Accepted Manuscript

A Novel TOPSIS Evaluation Scheme for Cloud Service Trustworthiness Combining Objective and Subjective Aspects

Lilei Lu, Yuyu Yuan

PII: S0164-1212(18)30092-X
DOI: 10.1016/j.jss.2018.05.004
Reference: JSS 10157

To appear in: *The Journal of Systems & Software*

Received date: 1 September 2017
Revised date: 16 April 2018
Accepted date: 3 May 2018

Please cite this article as: Lilei Lu, Yuyu Yuan, A Novel TOPSIS Evaluation Scheme for Cloud Service Trustworthiness Combining Objective and Subjective Aspects, *The Journal of Systems & Software* (2018), doi: 10.1016/j.jss.2018.05.004

Highlights

- The common QoS metrics of cloud services and quantifications are proposed.
- A CSTE model is given to apply the proposed evaluation scheme.
- The objectivity of cloud services is considered from two aspects.
- Trust preference reflecting the subjectivity of trust is introduced and obtained.
- The combined weight fusing entropy weight and trust preference is used in TOPSIS.

A Novel TOPSIS Evaluation Scheme for Cloud Service Trustworthiness Combining Objective and Subjective Aspects

Lilei Lu[a,b], Yuyu Yuan[a]

[a] *Key Laboratory of Trustworthy Distributed Computing and Service, Ministry of education, School of Software, Beijing University of Posts and Telecommunications, Beijing 100876, Beijing, PR China*

[b] *Department of Computer Science, Tangshan Normal University, Tangshan 063000, Hebei, PR China*

Abstract

Cloud computing has been paid more attention due to its outstanding advantages. However, trust issue greatly affects the adoption of cloud services. Selecting trustworthy cloud service from those with same functionality but different qualities has become a significant challenge. Since trustworthiness evaluation is multi-dimensional, how to assign weight for each influence factor is a non-trivial problem. In this paper, we put forward a novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects. First, we consider the objectivity of cloud services from two facets. For one thing, we concern the reliability of QoS information source and utilize monitored values of QoS attributes rather than feedback ratings from consumers. For another, we employ objective entropy weight for different QoS attributes to decrease the effect of false or artificial parameter information. Second, we introduce trust preference that reflects the subjectivity of trust. Most important of all, we propose the combined weight integrating the two aspects and apply it to TOPSIS method to develop a novel evaluation scheme. The results of two experiments based on the existing QWS dataset from real Web services demonstrate its feasibility, effectiveness, and better Satisfaction Degree from the perspective of consumers.

Keywords: Trust, TOPSIS, Cloud service trustworthiness, QoS, Trust preference

1. Introduction

In recent years, cloud computing has been paid more and more attention as a growing computing paradigm for supporting flexible and on-demand services which includes infrastructure as a service (IaaS), platform as a service (PaaS) and software as a service (SaaS) [1]. It consists of five striking features [2]: (1) common infrastructure using dynamic and shared resource pool, (2) location independence supporting everywhere service, (3) online accessibility obtained from anywhere over the network, (4) on-demand resources acquired according to consumer requirements, (5) utility pricing charged depending on usage. These features make it possible to provide tempting services with low cost, usability, reliability, and feasibility, but not require extra or pre-deployed infrastructure. As a result, diverse cloud services are inevitably attracting not only many small and medium enterprises (SMEs) but also lots of individual consumers to construct business systems or personal applications [3].

