Future Generation Computer Systems 68 (2017) 275–288

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

Multi-valued collaborative QoS prediction for cloud service via time series analysis

Hua Ma [a,b,*], Haibin Zhu [b], Zhigang Hu [c], Wensheng Tang [a], Pingping Dong [a]

[a] Department of Computer, Hunan Normal University, Changsha 410081, China

[b] Collaborative Systems Laboratory, Nipissing University, North Bay, ON P1B8L7, Canada

[c] School of Software, Central South University, Changsha 410075, China

HIGHLIGHTS

- Propose a collaborative QoS prediction approach via time series analysis.
- Use cloud model theory to model time series feature of multi-valued QoS evaluations.
- Propose a vector comparison method to measure similarity between QoS cloud models.
- Employ the FAHP method to identify the weight of every time period objectively.
- Verify the effectiveness of the proposed approach based on a real-world datasets.

ARTICLE INFO

Article history:
Received 5 June 2016
Received in revised form
10 September 2016
Accepted 8 October 2016
Available online 18 October 2016

Keywords:
Cloud model
Collaborative QoS prediction
Multi-valued evaluations
Similarity measurement
Time series analysis

ABSTRACT

Aiming at the diversity of user features, the uncertainty and the variation characteristics of quality of service (QoS), by exploiting the continuous monitoring data of cloud services, this paper proposes a multi-valued collaborative approach to predict the unknown QoS values via time series analysis for potential users. In this approach, the multi-valued QoS evaluations consisting of single-value data and time series data from consumers are transformed into cloud models, and the differences between potential users and other consumers in every period are measured based on these cloud models. Against the deficiency of existing methods of similarity measurement between cloud models, this paper presents a new vector comparison method combining the orientation similarity and dimension similarity to improve the precision of similarity calculation. The fuzzy analytic hierarchy process method is used to help potential users determine the objective weight of every period, and the neighboring users are selected for the potential user according to their comprehensive similarities of QoS evaluations in multiple periods. By incorporating the multi-valued QoS evaluations with the objective weights among multiple periods, the predicted results can remain consistent with the periodic variations of QoS. Finally, the experiments based on a real-world dataset demonstrate that this approach can provide high accuracy of collaborative QoS prediction for multi-valued evaluations in the cloud computing paradigm.

1. Introduction

1.1. Motivation

In recent years, the cloud computing has been gaining enormous momentum. The cloud service providers (CSPs) around the world have publicized many services by encapsulating the various software applications, computing power and storage capacity [1]. With the increasing presence of cloud services, the accurate quality of service (QoS) data is required for cloud service selection approaches [2,3] and cloud services composition approaches [4,5] to work well. The QoS consists of both user-independent attributes and user-dependent attributes [6]. The user-independent QoS attributes, such as price and popularity, can be measured at the server-side and have identical values for different cloud service consumers (CSCs), while user-dependent QoS attributes, such as response time and throughput, can be measured at the client-side. How to obtain the sufficient and

* Corresponding author at: Department of Computer, Hunan Normal University, Changsha 410081, China.

E-mail addresses: huama@hunnu.edu.cn (H. Ma), haibinz@nipissingu.ca (H.B. Zhu), zghu@csu.edu.cn (Z.G. Hu).

http://dx.doi.org/10.1016/j.future.2016.10.012
0167-739X/

Table 1
An example of user-service matrices for response time.

Users	Cloud services			
	s_1	s_2	s_3	s_4
u_1	1.53	2.35	1.37	2.20
u_2	1.15	2.28		2.91
u_3	0.95	1.28		2.15
u_4	0.86	0.88		4.38
u_5		0.90	3.95	4.60

accurate user-dependent QoS attributes data has been the critical issue for selecting the optimal cloud service for potential users from a large number of candidates. This task includes the following challenges:

(1) The performance of a cloud service observed from the CSCs' perspective is usually very different from that declared by CSPs in service level agreement (SLA). The differences mainly due to the following reasons [7–9]:

(a) The QoS performance of cloud services is highly related to the invocation time, since the service status, such as the workload and the number of clients, and the network environment, such as congestion, change over time. Taking the real-world WS-DREAM dataset #3 for example, we have analyzed the QoS evaluations of 4532 services in 64 time intervals, and demonstrated the fact that one cloud service may have the quite different QoS performance in different periods [10].

(b) The CSCs are typically distributed in different geographical locations or network locations. The CSCs-observed QoS performance of cloud services is greatly influenced by the Internet connections between CSCs and cloud services. In the previous research [11], we proposed a user feature model to systematically analyze the influence of both geographical locations and network locations for 339 users and 5825 services in experiments based on the real-world WS-DREAM dataset #2. The results demonstrated that the different CSCs might observe quite different QoS performance even if they invoke the same service.

The SLA contains the service-level objectives (SLOs) and various QoS objectives, which the service must fulfill. Considering the cases of violations, SLA often defines the monetary penalties and prompts CSPs to reduce the number of SLA violations for their services. Thereby, the prior identification of SLA violations has become a very important research topic [2,12], which focuses on predicting the SLA violations by comparing the prediction values of SLOs with the existing customer SLAs. The SLO is usually composed of one or more QoS measurements that are combined to produce the SLO values [8]. Thus, the approach for predicting the unknown QoS values with high accuracy for potential users is really important for the identification of SLA violations.

(2) In reality, the long-term QoS guarantees from a CSP may not be always available [4]. For example, in Amazon EC2, only the "availability" attribute of QoS is advertised for a long-term guarantee [13]. Obviously, detailed QoS information facilitates potential users to make the sound and timely decisions when selecting the optimal cloud service from a large number of candidates. Hence, on the basis of the limited historical QoS data, predicting other unknown QoS values for potential users is always a valuable research area.

(3) A user usually only invokes a small number of cloud services in the past and thus only observes the QoS values of these invoked cloud services. Without sufficient QoS information, it is difficult for potential users to select the optimal service from the candidates. To obtain accurate QoS values about more user-dependent attributes for different users, the client-side QoS evaluations are usually needed. However, invoking all of the cloud services from the CSCs' perspectives for the evaluation purpose is quite difficult and includes the following critical drawbacks [7,14]:

(a) The invocations of services may be too expensive for CSCs because CSPs may charge for these invocations.

(b) It is time-consuming to evaluate all the services if there are a large number of candidate services.

To address the above problems, an efficient way is via a collaborative filtering algorithm (CFA) by employing the historical QoS data [6,8,15,16]. In CFA, the user-service QoS matrix [5,14,17,18] is used as the fundamental data source. An example of user-service matrices for response time is shown in Table 1, in which each column represents a service and each entry is a historical response time data invoked by a user on the specified service. Considering that each user can only invoke limited cloud services, such a matrix is filled with numerous unknown entries and thus the key is how to accurately predict them based on the known ones [18,19].

Over the past few years, security problems have increasingly emerged in cloud services. In order to facilitate users to select the highly trustworthy cloud services, continuous monitoring of QoS for cloud services has been an urgent need [10]. Currently, some organizations [20–22] have carried out work on continuous monitoring of cloud services and it becomes possible to thoroughly analyze the QoS of cloud services based on time series data. For example, Cloud Security Alliance (CSA) launched the Security, Trust, and Assurance Registry (STAR) Program [20]; Yunzhiliang.net [21] released the assessment reports for popular cloud services deployed in China. China Cloud Computing Promotion and Policy Forum (3CPP) published the trusted services authentication standards and the evaluation result for trusted services [22]. Besides, Zheng et al. explored the Planet-lab project to collect real-world QoS evaluations from 142 users on 4532 services over 64 timeslots [7,14,23]. Additional studies [24,25] have demonstrated that the agent software deployed in the CSCs' terminal devices can easily capture real-time monitoring data. In contrast to the discrete QoS data observed in a single timeslot, the time series QoS data produced by continuous monitoring is more likely to help potential users, especially those who pay attention to a service's performance during specified periods, to investigate the QoS of candidate services from a comprehensive perspective.

However, continuous monitoring inevitably degrades the client's performance and creates hidden security threats. Please note that, by "clients" we mean the computers operated by the users. CSCs tend to prohibit agents from running for extended periods in clients. Instead, CSCs prefer to actively submit the QoS evaluations at discrete time points or specified periods to a trustworthy third-party platform. Thus, in practical applications, the QoS evaluation data is usually multi-valued, consisting of the time series data and single-valued data. Existing studies mainly involving the single-valued QoS data cannot directly support the users' decision-making based on multi-valued QoS evaluations.

Meanwhile, the conventional studies [10,26] have revealed that cloud services' QoS has the apparent characteristic of periodic variation. Cloud services perform the best in idle hours and performance become worse during busy hours. Furthermore, CSCs have different requirements for the performance of services in different periods. For example, a security company that pays particular attention to the busy times for buying and selling stocks, such as the periods from 9:30 a.m. to 11:30 a.m. and from 1:00 p.m. to 3:00 p.m., expects a cloud storage service to provide high performance in concurrent reading and writing during these two periods. The cloud storage service with enough concurrent reading capacity in other periods is satisfactory because the security company provides only query services for stock information. In contrast, for a logistics company, statistical data indicates that the peak times for querying express packages occur between 12:00 a.m. and 2:00 p.m. and between 6:00 p.m. and 8:00 p.m. Therefore, this company needs cloud storage services that demonstrate superior concurrent reading performance during these two periods compared to other periods. Obviously, considering the periodic variations in QoS and the CSCs' application requirements during different periods can improve the accuracy of QoS prediction.

1.2. Our contributions

To address this challenge, we propose a multi-valued collaborative approach for the time-aware QoS prediction of cloud services. The basic idea is that the QoS values of user-dependent attributes for one user can be predicted via time series analysis by utilizing the past usage experiences of other users. Building on the time series feature analysis of QoS data, this paper introduces the cloud model theory to describe the changeable characteristics of QoS in every period. The multi-valued data of QoS evaluations from CSCs are transformed into cloud models, and the differences between a potential user and other CSCs in different periods are analyzed based on these cloud models. In order to accurately identify the neighboring users for a potential user, we propose a new vector comparison method to calculate the similarity between cloud models and employ the fuzzy analytic hierarchy process (FAHP) method to determine the objective weights of different periods. By incorporating the multi-valued QoS evaluations from neighboring users with the objective weights among multiple periods, the unknown QoS values of a cloud service can be predicted for the potential user.

