting PCC (IPCC) [44], and Extended PCC approach (f-PCC) [31].

Evaluation metric. We use Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) to evaluate the prediction performance.
performance of our approach in comparison with other approaches. MAE and RMSE are defined as:

\[
\begin{align*}
MAE &= \frac{1}{n} \sum_{m=1}^{n} |q_{m,n} - \hat{q}_{m,n}| \\
RMSE &= \sqrt{\frac{1}{n} \sum_{m=1}^{n} (q_{m,n} - \hat{q}_{m,n})^2}
\end{align*}
\]

where \(q_{m,n}\) and \(\hat{q}_{m,n}\) are the predicted QoS value and the actual value, respectively.

**Experimental setup and results.** The size of top-\(k\) similar service set plays an important role in CF approach, which determines how many neighbors’ historical records are employed to generate predictions. To study the impact of neighbor size \(k\), we separate the customer-service matrices into two parts: training set (80% historical records in the matrix) and test set (the remaining 20% records). We set the density to 50%, the training set (80% historical records in the matrix) and test set (the remaining 20% records). We set the density to 50%, the significance parameter \(\alpha\) to 0.7, and vary \(k\) from 5 to 30 in increment of 5. Figure 5 shows the experimental results for response-time and throughput. Under the same simulation condition, JV-PCC and f-PCC significantly outperform IPCC. The observations also suggest that better accuracy can be achieved by our model when more historical records are available in the service selection study.

**Conclusions**

Trustworthiness-aware service selection is a critical issue among cloud computing and service-oriented architecture communities.

**References**

1. Katzan H (2011) Cloud software service: concepts, technology, economics. Service Science 1(4): 256–269.
2. Stolyar AL (2013) An infinite server system with general packing constraints. Operations Research 61(5): 1209–1217.
3. Chen PY, Wu SY (2013) The impact and implications of on-demand services on market structure. Information Systems Research 24(3): 750–767.
4. Sunyaev A, Schneider S (2013) Cloud services certification. Communications of the ACM 56(2): 33–36.
5. Sujanika D, Annu YD (2011) Risk management and optimal pricing in online storage grids. Information Systems Research 22(4): 756–773.
6. Marston S, Li Z, Bandyopadhyay S, Zhang JH, Ghaisasi A (2011) Cloud computing – The business perspective. Decision Support Systems 51: 176–189.
7. Cusumano M (2010) Cloud computing and SaaS as new computing platforms. Communications of the ACM, 53(4): 27–29.
8. Alhamad M, Dillon T, Chang E (2011) A trust-evaluation metric for cloud computing. Decision Support Systems 51: 176–189.
9. Chen L, Xu XD, Huang ZC, Sun HL (2010) Region KNN: A scalable hybrid collaborative filtering algorithm for personalized web service recommendation. IEEE International Conference on Web Service (ICWS), pp.9–16.
10. Zheng ZB, Wu XM, Zhang YL, Lyu MR, Wang JM (2013) QoS ranking prediction for cloud services. IEEE Transactions on Parallel and Distributed Systems, 24(6): 1213–1222.
11. Pal R, Hui P (2013) Economic models for cloud service markets: Pricing and capacity planning. Theoretical Computer Science, 496(22): 113–124.
12. Sundareswaran S, Saequicairii A, Lin D (2012) A brokerage-based approach for cloud service selection. IEEE Fifth International Conference on Cloud Computing, pp.538–546.
13. Remrick P, LAouou N, Suchak M, Bergstrom P, Riedl J (1994) Groupsml: An open architecture for collaborative filtering of networks. Proceeding of ACM Conference on Computer Supported Cooperative Work, pp.175–186.
14. Breese JS, Heckerman D, Kadie C (1998) Empirical analysis of predictive algorithms for collaborative filtering. Proceedings of 14th Annual Conference on Uncertainty in Artificial Intelligence, 43–52.
15. Ding S, Ma XJ, Yang SL (2012) A software trustworthiness evaluation model using objective weight based evidential reasoning approach. Knowledge and Information Systems, 33(1): 171–189.
16. Jin R, Chai J, Si L (2005) An automatic weighting scheme for collaborative filtering. SIGIR, 337–344.
17. Bobadilla J, Hernandez A, Ortega F, Gutierrez A (2012) Collaborative filtering based on significances. Information Sciences 185: 1–17.
18. Seidmann H, Djassemi M (1991) The production data-based similarity coefficient versus Jaccard’s similarity coefficient. Computer & Industrial Engineering 21(1–4): 263–266.
31. Cheng HD, Chen JR (1997) Automatically determine the membership function based on the maximum entropy principle. Information Sciences 96(4): 163–182.
32. Shao LS, Zhang J, Wei Y, Zhao JF, Xie B, et al. (2007) Personalized QoS Prediction for Web Services via Collaborative Filtering. IEEE International Conference on Web Services (ICWS), pp.439–446.
33. Dózsa P, Basili V (2006) A practical framework for eliciting and modeling system dependability requirements: Experience from the NASA high dependability computing project. Journal of Systems and Software 79(1): 107–119.
34. Hwang SY, Lee CH (2013) Reliable web service selection in choreographed environments. Decision Support Systems54(3): 1463–1476.
35. Anderson EW, Sullivan MW (1993) The antecedents and consequences of customer satisfaction of firms. Marketing Science 12(2).
36. Karaoglou P, Basili V (2006) A practical framework for eliciting and modeling system dependability requirements: Experience from the NASA high dependability computing project. Journal of Systems and Software 79(1): 107–119.
37. Cruz JM, Liu ZG (2012) Modeling and analysis of the effects of QoS and reliability on pricing, profitability, and risk management in multiperiod grid-computing networks. Decision Support Systems 52(3): 562–576.
38. Rosario S, Benveniste A, Haar S, Jard C (2008) Probabilistic QoS and soft contracts for transaction-based web services orchestrations. IEEE Transactions on Services Computing 1(4): 187–200.
39. Zheng ZB, Zhang YL, Lyu MR (2010) Distributed QoS evaluation for real-world web services. Proceedings of the 8th International Conference on Web Services (ICWS), Miami, Florida, USA, pp. 83–90.
40. Deshpande M, Karypis G (2004) Item-based top-N recommendation. ACM Transactions on Information System 22(1): 143–177.
41. Benlian A, Hess T (2011) Opportunities and risks of software-as-a-service: Findings from a survey of IT executives. Decision Support Systems 52: 232–246.
42. Demirkan H, Cheng HK, Bandyopadhyay S (2010) Coordination strategies in a SaaS supply chain. Journal of Management Information Systems 26(4): 119–143.
43. Tsesmetzis D, Roussaki I, Sykas E (2008) QoS-aware service evaluation and selection. European Journal of Operational Research 191: 1101–1112.
44. Wang Q, Wang C, Ren K, Lou KJ, Li J (2011) Enabling public auditability and data dynamics for storage security in cloud computing. IEEE Transactions on Parallel and Distributed Systems 22(5): 847–859.
45. Iooup A, Ostermann S, Yigitbasi MN, Prodan R, Fahringer T, et al. (2011) Performance analysis of cloud computing services for many-tasks scientific computing. IEEE Transactions on Parallel and Distributed Systems 22(6): 931–945.