However, trust has been regarded as one of the most concerned issues for the acceptance and growth of cloud services in cloud computing environment due to several reasons [4]. First of all, the highly distributed, virtual and non-transparent nature of cloud computing spells a considerable obstacle for the adoption and market success of cloud services [5]. Meanwhile, dynamic and diverse services are hosted in complex distributed systems, and they are offered to consumers at different levels [6, 7]. Most of all, the control of computing resources is partly or completely transferred to cloud service providers (CSPs) [8]. Cloud service consumers (CSCs) are vulnerable to risks due to potentially incomplete or distorted information provided by CSPs. Hence, potential CSCs cannot ensure whether CSPs are capable of providing reliable services in accordance with their own requirements [9]. In addition, competing CSPs often provide similar services with the same kind of functionality. The diversity of services increases the complexities and difficulties to choose trustworthy service for CSCs. In conclusion, trust represents the confidence of consumers in adopting cloud services [10] and it has become an essential influence factor when consumers have to interact with unfamiliar providers in selecting and accomplishing a service.

In computer science, it was Marsh that first introduced trust into the distributed artificial intelligence community as a computable concept and formalized it using mathematics method [11]. Since then, trust has been regarded as a measurable belief and can be modeled so that it can be used to help a trustor to judge whether a trustee is trustworthy in order to make a decision. Nowadays, trust has become a necessary security relationship in distributed network environment and a solution to security issues. In recent years, various trust models and trust management systems are proposed in different application scenarios such as electronic commerce [11, 12], ubiquitous computing [13], Mobile Ad-Hoc Network [14], P2P networks [15, 16], the Internet of things [17], cloud computing [10, 18] and so on.

In order to help CSCs to cope with the challenge of trust issue in cloud computing, objective and effective technical mechanisms are essential. It is of real significance to evaluate cloud service trustworthiness to deal with trust challenge that CSCs have to confront in diverse cloud service marketplaces. Although the concept of cloud service trustworthiness has been proposed and several evaluation solutions have been discussed, the related researches are still not sufficient [3]. In particular, there is a growing demand to assist the non-expert customers to choose the trustworthy cloud service [19]. In fact, the evaluation result of cloud service trustworthiness such as reputation score or ranking can become an objective and concise reference for cloud service consumers, enterprise providers, and administrators in decision-making. For consumers, it can help them to select and purchase satisfactory cloud service. For enterprise providers, it can provide them with a reference on whether their service quality is welcome and which aspects need to be improved. For administrators, it can offer them a guide on which enterprises they should pay more attention in supervision. Therefore, cloud service trustworthiness evaluation can be regarded as one of the important way to solve trust issue in cloud computing.

TOPSIS (Technique for Order Preference by Similarity to an Ideal Solution) has received much interest from researchers and practitioners as a popular Multi-Criteria Decision Making (MCDM) method [20]. To the best of our knowledge, there is no straightforward application of TOPSIS method in evaluating cloud service trustworthiness. In this paper, we develop the common QoS (Quality of Service) metrics for cloud service trustworthiness based on existing research work. Most important of all, we put forward a novel evaluation scheme for cloud service trustworthiness based on TOPSIS method.

The rest of this paper is organized as follows. In Section 2, we describe the background of our research. In Section 3, we investigate the related research work. In Section 4, we propose the common QoS attributes of cloud services for trustworthiness evaluation and provide quantitative descriptions. In Section 5, we first outline the general idea and CSTE evaluation model in this study, and then depict the proposed TOPSIS evaluation scheme in detail. In Section 6, we perform two experiments based on the existing QWS Web service dataset and make detailed analysis in effectiveness, feasibility and satisfaction degree respectively and make detailed analysis. In Section 7, we make a brief overview of our work and future research directions.

2. Background

In this section, we first investigate the basic concepts of trust, trustworthiness and their implications in our study. We then outline TOPSIS method and analyze its advantages and feasibility in evaluating cloud service trustworthiness.