The main contributions of this paper are as follows:

(1) Aiming at the time series feature of QoS and the multi-valued feature of QoS evaluations, we propose a collaborative QoS prediction approach based on cloud model theory and time series analysis, in which the multi-valued QoS data is preprocessed and modeled as cloud models in multiple periods. In order to identify exactly the neighboring users for a potential user, we measure the similarity between cloud models in every period separately and employ FAHP method to determine the objective weights of periods according to the application requirements of potential users. The predicted results can remain consistent with the periodic variations of QoS in multiple periods with high prediction precision.

(2) Against the deficiency of existing methods of similarity measurement between cloud models, we propose a novel vector comparison method combining the orientation similarity and dimension similarity to improve the precision of similarity calculation. This method supports the different measurement scales of three numerical characteristics in cloud models. Numerical examples demonstrate that this method can yield a more accurate measurement of the similarity between cloud models than the existing methods.

(3) We examine the proposed approach through experiments using a real-world dataset. Results demonstrate that this approach can provide high accuracy of multi-valued collaborative QoS prediction in the dynamic and vulnerable cloud environment, and can contribute to selecting the suitable cloud services with consideration of both the periodic variations of QoS and the application requirements of potential users in the multiple periods.

The rest of this paper is organized as follows. Section 2 introduces the related work. Section 3 is the problem statement. Section 4 gives the time series analysis of QoS based on cloud models. Section 5 proposes the scheme of collaborative QoS prediction. Section 6 presents a new method to measure the similarity between cloud models. Section 7 puts forward the procedure of multi-valued collaborative QoS prediction. Section 8 analyzes the experiments and results. Finally, Section 8 presents conclusions.

2. Related works

On the whole, QoS prediction is still a new direction in cloud computing [27], there are some literatures on the QoS prediction approaches, which are valuable to estimate the unknown QoS values of cloud service for potential users based on the multi-valued evaluations. In this section, we categorize the related works into three groups as follows:

2.1. Collaborative QoS prediction based on neighboring users

CFA is a classic prediction method of employing collective intelligence, which can extract the users' preferences and their behavior characteristics from the historical evaluation data provided by consumers. Inspired by the successes of CFA achieved by recommender systems, many studies have utilized collaborative method based on neighboring users to predict the unknown QoS values for cloud services [19,28–30]. These approaches generally consist of two steps: (a) finding similar users or services and mining their similarities; and (b) calculating unknown QoS values according to the known data of similar users or services.

Zheng et al. [28] employed CFA to predict the reliability of services by computing the user similarity and item similarity based on the Pearson correlation coefficient (PCC). Chen et al. [29] used a region model to study personalized service recommendation based on the hybrid CFA. Considering that the QoS evaluations of cloud services are related to not only objective measurements but also subjective perceptions of consumers, Ding et al. [19] employed the collaborative filtering recommendation technology and utility theory to predict the unknown QoS value based on the usage experiences of other similar services. In order to improve the prediction accuracy of CFA, Hu et al. [30] proposed a time-aware CFA to predict the unknown QoS values, in which the users' historic data about services at different time intervals is collected for calculating the similarity between services or that between users.

However, the recommender systems mainly process subjective data, while the QoS data related to cloud services is objective. Ma et al. [6] found that some significant differences between subjective data and objective data may bring errors to the prediction of unknown QoS values with traditional CFA. Two users may observe quite different QoS values on their commonly invoked services even they have a high similarity based on PCC. Moreover, the normalization methods cannot be used for cloud service QoS data. It is difficult to determine the QoS values scope of cloud services since the scope of QoS value often changes.

2.2. Collaborative QoS prediction based on matrix factorization

Matrix factorization (MF) has been employed for QoS prediction in recent years [14,15,31–34]. The collaborative QoS prediction based on MF aims to learn the latent factors that decide the QoS values. Usually, these approaches firstly identify the users' interests characteristic from the training dataset and the attribute characteristic of cloud services, and then predict the unknown values with the two characteristics.

Zhong et al. [31] proposed a time-aware personalized QoS prediction approach which analyzed the latent features of users, services and times by performing tensor factorization. Aiming at the important impact of context information for the QoS of cloud services, Xu et al. [32] proposed two matrix factorization-based QoS prediction models by employing both the geographical information from user side and the affiliation information from service side. Lo et al. [33] presented a collaborative framework for predicting QoS with location-based regularization, in which the latitude and longitude information of users are employed to identify the neighboring users. Yin et al. [34] proposed three service neighborhood prediction models based on probabilistic matrix factorization. These models learn the predicted value by utilizing the feature vectors of both the active service and its

Table 2
An example of user-service sub-matrices related to timeslot #1.

Users	Cloud services						
	s_1	s_2	s_3	s_4	s_5	s_6	s_7
u_1	1.53	2.35	1.37	2.20	3.05	0.85	4.50
u_2	1.15	2.28		2.91			1.57
u_3	0.95	1.28		2.15		1.41	2.28
u_4	0.36	0.88		4.38			3.05
u_5		0.90	3.95	4.60		0.55	1.70
u_6		1.67			2.51	1.46	
u_7	1.60		4.03		1.21	0.48	1.62

neighbors, and integrating with each neighbor's weight computed from the similarity.

Although the MF-based approaches achieve higher prediction accuracy than CFA-based approaches in some cases, there are some drawbacks due to the utilization of QoS values in similarity computation [32]: (1) The uncertainty of QoS in the dynamic and vulnerable cloud environment lowers the credibility of the similarity results for cloud services; (2) the similarity is likely to be inaccurate when the QoS evaluations are sparse.

2.3. Other collaborative QoS prediction approaches

Recently, some other methods, such as neural networks [17,35] and evidence theory [11], are introduced into the QoS prediction research.

Aiming at the nonlinear and dynamic property of QoS data, Luo et al. [17] extracted the fuzzy rules from QoS data by constructing fuzzy neural networks (FNN) as well as the action network, and proposed a QoS prediction approach through the fusion of FNN and adaptive dynamic programming. Kumar et al. [35] presented an artificial neural network (ANN) model with Bayesian-regularization to predict the unknown QoS values based on the past QoS performance data. To improve the precision of predicting the QoS for cloud services, Ma et al. [11] put forward an evidence theory-based approach combining the user features similarity calculation and incremental refinement iteration to filter the unreliable QoS evaluations.

For the above approached based on FNN and evidence theory, these abnormal data need be filtered out from the training dataset for the purpose of saving the training time or improving the calculation precision. However, in a dynamic cloud environment, the abnormal data itself probably becomes the important feature information. For example, both user #1 and user #2 experienced an abnormal QoS value in a specified period when invoking the same service, which means that they maybe distribute in the similar geographical locations or network locations. In that case, the QoS evaluations from user #1 have a higher value for user #2 than other users to improve the QoS prediction quality.

It has been a novel idea to predict the QoS of cloud service based on time series analysis [4,8,10]. With the consideration of correlations among the QoS attributes, Ye et al. [4] proposed a prediction model based on multivariate QoS analysis to predict the long-term QoS provisions according to the service providers' past QoS data and short-term advertisements. To overcome the shortcomings of both memory-based methods and model-based methods, Yu et al. [8] proposed a time-aware and location-aware CFA to predict the QoS values. In their method, the average similarity between target services at every time interval is calculated, and these services similar to the target service are selected for calculating the users' average similarity. In order to support the tradeoffs between performance-costs and potential risks, Ma et al. [10] proposed a trustworthy service selection approach via time series analysis, and the problem of time-aware trustworthy service selection is formulated as a multi-criterion decision-making problem of creating a ranked services list, solved by developing a ranking method based on interval neutrosophic set theory.

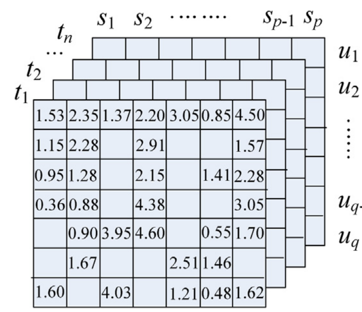


Fig. 1. A user-service-time QoS matrix for response time.

To the best of our knowledge, no similar research has investigated the collaborative QoS prediction approach via time series analysis based on the historical QoS data from users or inquired into employing the cloud model theory to analyze the multi-valued evaluations containing the abnormal QoS data in multiple periods from the perspective of similarity measurement.

3. Problem statement

Since many cloud services can provide the similar functions, consumers are not possible to use every service. Therefore, the QoS data of unused services plays an important role in providing suitable services to potential users. Just like the example of user-service QoS matrices shown in Table 1, each entry in it represents the response time data of a cloud service invoked by a user. A blank entry means that the user has not invoked this specified service. For the purpose of service selection or service composition, the potential users need to know all of the QoS data of candidate services related to them. However, the real-world QoS data is similar to this example in Table 1. Thus, it is significant to predict the blank entries before QoS-based service selection or service composition. The researchers have proposed some collaborative approaches [8,17,19,28–35] to predict the unknown QoS values.

Nevertheless, a series of security incidents emerged in cloud services have raised great concerns about the consistent capacity of service provision. In order to facilitate users to select the highly trustworthy cloud services, the continuous monitoring of QoS for cloud services has been an urgent need currently [20–22,24,25]. Some organizations have carried out work on the continuous monitoring of cloud services. Compared with the user-service QoS matrix in Table 1 used by the existing QoS prediction approaches [5,14,17,18], the continuous monitoring of cloud services can produce the time series QoS data, presenting a user-service-time QoS matrix shown in Fig. 1, and facilitates the potential users, especially those who pay attention to a service's performance during specified periods, to investigate QoS from a comprehensive perspective.

According to Fig. 1, this user-service-time matrix records the response time data of p services invoked by q users in n timeslots. From the perspective of a timeslot, we can get a user-service sub-matrix just like Table 1 for every timeslot. Some blank entries may still exist in the user-service sub-matrix. For example, the user-service sub-matrix related to timeslot t_1 is shown in Table 2.

From the perspective of a service, we can similarly get a user-timeslot sub-matrix for every service. In the user-timeslot sub-matrix, the QoS data of one user may compose a time series if there are no blank entries related to this user. However, providing the complete time series QoS data for every user is still an illusion due to the time-consuming and costly cloud service invocation.

Table 3
An example of user-timeslot QoS sub-matrices for multi-valued response time data.

Users	Timeslots									
	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}
u_1	1.53	0.28	0.21	0.29	0.25	0.42	0.31	1.24	0.25	1.38
u_2	1.15	0.27	0.44		0.64	0.67	0.27	0.88	1.22	8.29
u_3	0.95		0.35	1.01	0.66	0.83	0.24	0.95		8.20
u_4	0.36	0.24		0.23	0.40	0.23	0.25		0.83	1.57
u_5		0.33					0.20			
u_6					0.53					
u_7	1.60			0.87				1.28		

Therefore, the real user-timeslot QoS sub-matrix related to service s_1 is usually composed of the multi-valued data shown in Table 3.