2.1. Trust and trustworthiness

Trust is a multidisciplinary concept that can be studied from multiple perspectives, including sociology, psychology, economics, etc [14]. There are various definitions and understandings of trust in different domains. Jøsang et al. [21] define trust as the extent to which one party is willing to depend on something or somebody in a given situation with a feeling of relative security, even though negative consequences are possible. Adali et al. [22, 23] regard trust as a relationship between a trustor and a trustee, which indicates the trustor's willingness to be vulnerable under conditions of risk and interdependence. These two definitions demonstrate the subjective willingness of a trustor and uncertain result included in trust. One of the most widely accepted definitions on trust originates from the sociologist Gambetta [24]: trust is a particular level of the subjective probability with which an agent assesses that another agent or group of agents will perform a particular action. This definition implies that trust is a relationship between two parties involved in a specific cooperation and has a certain effect on the success of transaction. Based on the different understandings of trust mentioned above, we can conclude that trust is essentially a multi-dimensional concept with subjectivity, context-dependence, and uncertainty, including various aspects of competence and intent.

Trustworthiness is another concept accompanied with trust. According to Stanford Encyclopedia of Philosophy (`http://plato.stanford.edu/entries/trust/`), trust is an attitude that we have towards people whom we hope will be trustworthy, whereas trustworthiness is a property, not an attitude. Ideally, those whom we trust will be trustworthy, and those who are trustworthy will be trusted. This understanding reveals the relationship between trust and trustworthiness. It shows that trust is a subjective attitude of a trustor, while trustworthiness is an objective property of a trustee. However, this opinion confines trust and trustworthiness to a limited range of human society, not concerns online environment. Although the ideal situation that trust is equivalent to trustworthiness is not indeed the case under normal circumstances due to the subjectivity of trust, it provides us with a good reference and general idea to measure the trust through trustworthiness.

In practice, trustworthiness can be evaluated by both objective measurement and subjective perception [3]. The trustworthiness of cloud services depends on two facets of trust including objective trust and subjective trust [10]. It is necessary to combine both subjective dimension (e.g. user's feedback) and objective dimension (e.g. QoS performance monitoring) when evaluating the trustworthiness of Web services [25]. Based on these viewpoints, we evaluate cloud service trustworthiness from objective and subjective aspects in terms of different information sources. The former derives from objective measurement of facts or reliable evidences such as quality of service. The latter refers to the individual preference such as the need or judge from human beings. In fact, subjective aspects can also be regarded as direct expression of trust subjectivity. Objective measurement and subjective preference together constitute an overall trustworthiness, which is numerically equivalent to trust. Cloud service trustworthiness discussed in this study refers to this kind of overall trustworthiness.

2.2. TOPSIS method

TOPSIS, short for Technique for Order Preference by Similarity to an Ideal Solution, first developed by Hwang and Yoon in 1981 [20], is one of the classical MCDM methods. In essence, TOPSIS is a ranking method that can evaluate multiple objects with same attributes. It can help us to make decisions by means of figuring out final score for each candidate object with same structure but different parameter values. Among numerous MCDM methods developed to solve real decision problems, TOPSIS continues to work satisfactorily in various application areas [20]. It is welcome in practical applications for its special advantages such as simple, understandable, quick and reliable [26].

The basic idea of TOPSIS method [20, 26] is that the most desired alternative should simultaneously have the shortest distance from the positive ideal solution and the farthest distance from the negative ideal solution. The positive ideal solution is equal to a solution with the best value for each attribute by maximizing the benefit criteria and minimizing the cost criteria. On the contrary, the negative ideal solution is equal to a solution with the worst value for each attribute by minimizing the benefit criteria and maximizing the cost criteria. In fact, these two extreme solutions simply serve as a reference for comparison and may not really exist in the set of all alternatives. This may be the reason why they get their name.

It is reasonable and advantageous to evaluate cloud service trustworthiness employing TOPSIS method. For one thing, since trustworthiness evaluation usually concerns multiple attributes, many investigations identify it as a MCDM problem [3]. For another thing, TOPSIS is one of the basic MCDM methods and has unique advantages mentioned above. Most important of all, compared to other MCDM methods, it can make better use of various quantitative attribute information [20]. In addition, TOPSIS has been proved to be an effective way to evaluate quality credit because of its benefits [26]. In fact, there is a high similarity between the quality credit evaluation and cloud service trustworthiness evaluation. Although a variety of trust models have been proposed, to the best of our knowledge, TOPSIS method has not been directly applied in evaluating cloud service trustworthiness. Based on the above analysis, we determine to apply TOPSIS method to develop a novel evaluation scheme for cloud service trustworthiness.