To explain the multi-valued QoS data easily, we present three definitions as follows:

Definition 1 (*Single-Valued QoS Evaluations*)**.** This refers to the real QoS data measured by one user only in a timeslot.

For example, in Table 3, u_6 only observed the QoS of the service in $t5$, and thus the QoS data provided by u_6 is single-valued.

Definition 2 (*Time Series QoS Evaluations*)**.** This refers to the real QoS data measured by one user in at least 2 timeslots. Time series QoS data can be further subdivided into two categories, namely the complete time series data and incomplete time series data. If a user observed the QoS of one service in every timeslot, the QoS data measured by this user is a complete time series data. Namely there is no unknown QoS value of the service for this user.

For example, in Table 3, u_1 observed the QoS of the service in all of the timeslots. Assume that the total number of timeslots is n. The QoS data measured by this user is an incomplete time series data when the length of time series QoS data is less than n and greater than 1.

Definition 3 (*Multi-Valued QoS Evaluations*)**.** This refers to the real QoS data of candidate services observed by multiple users in multiple timeslots, composed of the single-value data and the time series data.

The existing approaches [4,7,8,30] mainly studied the time-aware QoS prediction from the perspective of user-service QoS sub-matrix, while this paper will start by the time series analysis of user-timeslot QoS sub-matrix, and aggregate the information associated to both user-service correlations and user-timeslot correlations. In the previous research [10], on the premise that the complete time series data is available, we have studied the trustworthy service selection approach via the time series analysis from the perspective of user-timeslot QoS sub-matrix. Considering that the difficulties of acquiring the complete time series data in a real-world cloud environment, as a result, the problem we study in this paper is how to precisely predict the unknown QoS values for potential users in accordance with the known multi-valued QoS data. In this paper, the abnormal data, such as the last column of Table 3, will be not filtered out directly, and become the significant feature information for accurately identifying the neighboring users. We will introduce cloud model theory to analyze the time series feature of multi-valued QoS data; these abnormal data can contribute to generating a more exact cloud model by the entropy and hyper entropy, which conforms to the real QoS situations of cloud services.

4. Time series analysis of QoS based on cloud models

Aiming at overcoming the deficiencies of existing methods in solving the uncertainty of information, Li et al. proposed the cloud model theory [36] combining the probability theory and fuzzy theory in 1995. Recently, cloud models have been applied successfully in many fields, such as knowledge discovery [37], trust evaluation [38] and decision analysis [39]. The cloud model theory can also provide strong support for analyzing the latent features hidden in time series data [40]. If the time series data is divided into multiple subsequences, the local feature of every subsequence can be described via a cloud model. Then, a set of cloud models corresponding to multiple subsequences facilitates depicting the global feature and local features of the time series data [41,42].

Considering the uncertainty of cloud environments, this paper introduces the cloud model theory to analyze the time series feature of QoS for cloud services. A QoS cloud model is composed of three numerical characteristics, namely Ex (expectation), En (entropy) and He (hyper entropy), defined as $cm = \{Ex, En, He\}$. Ex is the most representative value of QoS, En represents the granularity scale of QoS, and He depicts the uncertainty of the QoS granularity. From the viewpoint of fuzzy set, Ex is the expected value of QoS with membership degree 1, En represents the uncertainty of QoS values, which can be used to calculate the membership degree, and He depicts the uncertainty of membership degree. Cloud models make it possible to get the distributing range of a qualitative QoS based on the continuous monitoring data.

A QoS cloud consists of many cloud drops. The QoS evaluations from consumers can be viewed as cloud drops and sent to a reverse cloud generator (RCG). Assuming that there is a time series QoS data consisting of N timeslots, $Ei = \{e_{i,1}, e_{i,2}, \ldots, e_{i,N}\}$, which is provided by consumer #i, the numerical characteristics of consumer #i's QoS cloud model can be obtained by Eq. (1) [43]:

$$\begin{cases} Ex_i = \dfrac{1}{N}\displaystyle\sum_{j=1}^{N} e_{i,j} \\ En_i = \sqrt{\dfrac{\pi}{2}} \times \sigma = \sqrt{\dfrac{\pi}{2}} \times \dfrac{1}{N} \times \displaystyle\sum_{j=1}^{N} \left| e_{i,j} - Ex_i \right| \\ He_i = \sqrt{\left|S^2 - En_i^2\right|} = \sqrt{\left|\dfrac{1}{N-1}\displaystyle\sum_{j=1}^{N}(e_{i,j} - Ex_i)^2 - En_i^2\right|} \end{cases}, \quad (1)$$

where $e_{i,j}$ represents the QoS evaluation value from consumer #i in timeslot #j, Ex_i is the mean value of QoS evaluations, σ is the standard deviation of Ex_i, and S^2 is the sample variance of Ex_i.

We analyzed the QoS data of real services using the WS-DREAM dataset #3 [7,23], which collected the real-world QoS evaluations about the response time and throughput of 4532 services provided by 142 users in 64 timeslots based on PlanetLab. These datasets have frequently been applied in researches concerned with cloud computing [14,19]. We divide the response time data from timeslot #1 to #60 into six periods and therefore six subsequences, and create cloud models corresponding to these periods. Table 4 shows these cloud models that reflect the response time data of service #601 and #609 as provided by user #9.

In Table 4, every period consists of ten timeslots. The set of QoS cloud models is noted as $CM = \{cm_1, cm_2, \ldots, cm_6\}$, and the set of periods is noted as $TP = \{tp_1, tp_2, \ldots, tp_6\}$.

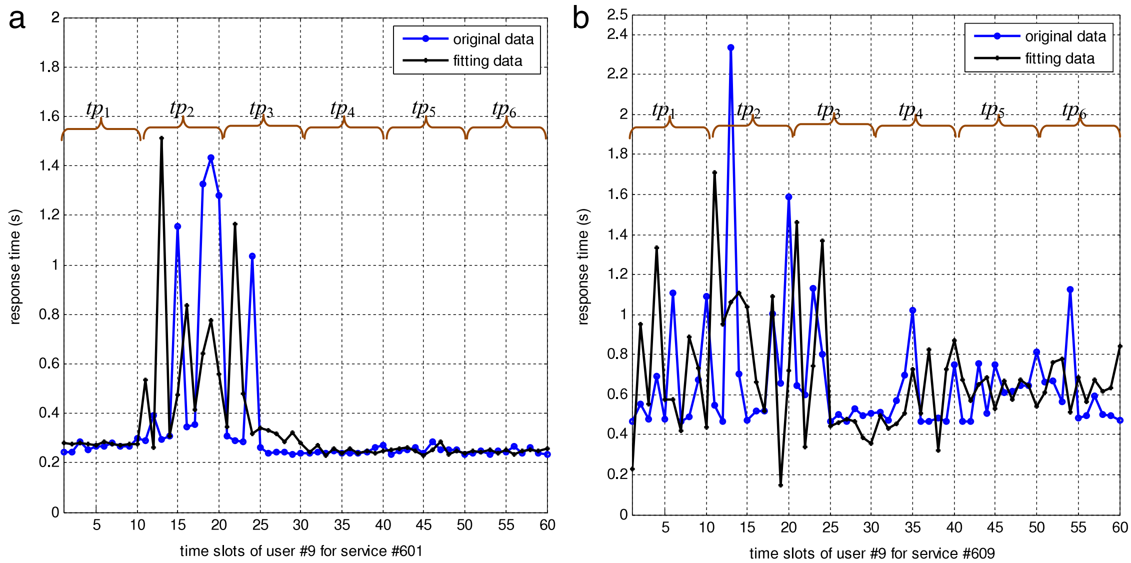


Fig. 2. Comparisons between fitting data and original data. (a) service #601; (b) service #609.

Table 4
Cloud models of response time for service #601 and service #609.

Periods	Cloud models for service #601			Cloud models for service #609		
	Ex	*En*	*He*	*Ex*	*En*	*He*
tp_1	0.2672	0.0163	0.0076	0.6494	0.2422	0.0683
tp_2	0.7173	0.5822	0.2896	0.8809	0.5726	0.2288
tp_3	0.3379	0.1750	0.1738	0.6139	0.1846	0.0986
tp_4	0.2457	0.0108	0.0049	0.5891	0.1749	0.0533
tp_5	0.2514	0.0120	0.0077	0.6278	0.1186	0.0270
tp_6	0.2461	0.0107	0.0035	0.6050	0.1604	0.1137

To further illustrate the effectiveness of the QoS cloud model in describing time series data, we compared the fitting data of cloud drops obtained by the forward cloud generator with the original data. The forward cloud generator produces ten cloud drops for every cloud model, and these cloud drops are arranged into a new sequence according to the order in which they were created. Fig. 2 demonstrates that the sixty cloud drops generated by the cloud models precisely reflect the variation characteristics of the original data in six periods.

Fig. 2 suggests that the cloud model theory is an effective tool for analyzing the time series features of QoS. Although the mean response time of service #601 and service #609 is similar, they have distinctly different performances in six periods. According to Fig. 2, the response speed of service #601 is much faster than service #609; the response time of service #601 is quite steady in most of the periods apart from tp_2 and tp_3, and the response time of service #609 varies greatly in all periods. Therefore, analyzing the time series features of QoS facilitates evaluating accurately the differences between cloud services.

5. Collaborative QoS prediction scheme

Assuming that $U = \{u_1, u_2, \ldots, u_q\}$ is the users set, u^p represents the potential user, $S = \{s_1, s_2, \ldots, s_p\}$ is the cloud services set, $TP = \{tp_1, tp_2, \ldots, tp_d\}$ is the periods set, and $T = \{t_1, t_2, \ldots, t_{r\times d}\}$ is the timeslots set, the problem of multi-valued collaborative QoS prediction for potential users via time series analysis can be formulated as shown in Fig. 3.

According to Fig. 3, the continuous monitoring data of QoS, namely the multi-valued QoS evaluations, can be provided by cloud service consumers who could be individual users or correlative organizations, such as CSA and 3CPP. The multi-valued QoS evaluations can be abstracted into a user-service-time matrix just like Fig. 1.

To facilitate a potential user to select an optimal service for service selection problem or service composition problem, the unknown QoS data, namely the blank entries of the user-service-time matrix, should be firstly estimated by employing the QoS prediction approach. Therefore, exactly predicting the unknown QoS value of s_k in t_i for potential users based on this user-service-time matrix is the core goal of this problem. In this problem, the input data includes the ID of u^p, the ID of s_k, the ID of t_i, the multi-valued QoS evaluation data (M) from consumers which consist of time series data and single-valued data, and the sensitivity scores for periods (B) provided by u^p. Potential users can give the sensitivity scores for every period by employing the five-level complementary scales of the FAHP method, and the scores should reflect the application requirements for different periods.

After the potential user provides the input data, the subsequent process via time series analysis ensures that the unknown QoS values are predicted accurately, which can be generalized into six steps as follows:

(1) Preprocessing the multi-valued QoS evaluation data. These data will be transformed into the cloud model matrix corresponding to multiple periods.
(2) Calculating the user similarity based on the training data by measuring the similarity between cloud models in different periods.
(3) Determining the weights of different periods by employing the FAHP method.
(4) Aggregating the cloud models with the weights in multiple periods.
(5) Identifying the neighboring users from cloud service consumers for the potential user according to the similarity threshold. The QoS cloud models of neighboring users should be highly similar to the QoS cloud models of potential users in multiple periods.
(6) Predicting the QoS for potential users according to the similarity between the cloud models.

In order to solve this problem, we must accurately measure the differences between cloud service consumers based on the QoS cloud models in multiple periods. In the following sections, we will discuss the similarity calculation methods of cloud models and the procedure of multi-valued collaborative QoS prediction.

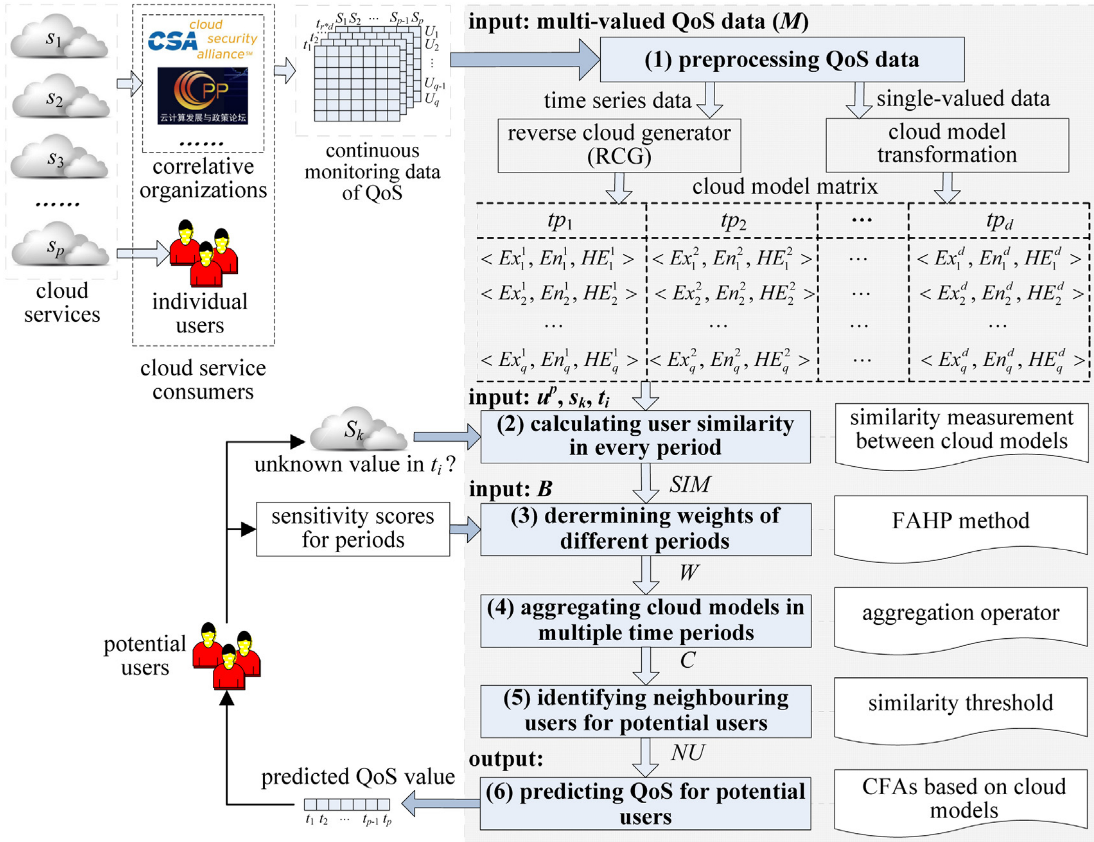


Fig. 3. Collaborative QoS prediction scheme.

6. Similarity measurement between cloud models

6.1. Existing methods

The existing methods for measuring similarities between cloud models mainly include:

(1) SCM [44]. This method measures the similarity between two cloud models by calculating the distances between sampling cloud drops. Assume that n_i is the number of cloud drops from cloud model cm_i and n_j is the number of cloud drops from cloud model cm_j. These cloud drops should satisfy $(Ex-3En) \le x \le (Ex+3En)$, assuming $n_i \le n_j$. $C_{n_j}^{n_i}$ represents the combinatorial number of pairwise comparisons in which any n_i sampling cloud drops are chosen from n_j cloud drops of cm_j, and they are compared with any n_i sampling cloud drops of cm_i. Then, the similarity between cm_i and cm_j is calculated by Eq. (2):

$$s(cm_i, cm_j) = \left(\frac{1}{C_n^m} \times \sum (drop(i) - drop(j))^2\right)^{1/2} \Big/ m, \tag{2}$$

where $drop(i)$ and $drop(j)$ represent the cloud drops from cm_i and cm_j, respectively. Obviously, The smaller $s\ (cm_i, cm_j)$ is, the more similar the two cloud models are. Considering that $C_{n_j}^{n_i}$ requires extensive calculations, a great deal of time is spent in calculating the similarity between two cloud models. Moreover, the calculation accuracy is limited and difficult to improve. Therefore, the SCM method is not widely applied in practice.

(2) LICM [45]. This method employs the cosine distance to measure the similarity between cm_i and cm_j by Eq. (3):

$$s(cm_i, cm_j) = \vec{V_i} \bullet \vec{V_j} / \left\| \vec{V_i} \right\| \left\| \vec{V_j} \right\|, \tag{3}$$

where $\vec{E_i} = (e_{i,1}, e_{i,2}, e_{i,3}) = (Ex_i, En_i, He_i)$; $\vec{E_j} = (e_{j,1}, e_{j,2}, e_{j,3}) = (Ex_j, En_j, He_j)$. The greater the cosine distance is, the more similar the two cloud models are. In a cloud model, Ex is usually much larger than En and He; however, the cosine distance treats the three numerical characteristics of a cloud model equally although they have different measurement scales. Therefore, it produces results that apparently different cloud models may be very similar.

(3) ECM and MCM [46]. ECM uses expectation curves to reflect the overall features of cloud models, and it calculates the similarity by integral operation of the two expectation curves. MCM uses the maximum boundary curve to compute the similarity between cloud models. Assuming x_0 is the intersection of the expectation curves of cm_i and cm_j, the similarity in ECM can be obtained by Eq. (4):

$$s(cm_i, cm_j) = \int_{-\infty}^{x_0} y_i(x)dx + \int_{x_0}^{\infty} y_j(x)dx, \tag{4}$$

where $y_i(x)$ and $y_j(x)$ represent the expectation curves of cm_i and cm_j, respectively. ECM totally ignores the influence of He. However, the real He values of cloud models are usually different, and they convey some critical information about the cloud features. Especially in cases where two cloud models have the same Ex and En and different He, ECM cannot provide correct similarity results. In contrast, MCM exaggerates the influence of He, easily producing calculation errors.

(4) Similarity measurement method based on Euclidean distance [47]. This method employs Euclidean distance to calculate the similarity between cloud models, noted as EDCM. The Euclidean distance between cm_i and cm_j is calculated by Eq. (5):

$$s(cm_i, cm_j) = 1 \Big/ \left(1 + \sqrt{\sum (e_k^i - e_k^j)^2}\right). \tag{5}$$

Table 5
Five cloud models.

Numerical characteristics of cloud model	Cloud models				
	cm_1	cm_2	cm_3	cm_4	cm_5
Ex	9.500	9.200	9.300	9.200	8.500
En	0.255	0.286	0.372	0.301	0.578
He	0.017	0.020	0.091	0.195	0.034

EDCM can provide more accurate results than other similarity measurement methods. However, it can still treat two distinctively different cloud models as very similar objects. The deficiency of the Euclidean distance model is that the Euclidean distance depends on the measurement scale because it is assumed that all coordinates use the same scale. In some practical applications, such as similarity measurements of cloud models, the measurement scales of coordinates are different. Therefore, when the dimensions of variables have different measurement scales, errors will inevitably be introduced into calculation results based on the Euclidean distances. This is especially true when calculating the similarity between cloud models, because the measurement scales of three numerical characteristics in cloud models are very different, and *Ex* is usually many times greater than *En* and *He*.

6.2. A new method based on vector comparison

To overcome the deficiency of existing methods, we proposed a new similarity measurement method based on vector comparison, noted as VCM.

Assume that $\overrightarrow{E_i} = (e_{i,1}, e_{i,2}, e_{i,3}) = (Ex_i, En_i, He_i)$ and $\overrightarrow{E_j} = (e_{j,1}, e_{j,2}, e_{j,3}) = (Ex_j, En_j, He_j)$ are the two vectors corresponding to cloud models cm_i and cm_j. Then, the angle between cm_i and cm_j can be calculated by Eq. (6):

$$\left\langle \overrightarrow{E_j}, \overrightarrow{E_j} \right\rangle = \arccos\left(\overrightarrow{E_i} \bullet \overrightarrow{E_j} / |\overrightarrow{E_i}|\,|\overrightarrow{E_j}| \right), \tag{6}$$

where $\overrightarrow{E_i} \bullet \overrightarrow{E_j} = \sum_{k=1}^{n}(e_{i,k} \times e_{j,k})$; $|\overrightarrow{E_i}| = \sqrt{\sum_{k=1}^{n} e_{i,k}^2}$; $|\overrightarrow{E_j}| = \sqrt{\sum_{k=1}^{n} e_{j,k}^2}$. The orientation similarity between cm_i and cm_j can be obtained by Eq. (7):

$$O(\overrightarrow{E_i}, \overrightarrow{E_j}) = 1 - \left\langle \overrightarrow{E_i}, \overrightarrow{E_j} \right\rangle \Big/ 90. \tag{7}$$

Meanwhile, in view of the differences of measurement scales of QoS cloud models, we define a dimension similarity to calculate the similarities between $\overrightarrow{E_i}$ and $\overrightarrow{E_j}$ from the perspectives of three numerical characteristics separately. The dimension similarity between $\overrightarrow{E_i}$ and $\overrightarrow{E_j}$ can be calculated by Eq. (8):

$$D(\overrightarrow{E_i}, \overrightarrow{E_j}) = 1 - \frac{1}{n} \times \sum_{k=1}^{n} \frac{|e_{i,k} - e_{j,k}|}{e_{i,k}}, \tag{8}$$

where $D(\overrightarrow{E_i}, \overrightarrow{E_j})$ is an aggregated value of the absolute deviations in the three dimensions of QoS cloud models.