3. Related work

The main contributions of our proposed evaluation scheme for cloud service trustworthiness are based on some existing representative research work. In this section, we first introduce the necessity of QoS in this study, then review the typical trust models based on QoS in cloud computing, and finally outline the difference of our proposed evaluation scheme.

With the growth of cloud service offerings, multiple CSPs often provide similar cloud services with same functionality. In this case, it is obviously not sufficient for CSCs to distinguish different services just according to their function. A practical way to solve this problem is to utilize QoS of cloud services. In the early stage, QoS attributes are often employed to depict non-functional attributes of Web services [27]. Since Web service is the most common realization technology of cloud service in cloud computing environment [4], QoS attributes can be regarded as the basic parameters of cloud service trustworthiness evaluation. In practice, evaluating QoS and ranking CSPs is one of the important challenges to be tackled when building cloud computing services evaluation models [28].

Existing trust and reputation systems largely depend on customer feedback while ignoring other sources and roots of information [29]. Most systems rely on ratings provided by consumers and this causes many issues involving the subjectivity and injustice of the service ratings [30]. Since reputation based on past experience has been used as a mechanism of addressing the issue of trust [31], feedback ratings are often used as the information source to determine the trustworthiness in reputation-based systems. For example, Noor et al. [18] design and implement a reputation-based trust management framework employing trust feedback. Different from regular research, this framework designs a novel protocol to prove the reliability of feedback information. Su et al. [32] presents a priority-based trust model for service-oriented systems. The proposed model successfully considers the preferred priority of a new consumer to each QoS attribute when evaluating the trustworthiness of services, selecting the referenced ratings by calculating the similarity of priorities between the requested service and referenced services in order to increase the satisfaction degree. However, the evaluation results largely depend on the ratings of referenced consumers and neglect the reliability of information source.

Since trustworthiness evaluation is multi-dimensional and concerns both objective and subjective aspects, how to assign weight for each influence factor is a non-trivial problem. Rajendran et al. [10] propose a hybrid model for dynamic trust evaluation of cloud services combining the objective and subjective aspects, however, the weights in the model are completely assigned subjectively when performing experiments. Li et al. [33] proposes a trust model using an aggregation of the trust degree based on the time series, in which the weights are subjectively assigned. Ding et al. [3] propose a framework combining QoS prediction and customer satisfaction estimation to evaluate cloud service trustworthiness. Although this framework consists of both objective QoS information and subjective satisfaction perception, it still lacks adaptability in weight assignment for trust attributes. In addition, it mainly focuses on the verification of QoS prediction method and ignores the empirical analysis of the final trustworthiness evaluation results. Alhamed et al. [34] develops a model for IaaS (Infrastructure as a Service) using the Sugeno fuzzy-inference approach to measure the trust value of cloud providers in e-learning application. The model determines the weight of each factor by consumers or equally assigned by the system. In conclusion, while the above trust models succeeded in solving real problems to some extent, they are confronting one common shortcoming in practice: how to assign the weight reasonably among multiple QoS attributes considering both objective and subjective aspects.

To make up the deficiencies of some existing work and inspired by a couple of related research, we introduce the combined weight and put forward a novel TOPSIS evaluation scheme for cloud service trustworthiness combining subjective and objective aspects. Our work is different from existing trust schemes in several ways. On the one hand, we take into account the reliability of QoS information source and employ monitored QoS attributes values (i.e. QoS parameters) instead of feedback ratings from consumers. On the other hand, we employ entropy weight method to assign weights for different QoS attributes. Entropy weight is an objective method in weight assignment and it can decrease the effect of false or artificial QoS parameters information. Furthermore, we also integrate individual trust preference information into the TOPSIS evaluation scheme. Due to the subjectivity nature of trust, it is of importance to take into account the subjective preference and requirements of CSCs when evaluating the trustworthiness of a CSP [35]. The proposed scheme in this study achieves the combination of objective and subjective aspects so that the evaluation results can better satisfy the trust requirements of different CSCs.