Then, the overall similarity between cm_i and cm_j can be calculated by Eq. (9):

$$s(cm_i, cm_j) = \alpha \times O(\overrightarrow{E_i}, \overrightarrow{E_j}) + (1-\alpha) \times D(\overrightarrow{E_i}, \overrightarrow{E_j}), \tag{9}$$

where α is the regulatory factor in determining the weights of the orientation similarity and dimension similarity, and α should be set in the range from 0.5 to 0.6. According to Eq. (9), the overall similarity between two cloud models is obtained by the vector comparison combining their orientation similarity and dimension similarity, and the different measurement scales of three numerical characteristics no longer cause the significant errors.

6.3. Numerical example

In this section, a numerical example is given to verify the efficiency of the VCM. Assume that there are five cloud models, noted as cm_1–cm_5, as shown in Table 5.

According to Table 5, the similarity between cm_2 and cm_1 is obviously larger than the similarity between cm_3 and cm_1. Nevertheless, the calculation results obtained by different similarity measurement methods vary greatly, shown in Table 6.

Table 6 demonstrates that the LICM, MCM and VCM can identify exactly cm_2 as the most similar cloud model with cm_1. However, this numerical example also reveals the obvious flaw of both LICM and MCM, in which some cloud models apparently different from cm_1, such as cm_3 and cm_4, got the fairly high similarity value. Especially, the difference among similarities obtained by the LICM is very small, and easily causes the calculation errors in the following process of QoS prediction. SCM, ECM and EDCM got the incorrect similarity values, and cannot accurately measure the differences between the cloud models. Table 6 exposed the superiority of VCM, which can provide the similarity measurement results coinciding well with the actual situation of cloud models.

7. Procedure of multi-valued collaborative QoS prediction

Let $m_{i,j}$ represent the QoS evaluation value of s_j provided by u_i. A $q \times p$ matrix of QoS evaluations is represented as follows:

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,p} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,p} \\ \vdots & \vdots & m_{i,j} & \vdots \\ m_{q,1} & m_{q,2} & \cdots & m_{q,p} \end{bmatrix}, \tag{10}$$

where q is the number of cloud service consumers and p is the number of cloud services.

Assume that the maximum length of a time series is $r \times d$, where d is the number of periods and r is the average length of a period. Considering the multi-valued feature of QoS evaluations, the multi-valued QoS matrix for s_i, noted as M_i, is as follows:

$$M_i = \begin{bmatrix} m_{1,i} \\ m_{2,i} \\ \vdots \\ m_{q,i} \end{bmatrix} = \begin{bmatrix} (p_{1,i}^1, p_{1,i}^2, \ldots, p_{1,i}^r)^1 & (p_{1,i}^1, p_{1,i}^2, \ldots, p_{1,i}^r)^2 & \cdots & (p_{1,i}^1, p_{1,i}^2, \ldots, p_{1,i}^r)^d \\ (p_{2,i}^1, p_{2,i}^2, \ldots, \varnothing)^1 & (p_{2,i}^1, \varnothing, \ldots, \varnothing)^2 & \cdots & (\varnothing, \varnothing, \ldots, p_{2,i}^r)^d \\ \vdots & \vdots & \vdots & \vdots \\ (\varnothing, \varnothing, \ldots, p_{q,i})^1 & (p_{q,i}^1, p_{q,i}^2, \ldots, p_{q,i}^r)^2 & \cdots & (\varnothing, p_{q,i}^2, \ldots, p_{q,i}^r)^d \end{bmatrix}, \tag{11}$$

where $m_{q,i}$ represents the QoS evaluations of s_i submitted by u_q; $(p_{1,i}^1, p_{1,i}^2, \ldots, p_{1,i}^r)^d$ is the QoS data for s_i in period tp_d submitted by u_1, which is a complete time series data with the length r; $\varnothing$ represents an unknown value; $(\varnothing, \varnothing, \ldots, p_{q,i})^1$ is the single-value QoS evaluation of s_i submitted by u_q in tp_1, where the length of the time series data is equal to 1; and $(p_{2,i}^1, p_{2,i}^2, \ldots, \varnothing)^1$ is the incomplete time series data for s_i in tp_1 submitted by u_2.

Then, the procedure of multi-valued collaborative QoS prediction is as follows:

(1) Preprocessing the QoS evaluations data into a cloud model for every period.

For period tp_i such that the QoS evaluations are the time series data, all evaluations in tp_i are sent to RCG, and a cloud model including three numerical characteristics can be established for tp_i according to Eq. (1). For period tp_i satisfying the QoS evaluations are the single-valued data, let p^i be the unique evaluation value

Table 6
Similarity measurement results between cloud models.

Measurement methods	Cloud models				
	cm_1	cm_2	cm_3	cm_4	cm_5
SCM	0.9999	0.9933	**0.9940**	0.9930	0.9793
LICM	1.0000	**1.0000**	0.9998	0.9998	0.9994
ECM	1.0000	0.6135	**0.7403**	0.6223	0.3075
MCM	1.0000	**0.6731**	0.6432	0.5668	0.3938
EDCM	1.0000	0.7692	**0.7860**	0.7437	0.5142
VCM	1.0000	**0.9314**	−0.4881	−2.1295	0.1820

in tp_i; then, the cloud model of tp_i can be transformed into a specific cloud model $\{p^i, 0, 0\}$. Thus, the cloud model matrix of QoS evaluations for s_i is obtained as follows:

$$C_i = \begin{bmatrix} cm_{1,1} & cm_{1,2} & \cdots & cm_{1,d} \\ cm_{2,1} & cm_{2,2} & \cdots & cm_{2,d} \\ \vdots & \vdots & cm_{i,j} & \vdots \\ cm_{q,1} & cm_{q,2} & \cdots & cm_{q,d} \end{bmatrix} = \begin{bmatrix} (Ex_1^1, En_1^1, He_1^1) & (Ex_1^2, En_1^2, He_1^2) & \cdots & (Ex_1^d, En_1^d, He_1^d) \\ (Ex_2^1, En_2^1, He_2^1) & (Ex_2^2, En_2^2, He_2^2) & \cdots & (Ex_2^d, En_2^d, He_2^d) \\ \vdots & \vdots & \vdots & \vdots \\ (Ex_q^1, En_q^1, He_q^1) & (Ex_q^2, En_q^2, He_q^2) & \cdots & (Ex_q^d, En_q^d, He_q^d) \end{bmatrix}, \tag{12}$$

where $cm_{i,j} = \left(Ex_i^j, En_i^j, He_i^j\right)$ represents the QoS cloud model of u_i in tp_j.

(2) Calculating the QoS cloud model similarity between the potential user and other consumers.

Assume that u_1 is the potential user, compare QoS cloud model of u_1 with QoS cloud models of other $q-1$ consumers in every period, and the similarity matrix of QoS cloud models can be noted as follows:

$$SIM = \begin{bmatrix} sim_2^1 & sim_2^2 & \cdots & sim_2^d \\ sim_3^1 & sim_3^2 & \cdots & sim_3^d \\ \vdots & \vdots & sim_i^k & \vdots \\ sim_q^1 & sim_q^2 & \cdots & sim_q^d \end{bmatrix}, \tag{13}$$

where $sim_i^k = s(cm_{1,k}, cm_{i,k})$ represents the similarity between u_1 and u_i in tp_k. The similarity between QoS cloud models can be calculated by employing those methods introduced in Section 6, such as SCM, LICM, ECM, MCM, EDCM or VCM.

(3) Determining the objective weights of periods based on sensitivity scores.

Considering that the weight of a period varies widely in different application scenarios, it is not appropriate to synthesize the similarity values of periods with the weighted mean method. The analytic hierarchy process (AHP) is a scientific evaluation analysis method. But its accuracy is not high enough. Combining the fuzzy logic, the FAHP method [48] can overcome the shortcomings of AHP, which is suitable for solving the multiple attribute decision-making problems. Therefore, the FAHP method is used to identify the objective weights of different periods.

Supposing $B = (b_{i,j})_{d\times d}$ is the sensitivity score for periods submitted by a potential user, which is a fuzzy judgment matrix with $0 \le b_{i,j} \le 1$, where d is the number of periods, and $b_{i,j}$ is the importance ratio of period tp_i compared to period tp_j. b_{ij} can be determined according to the five-level complementary scale shown in Table 7.

If $b_{i,j} + b_{j,i} = 1$ and $b_{i,i} = 0.5$, B is a fuzzy complementary judgment matrix. Giving an integer k, if $b_{i,j} = b_{i,k} - b_{j,k} + 0.5$, B is a fuzzy consistency matrix. For transforming B into a fuzzy complementary judgment matrix, the sum of each row of B is computed by $b_i = \sum_{k=1}^{n} b_{i,k}$ and the mathematical manipulation is performed by Eq. (14):

Table 7
Five-level complementary scales used in FAHP method.

Scales	Description
0.1	tp_i is much less important than tp_j
0.3	tp_i is less important than tp_j
0.5	tp_i is equally important as tp_j
0.7	tp_i is more important than tp_j
0.9	tp_i is much more important tp_j

$$c_{i,j} = 0.5 + (b_i - b_j)/2(n-1). \tag{14}$$

A new fuzzy matrix $C = (c_{i,j})_{n\times n}$ can be obtained, which is a fuzzy consistency judgment matrix. Then, the sum of each row is computed and standardized. Accordingly, the weight of tp_i is calculated by Eq. (15):

$$w_i = \frac{1}{n(n-1)} \sum_{j=1}^{n} c_{i,j} + \frac{n}{2} - 1, \quad i = 1, 2, \ldots, n. \tag{15}$$

The weight matrix of periods (W) can be established as follows:

$$W = \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & 0 & \cdots \\ \vdots & \vdots & w_i & \vdots \\ 0 & 0 & \cdots & w_d \end{bmatrix}, \tag{16}$$

where w_i represents the importance degree of tp_i for the potential user; $0 \le w_i \le 1$, $\sum_{i=1}^{d} w_i$.

(4) Aggregating the cloud models in multiple periods and obtain the comprehensive similarities between the potential user and other cloud service consumers.

Aggregate the cloud models with weight matrix (W) in multiple periods by Eq. (17):

$$C = SIM \times W = \begin{bmatrix} sim_2^1 & sim_2^2 & \cdots & sim_2^d \\ sim_3^1 & sim_3^2 & \cdots & sim_3^d \\ \vdots & \vdots & sim_i^k & \vdots \\ sim_q^1 & sim_q^2 & \cdots & sim_q^d \end{bmatrix} \begin{bmatrix} w_1 & 0 & \cdots & 0 \\ 0 & w_2 & 0 & \cdots \\ \vdots & \vdots & w_i & \vdots \\ 0 & 0 & \cdots & w_d \end{bmatrix} = \begin{bmatrix} c_{2,1} & c_{2,1} & \cdots & c_{2,d} \\ c_{3,1} & c_{3,1} & \cdots & c_{3,d} \\ \vdots & \vdots & \vdots & \vdots \\ c_{q,1} & c_{q,1} & \cdots & c_{q,d} \end{bmatrix}. \tag{17}$$

Then, the comprehensive similarity between u_1 and u_i is obtained by Eq. (18):

$$C(u_1, u_i) = \sum_{j=2}^{d} c_{i,j}. \tag{18}$$

(5) Selecting the neighboring users for potential user u_1 according to a similarity threshold.

The neighboring users set of potential user u_1, noted as NU, can be selected by Eq. (19):

$$NU = \{u_i | u_i \in U, C(u_1, u_i) \geq s^{th}\}, \tag{19}$$

where U is the set of cloud service consumers and s^{th} is the similarity threshold.

(6) Predicting the QoS according to the time series data provided by the neighboring users.

The unknown QoS value of service s_k in timeslot t_i is predicted for the potential user u_1 by Eq. (20):

$$pred(u_1, i) = \frac{\sum_{b \in NU} C(u_1, b) \times m_{b,k}^i}{\sum_{b \in NU} C(u_1, b)}, \tag{20}$$

where $m_{b,k}^i$ is the QoS evaluation data of s_k provided by u_b in t_i.

8. Experiments

8.1. Experiment setup

The WS-DREAM dataset #3 [7,23] is used to verify the effectiveness of the proposed approach. We firstly analyze the QoS data from this real-world dataset in experiments. The coefficients of variation (CV) of response time data of 3873 services and the CV of throughput data of 2630 services are larger than 1.0, as shown in Fig. 4. The results explain the dynamism and vulnerability of cloud computing environment. The main reasons for this finding are significant differences of client features among users, unpredictable network congestion, and unexpected exceptions.

The mean absolute error (MAE) method is used to measure the accuracy of QoS prediction approaches, which is defined by Eq. (21):

$$MAE = \frac{1}{N} \sum_{i=1}^{N} \left| v_i^* - v_i^o \right|, \tag{21}$$

where N denotes the total number of executed QoS predictions; v_i^* represents the real QoS value in timeslot t_i experienced by the potential user; v_i^o represents the predicted QoS value in t_i for the potential user. The smaller the MAE is, the better the accuracy is.

In the following experiments, the regulatory factor α of VCM is set as 0.6. To ensure the repeatability of experiments, all of the periods are given equal weights.

8.2. Impacts of similarity threshold s^{th}

In the procedure of multi-valued collaborative QoS prediction, similarity threshold s^{th} decides the number of neighboring users. To analyze the impact of s^{th}, we compare the MAE values when s^{th} varies from 0.06 to 0.54 with a step value of 0.03 in our experiments. Those services with their CVs smaller than 1.5 are selected as candidate services and the density of training matrix is set as 30% for ensuring the relative stability of experiment results. In the experiments, 100 consumers are chosen at random and a potential user is selected from these consumers at random. Fig. 5 shows the experiment results.

According to Fig. 5(a) and (b), the prediction quality of response time is the best when $s^{th} = 0.18$, and the MAE values become rather poor when $s^{th} > 0.30$; the prediction quality of the throughput is the best when $s^{th} = 0.15$, and the MAE values become rather poor when $s^{th} > 0.36$. We can analyze the experiment results by comparing Fig. 5 with Fig. 4 as follows: Fig. 4 shows that the CV values of the throughput are much larger than the CV values of response time; the large CV values mean the high dispersion of data and the low similarity of QoS evaluations; for getting the enough neighboring users, the smaller similarity threshold of throughput data between consumers are required in contrast to the response time data. Fig. 5 also displays that the MAE values of throughput rise sharply when $s^{th} > 0.3$. The reason is that fewer neighboring users in NU fail to resist the disruption of noisy data to prediction results.

8.3. Impacts of period length

Considering that the period length has a major impact on the accuracy of cloud models and the precision of QoS prediction. In order to analyze the impacts of period length, some response time data about service #741 invoked by user #9–#13 in timeslot #1–#40 is extracted from the dataset, shown in Table 8.

In Table 8, 40 unknown QoS values are marked by a strikethrough. In the experiment, we compared the proposed approach of multi-valued collaborative QoS prediction employing VCM to measure the similarity between cloud models, noted as QP_VCM, with the other five approaches as follows: (1) The multi-valued collaborative QoS prediction employing LICM, noted as QP_LICM; (2) The multi-valued collaborative QoS prediction employing EDCM, noted as QP_EDCM; (3) The user-based collaborative filtering employing Pearson correlation coefficient (PCC) to measure the similarity between two users, noted as UPCC; (4) item-based collaborative filtering employing cosine similarity to identify the similar users, noted as ICOS; and (5) item-based collaborative filtering employing improved cosine similarity to identify the similar users, noted as IICOS. In ICOS and IICOS, a timeslot is viewed as an item. Considering the obvious drawback of the SCM, ECM and MCM in measuring the similarity between cloud models, they were not compared in experiments. Additionally, MF-based prediction approaches were also not compared in the experiments because the evaluation data in every period is quite sparse. The MAE values obtained by the six approaches when the period length varies from 6 to 40 are shown in Table 9.

Table 9 demonstrates that four approaches including UPCC, QP_LICM, QP_EDCM and QP_VCM, get the smallest MAE values respectively when the period length is equal to 8 compared to other period lengths. Especially, QP_VCM obtained the optimal MAE value when the period length is equal to 8. In general, more periods may depict the variation characteristic QoS more exactly since the service status and the network environment change over time. However, continuous monitoring of cloud services is still time-consuming and costly, which causes that the total number of timeslots is limited. In that case, more periods mean a smaller amount of timeslots in every period, which make it difficult to employ the cloud model to accurately describe the QoS feature of a cloud service in every period based on the limited timeslots data. Accordingly, dividing the limited timeslots into more periods usually produces a lower precision of QoS prediction.

8.4. Comparisons of prediction approaches

In the following experiments, we compared QP_VCM with QP_LICM, QP_EDCM, UPCC, ICOS and IICOS when the density of training matrix varies from 5% to 32%. $s^{th} = 0.18$ for the response time data and $s^{th} = 0.15$ for the throughput data. The period length is set as 8. The results are shown in Fig. 6.

Fig. 6 demonstrates that the time series analysis based on QoS cloud models facilitates QP_VCM, QP_LICM and QP_EDCM to achieve a higher precision of QoS prediction compared with traditional approaches, such as UPCC, ICOS and IICOS. The main reason is that UPCC, ICOS and IICOS are incapable of extracting the

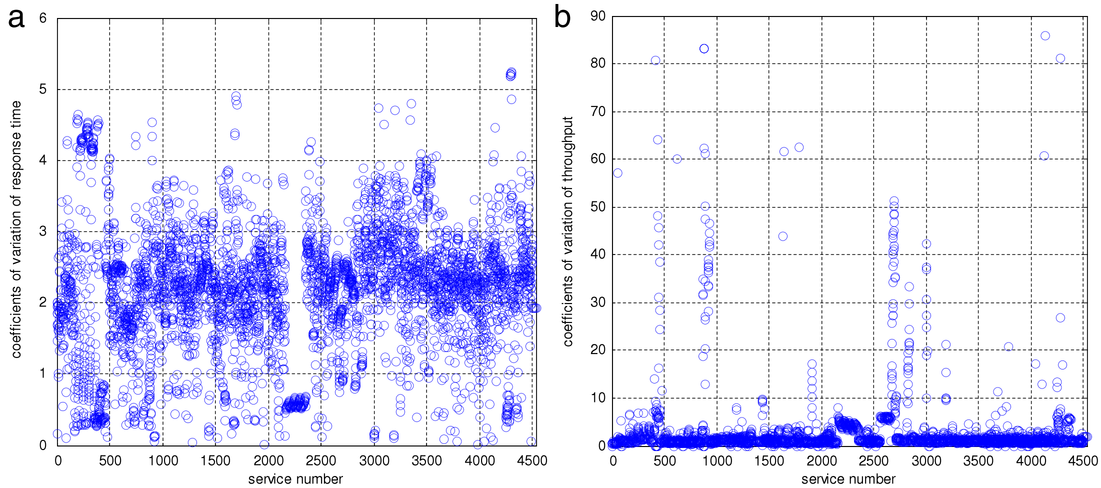


Fig. 4. Coefficients of variation. (a) response time; (b) throughput.

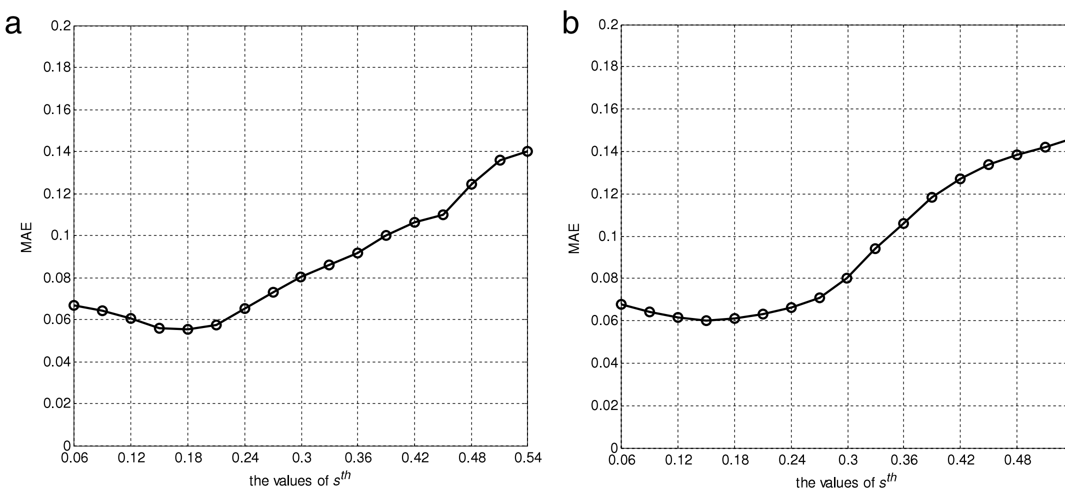


Fig. 5. Impacts of similarity threshold s^{th}. (a) response time; (b) throughput.