4. Common QoS attributes for cloud services

In order to define a trust metric, we need to determine the trust parameters that can be used to assess the trust first [36]. It is the same with cloud service trustworthiness. In cloud computing environment, cloud service trustworthiness can be regarded as a combined quality measure of cloud services [3]. It largely depends on QoS attributes in objective aspects.

In cloud computing environment, cloud service trustworthiness can be regarded as the combined quality measure of cloud services [3]. QoS has been regarded as a necessary way to describe the non-functional features of Web services. W3C group first issued a draft on QoS requirements for Web Services in November 2003 (`http://www.w3c.or.kr/kr-office/TR/2003/ws-qos`). Yao Wang et al. [37] list the QoS metrics for Web services that are needed for a trust and reputation mechanism. As Web service is an important realization technology in cloud computing [4], many trust models employ these QoS attributes as the standard to determine whether a cloud service is trustworthy. Garg et al. [28] present and quantify the metrics for their proposed SMICloud framework that help users to select services on IaaS platform.

QoS attributes can be classified into two types: quantitative attributes and qualitative attributes. Quantitative attributes are those that can be measured directly employing software and hardware monitoring tools while qualitative ones are those mostly inferred based on user experiences [28]. Cloud service trustworthiness analysis largely concentrates on quantitative measurement relying on a full assessment dataset [3]. To the best of our knowledge, most of existing literatures build their trust evaluation models just according to several QoS attributes. Obviously, it is not sufficient to build a trustworthiness evaluation model simply in terms of several quantitative QoS attributes.

In order to solve cloud service trustworthiness metrics in objective aspects and simplify the evaluation process, we concentrate on the common QoS attributes concerning both quantitative and qualitative ones and their quantification. Based on the above existing work, we present the common attributes for cloud service trustworthiness evaluation. Table 1 lists these attributes of cloud services including their detailed meanings and quantification descriptions. It includes basic mathematical operators in the column of Quantification descriptions to make it clear. As shown in the last column of Table 1, quantitative QoS attributes can be divided into two categories [3], i.e. positive attributes and negative attributes, which will be clarified in the later section.

5. TOPSIS evaluation scheme for cloud service trustworthiness

The aim of this study is to propose a novel evaluation scheme for cloud service trustworthiness from the perspective of CSCs. The major content of the proposed methodology involves: (a) basic idea on trustworthiness evaluation; (b) a CSTE (cloud service trustworthiness evaluation) model; (c) model solution. The focus is on the detailed steps of the novel TOPSIS evaluation scheme combining objective entropy weight and subjective preference weight of a CSC. Details of each part are described in the following sections.

5.1. Basic idea

In essence, cloud service trustworthiness is equivalent to the trust a CSC holds for the quality of a cloud service and it concerns both objective and subjective aspects. Objective aspects

Table 1
Common QoS attributes of cloud services

Attributes	Meanings/Definitions	Quantification descriptions	Positive/ Negative
Response time	Response time is the time taken by a service between being requested and being responded to.	the time a service is responded − the time the service is requested	Negative
Throughput	Throughput is the number of service requests served in a given time interval.	the number of service requests served / total service time	Positive
Reliability	Reliability represents the ability of a service to perform its required functions under stated conditions for a specified time interval.	the number(or time) of failures / the mean number (or time) of failures promised by a provider in a given time interval	Positive
Scalability	Scalability represents the capability of increasing the computing capacity of service provider's computer system and system's ability to process more users' requests, operations or transactions in a given time interval.	maximum available increase capacity / a given time interval	Positive
Availability	Availability is the probability that the system is up.	the time of a service is available / total service time	Positive
Accuracy	Accuracy is defined as the correct rate generated by the service.	the number (or time) that a service satisfies promised values / total number (or time) of the service in a given time interval	Positive
Interoperability	Services should be interoperable between different development environments used to implement services.	the number of resources offered by the provider / the number of resources required by users for interoperability	Positive

mainly derive from the reliable QoS attributes information of cloud services. Subjective aspects concern various trust preferences of CSCs to different QoS attributes. These two aspects together constitute the overall trustworthiness.