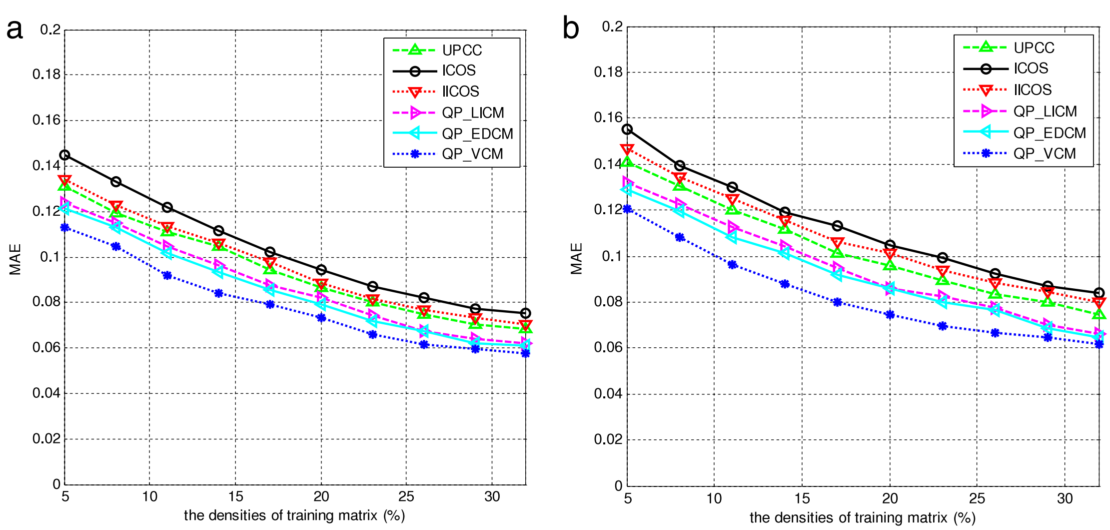


Fig. 6. Comparisons of prediction approaches. (a) response time; (b) throughput.

variation characteristic QoS evaluations to improve the selection quality of neighboring users and lead to the larger MAE values than those of QP_VCM, QP_LICM and QP_EDCM. The reason that QP_VCM obtained the optimal prediction quality lies in the advantage of the proposed VCM in similarity calculation between cloud models. Because VCM calculates the similarity with consideration of the difference of measurement scales, VCM can ensure the identification accuracy of neighboring users. In addition, Fig. 6 showed that the results are consistent with those shown in Fig. 4. According to Fig. 4, the throughput data is more

Table 8
Some response time data from the dataset.

Users	Timeslots									
	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10
User #9	0.283	0.282	0.281	0.296	0.252	~~0.381~~	0.319	0.240	0.255	0.38
User #10	~~0.495~~	0.279	0.449	~~0.297~~	0.247	0.275	0.279	0.290	0.272	0.294
User #11	0.554	~~0.230~~	0.256	0.212	0.260	0.833	0.247	0.258	~~0.244~~	0.207
User #12	0.498	0.244	~~0.242~~	0.239	0.205	0.239	0.255	~~0.215~~	0.235	0.576
User #13	0.378	0.311	0.295	0.277	~~0.254~~	0.321	~~0.291~~	0.282	0.324	~~0.442~~

Users	Timeslots									
	#11	#12	#13	#14	#15	#16	#17	#18	#19	#20
User #9	0.253	0.441	~~0.245~~	~~0.258~~	0.289	0.316	0.350	0.302	~~0.262~~	0.330
User #10	0.290	0.293	0.296	0.255	0.315	0.247	~~0.292~~	~~0.294~~	0.266	0.251
User #11	0.216	0.219	0.199	0.599	0.272	~~0.199~~	0.250	0.228	0.507	~~0.201~~
User #12	~~0.239~~	0.508	0.253	0.243	~~0.248~~	0.211	0.514	0.263	0.204	0.252
User #13	0.300	~~0.272~~	0.321	0.303	0.274	0.281	0.312	0.422	0.264	0.328

Users	Timeslots									
	#21	#22	#23	#24	#25	#26	#27	#28	#29	#30
User #9	~~0.307~~	0.320	0.283	~~0.258~~	0.254	0.267	0.373	0.258	0.477	0.407
User #10	0.310	~~0.233~~	0.276	0.338	~~0.239~~	0.326	0.440	0.311	0.431	0.394
User #11	0.476	0.200	~~0.235~~	0.245	0.218	0.530	~~0.225~~	0.236	0.231	0.678
User #12	0.241	0.202	0.399	0.539	0.201	~~0.225~~	0.216	~~0.860~~	0.221	0.215
User #13	0.354	0.263	1.214	0.277	0.363	0.413	0.323	0.353	~~0.397~~	~~0.433~~

Users	Timeslots									
	#31	#32	#33	#34	#35	#36	#37	#38	#39	#40
User #9	0.266	0.335	0.311	0.304	~~1.189~~	0.334	0.326	0.348	0.450	~~0.730~~
User #10	0.362	~~0.315~~	0.419	0.320	0.323	0.325	0.425	~~0.380~~	0.301	0.304
User #11	0.236	0.274	~~0.560~~	0.227	0.249	0.240	~~0.219~~	1.290	0.221	0.217
User #12	~~0.276~~	0.236	0.209	0.237	0.246	~~0.263~~	0.202	0.227	0.240	0.220
User #13	0.333	0.370	0.316	~~0.361~~	0.376	0.332	0.502	0.401	~~0.404~~	0.352

Table 9
Impacts of period length.

Period length	QoS prediction approaches					
	UPCC	ICOS	IICOS	QP_LICM	QP_EDCM	QP_VCM
6	0.0723	0.0970	0.1033	0.0402	0.0399	0.0393
8	**0.0428**	0.0664	0.0697	**0.0301**	**0.0300**	**0.0295**
10	0.0735	0.0773	0.0732	0.0377	0.0376	0.0370
15	0.0581	0.0688	**0.0559**	0.0395	0.0394	0.0387
20	0.0785	0.0724	0.0796	0.0501	0.0500	0.0494
25	0.1047	**0.0656**	0.0662	0.0546	0.0546	0.0542
30	0.0971	0.0908	0.0883	0.0795	0.0794	0.0792
35	0.1179	0.1074	0.1085	0.1042	0.1041	0.1039
40	0.1271	0.1129	0.1122	0.1142	0.1141	0.1135

discrete than the response time data; and the more discrete the evaluation data is, the larger the MAE of QoS prediction based on these data becomes.

9. Conclusions and future work

In the dynamic and vulnerable cloud computing environment, predicting accurately the uncertain QoS of cloud services for potential users has become a tough task. The continuous monitoring of cloud services can provide the time series QoS data and help potential users to investigate QoS during specified periods from a comprehensive perspective. However, the multi-valued QoS evaluations consisting of time series data and single-valued data, in turn, present a new challenge for the QoS prediction research.

Aiming at the time series feature of QoS and the multi-valued feature of QoS evaluations, this paper proposes a collaborative QoS prediction approach based on cloud model theory and time series analysis, in which the multi-valued QoS evaluations are preprocessed and modeled as a set of cloud models in multiple periods. In order to identify exactly the neighboring users for potential users, we put forward a new vector comparison method combining the orientation similarity and dimension similarity to improve the accuracy of similarity calculation between cloud models and employed the FAHP method to determine the objective weights of periods according to application requirements of potential users. The unknown QoS values of cloud services are predicted with consideration of both the periodic variations of QoS and the users' application requirements in different periods. The experiments based on a real-world dataset demonstrate that this approach can provide high accuracy of multi-valued collaborative QoS prediction for potential users in cloud paradigm.

As for future work, we will study the following problems:

(1) to automatically identify the optimal period length for specific application scenarios; and
(2) to achieve the dynamic pattern recognition between the time series feature of QoS and the users' application requirement in multiple periods.

Acknowledgment

This work was partially supported by National Natural Science Foundation of China (No. 61602171).

References

[1] P. Casas, R. Schatz, Quality of experience in cloud services: Survey and measurements, Comput. Netw. 68 (1) (2014) 149–165.

[2] F. Lin, W.H. Zeng, L.Q. Yang, Y. Wang, S.F. Lin, J.S. Zeng, Cloud computing system risk estimation and service selection approach based on cloud focus theory, Neural Comput. Appl. (2016) http://dx.doi.org/10.1007/s00521-00015-02166-00527.

[3] S.K. Garg, S. Versteeg, R. Buyya, A framework for ranking of cloud computing services, Future Gener. Comput. Syst. 29 (4) (2013) 1012–1023.

[4] Z. Ye, S. Mistry, A. Bouguettaya, H. Dong, Long-term QoS-aware cloud service composition using multivariate time series analysis, IEEE Trans. Serv. Comput. 9 (3) (2016) 382–393.

[5] C.Y. Mao, J.F. Chen, D. Towey, J.F. Chen, X.Y. Xie, Search-based QoS ranking prediction for web services in cloud environments, Future Gener. Comput. Syst. 50 (2015) 111–126.

[6] Y. Ma, S.G. Wang, P.C.K. Hung, C.-H. Hsu, Q.B. Sun, F.C. yang, A highly accurate prediction algorithm for unknown web service QoS values, IEEE Trans. Serv. Comput. 9 (4) (2016) 511–523.

[7] Y.L. Zhang, Z.B. Zheng, M.R. Lyu, WSPred: A time-aware personalized QoS prediction framework for web services, in: Proc. of the 2011 IEEE 22nd International Symposium on Software Reliability Engineering, (ISSRE), IEEE, 2011, pp. 210–219.

[8] C.Y. Yu, L.P. Huang, A web service QoS prediction approach based on time- and location-aware collaborative filtering, Serv. Oriented Comput. Appl. 10 (2016) 135–149.

[9] M. Jarschel, D. Schlosser, S. Scheuring, T. Hoßfeld, Gaming in the clouds: QoE and the users' perspective, Math. Comput. Modelling 57 (11) (2013) 2883–2894.

[10] H. Ma, Z.G. Hu, K.Q. Li, H.Y. Zhang, Toward trustworthy cloud service selection: A time-aware approach using interval neutrosophic set, J. Parallel Distrib. Comput. 96 (2016) 75–94.

[11] H. Ma, Z.G. Hu, L. Yang, T. Song, User feature-aware trustworthiness measurement of cloud services via evidence synthesis for potential users, J. Vis. Lang. Comput. 25 (6) (2014) 791–799.

[12] P. Leitner, J. Ferner, W. Hummer, S. Dustdar, Data-driven and automated prediction of service level agreement violations in service compositions, Distrib. Parallel Databases 31 (3) (2013) 447–470.

[13] Amazon, Amazon EC2 Service Level Agreement, Aug. 28, 2016. URL: https://aws.amazon.com/cn/ec2/sla/.

[14] Z.B. Zheng, H. Ma, M.R. Lyu, I. King, Collaborative web service QoS prediction via neighborhood integrated matrix factorization, IEEE Trans. Serv. Comput. 6 (3) (2013) 289–299.

[15] X. Luo, M.C. Zhou, Y.N. Xia, Q.S. Zhu, Generating highly accurate predictions for missing QoS data via aggregating nonnegative latent factor models, IEEE Trans. Neural Netw. Learn. Syst. 27 (3) (2016) 524–537.

[16] Y.L. Zhang, Z.B. Zheng, M.R. Lyu, Exploring latent features for memory-based QoS prediction in cloud computing, in: Proc. of 2011 30th IEEE Symposium on Reliable Distributed Systems, (SRDS 2011), IEEE, Madrid, Spain, 2011, pp. 1–10.

[17] X. Luo, Y.X. Lv, R.X. Li, Y. Chen, Web service QoS prediction based on adaptive dynamic programming using fuzzy neural networks for cloud services, IEEE Access 3 (2015) 2260–2269.

[18] J.W. Yin, W. Lo, S.G. Deng, Y. Li, Z.H. Wu, N.X. Xiong, Colbar: A collaborative location-based regularization framework for QoS prediction, Inform. Sci. 265 (2014) 68–84.

[19] S. Ding, S.L. Yang, Y.T. Zhang, C.Y. Liang, C.Y. Xia, Combining QoS prediction and customer satisfaction estimation to solve cloud service trustworthiness evaluation problems, Knowl.-Based Syst. 56 (1) (2014) 216–225.

[20] Cloud Security Alliance, CSA Security, Trust & Assurance Registry (STAR), 2016-05-01. URL: https://cloudsecurityalliance.org/star/.

[21] Yunzhiliang.net, Quality evaluations of cloud, May 2016. URL: http://www.yunzhiliang.net/cloudtest/cloudtest.html.

[22] China Cloud Computing Promotion and Policy Forum, The evaluation results of trusted services authentication, May 2016. URL: http://www.3cpp.org/news/201.html.

[23] Chinese University of Hong Kong, WS-DREAM, 2016-05-01. URL: http://wsdream.com/dataset.html.

[24] D. Rosaci, G.M. Sarné, Recommending multimedia web services in a multi-device environment, Inf. Syst. 38 (2) (2013) 198–212.

[25] Y. Wei, M.B. Blake, An agent-based services framework with adaptive monitoring in cloud environments, in: Proc. of the 21st IEEE International Workshop on Enabling Technologies - Infrastructure for Collaborative Enterprises, (WETICE), IEEE, ISAE, Toulouse, France, 2012, pp. 4–9.

[26] C. Yu, L. Huang, Time-aware collaborative filtering for QoS-based service recommendation, in: Proc. of the 2014 IEEE International Conference on Web Services, (ICWS), IEEE, 2014, pp. 265–272.

[27] Z.U. Rehman, O.K. Hussain, F.K. Hussain, User-side cloud service management: State-of-the-art and future directions, J. Netw. Comput. Appl. 55 (2015) 108–122.

[28] Z.B. Zheng, M.R. Lyu, Collaborative reliability prediction of service-oriented systems, in: Proc. of the 32nd ACM/IEEE International Conference on Software Engineering, (ICSE2010), ACM, 2010, pp. 35–44.

[29] X. Chen, X.D. Liu, Z.C. Huang, H.L. Sun, RegionKNN: A scalable hybrid collaborative filtering algorithm for personalized web service recommendation, in: Proc. of the 2010 IEEE International Conference on Web Services, (ICWS), IEEE, 2010, pp. 9–16.

[30] Y. Hu, Q.M. Peng, X.H. Hu, A time-aware and data sparsity tolerant approach for web service recommendation, in: Proc. of the 2014 IEEE International Conference on Web Services, (ICWS), IEEE, 2014, pp. 33–40.

[31] Y. Zhong, Y. Fan, K. Huang, W. Tan, J. Zhang, Time-aware service recommendation for mashup creation, IEEE Trans. Serv. Comput. 8 (3) (2015) 356–368.

[32] Y.S. Xu, J.W. Yin, S.G. Deng, N.N. Xiong, J.B. Huang, Context-aware QoS prediction for web service recommendation and selection, Expert Syst. Appl. 53 (2016) 75–86.

[33] W. Lo, J.W. Yin, S.G. Deng, Y. Li, Z.H. Wu, Collaborative web service QoS prediction with location-based regularization, in: Proc. of the 2012 IEEE 19th International Conference on Web Services, (ICWS), IEEE, 2012, pp. 464–471.

[34] J. Yin, Y. Xu, Personalized QoS-based web service recommendation with service neighborhood-enhanced matrix factorization, Int. J. Web Grid Serv. 11 (1) (2015) 39–56.

[35] S. Kumar, M.K. Pandey, A. Nath, K. Subbiah, Missing QoS-values predictions using neural networks for cloud computing environments, in: Proc. of 2015 International Conference on Computing and Network Communications, (CoCoNet), IEEE, Trivandrum, 2015, pp. 414–419.

[36] D.Y. Li, H.J. Meng, X.M. Shi, Membership clouds and membership cloud generators, J. Comput. Res. Dev. 32 (6) (1995) 15–20.

[37] G. Wang, C. Xu, D. Li, Generic normal cloud model, Inform. Sci. 280 (2014) 1–15.

[38] C.S. Li, S.L. Wang, L. Kang, L. Guo, Y. Cao, Trust evaluation model of cloud manufacturing service platform, Int. J. Adv. Manuf. Technol. 75 (1–4) (2014) 489–501.

[39] H.Y. Zhang, P. Ji, J.Q. Wang, X.H. Chen, A neutrosophic normal cloud and its application in decision-making, Cog. Comput. 8 (4) (2016) 649–669.

[40] J. Wu, Research on methods of image feature extraction based on cloud model and data field (Ph.D.thesis), Wuhan University, Wuhan, 2010.

[41] H.L. Li, C.H. Guo, Piecewise aggregate approximation method based on cloud model for time series, Control Decis. 26 (10) (2011) 1525–1529.

[42] S.X. Wang, L. Zhang, S. Wang, X. Qiu, A cloud-based trust model for evaluating quality of web services, J. Comput. Sci. Tech. 25 (6) (2010) 1130–1142.

[43] C.Y. Liu, M. Feng, X.J. Dai, D.Y. Li, A new algorithm of backward cloud, J. Syst. Simul. 16 (11) (2004) 2417–2420.

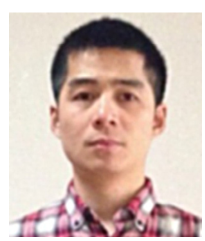
[44] Y. Zhang, D. Zhao, D.Y. Li, The similar cloud and the measurement method, Inf. Control 33 (2) (2004) 129–132.

[45] G.W. Zhang, D.Y. Li, P. Li, J.C. Kang, G.S. Chen, A collaborative filtering recommendation algorithm based on cloud model, J. Softw. 18 (10) (2007) 2403–2411.

[46] H.L. Li, C.H. Guo, W.R. Qiu, Similarity measurement between normal cloud models, Acta Electron. Sin. 39 (11) (2011) 2561–2567.

[47] L.C. Zhang, Y.H. Yang, X.H. Zhao, SaaS decision-making method based on cloud model, Acta Electron. Sin. 43 (5) (2015) 987–992.

[48] L. Mikhailov, P. Tsvetinov, Evaluation of services using a fuzzy analytic hierarchy process, Appl. Soft Comput. 5 (1) (2004) 23–33.



Hua Ma received his B.S. degree in computer science and technology and his M.S. degree in computer application technology from Central South University, Changsha, China in 2003 and in 2006, respectively, and received his Ph.D. degree in software engineering from Central South University, Changsha, China in 2016. His research interests focus on cloud computing, trusted computing and high performance computing. He is currently an associate professor of Hunan Normal University, Changsha, China. He is a member of the IEEE.

Haibin Zhu is a Full Professor of the Department of Computer Science and Mathematics, Founding Director of the Collaborative Systems Laboratory, Nipissing University, Canada. He has 150+ research publications. He is a senior member of IEEE and is serving and served as co-chair of the technical committee of Distributed Intelligent Systems of IEEE SMC Society, and associate editor of IEEE SMC Magazine. He is a founding researcher of Role-Based Collaboration and Adaptive Collaboration. Please feel free to visit Dr. Zhu's Website at http://faculty.nipissingu.ca/haibinz/ for more information.

Zhigang Hu received his B.S. degree and his M.S. degree from Central South University, Changsha, China in 1985 and in 1988, respectively, and received his Ph.D. degree from Central South University, Changsha, China in 2002. His research interests focus on the parallel computing, high performance computing and energy efficiency of cloud computing. He is currently a professor and Ph.D. supervisor of Central South University, Changsha, China.

Wensheng Tang received his B.S. degree from Hunan Normal University, Changsha, China in 1992, and received his M.S. degree and Ph.D. degree from National University of Defense Technology, Changsha, China in 1997 and in 2009, respectively. His research interests focus on the distributed computing and cloud computing. He is currently a professor of Hunan Normal University, Changsha, China.

Pingping Dong received her B.S. degree, M.S. degree and Ph.D. degree from the School of Information Science and Engineering at Central South University, China in 2008, in 2011 and in 2015, respectively. Her research interests focuses on the wide area networks, wireless local area networks and distributed computing. Currently she is a lecturer of Hunan Normal University, Changsha, China.