In order to ensure the objectivity of QoS information source and avoid intentional malicious tampering, we take two ways to achieve the purpose. On the one hand, we employ QoS attributes information derived from a reliable third-party monitor procedure instead of pure feedback ratings of consumers. On the other hand, to further enhance the objectivity of evaluation results, we apply entropy weight into TOPSIS method to assign weights for different QoS attributes, instead of classical average weight or simply subjective designated weight. Therefore, our proposed scheme can better focus on the objective evaluation aspects.

Apparently, it is still not enough to evaluate cloud service trustworthiness if we just take into account the objective aspects. Since trust is a subjective concept, we should also consider the individual trust preference of each CSC to different QoS attributes. It is necessary to obtain high satisfaction degree of CSCs. In our scheme, we introduce the concept of trust preference to describe the different requirements for various QoS attributes and formalize it into a weight vector, which is called trust preference vector in our study. For example, suppose the number of total QoS attributes is five, the trust preference vector of a particular CSC may look like this: $(0.2, 0.3, 0.3, 0.1, 0.1)$. The sum of each weight to each QoS attribute for a single cloud service is equal to 1.

In practice, how to combine the subjective trust preference of a certain consumer with objective QoS information is a very significant issue [35]. To solve this problem, we introduce another concept of the combined weight in our proposed scheme. Its application is just the novelty and main task of our proposed TOPSIS evaluation scheme for cloud service trustworthiness.

5.2. CSTE model

Many cloud service trustworthy attributes can be measured in terms of QoS values[3]. At present, the most common way to evaluate trustworthiness is to rank or make recommendations [3, 10, 28, 38]. Inspired by the existing work and based on the common QoS attributes of cloud services in Table 1, we present a model for cloud service trustworthiness evaluation (abbreviated as CSTE) using TOPSIS method from the perspective of CSCs, as shown in Figure 1.

Applying CSTE model to evaluate cloud service trustworthiness consists of three main phases. First, collect and preprocess all QoS parameters of each cloud service from reliable third-party monitor procedure. Second, calculate trustworthiness for each candidate cloud service employing the proposed TOPSIS evaluation scheme in this paper. Third, rank all the candidate cloud services in terms of trustworthiness evaluation results and provide the requesting CSC with ranking results or a list of Top-K trustworthy services.

The QoS parameters acquired from monitor procedure can dramatically affect the reputation of a certain cloud service positively or negatively [39], whereas the reputation of cloud service influences its trustworthiness directly. In order to keep the objectivity and reliability as possible, when calculating the trustworthiness of each cloud service, we suppose the needed QoS attributes information for each cloud service come from reliable third-party monitor procedure. Related research work on reliable QoS information collection have been done such as Limam's work in [30] and Noor's work in [18]. In addition, many performance monitoring and analysis tools on clouds are proposed in related work such as [40]. Tools like CopperEgg, Cedexis and Anturis are used in service monitoring [10]. Based on the above investigations, collecting reliable QoS information for each cloud service in phase 1 is beyond our work in this study. The rest work in phase 2 and 3 will be described in the following sections respectively.

5.3. TOPSIS method for calculating cloud service trustworthiness

In this section, we develop a novel TOPSIS evaluation scheme gradually to calculate the trustworthiness of each cloud service.

Step 0: establishing matrices for CSCs and QoS attributes of each cloud service

In this step, we build matrices for CSCs and QoS attributes of each candidate cloud service and preprocess data. Take a certain cloud service s_k as an example, we build its CSC-QoS matrix

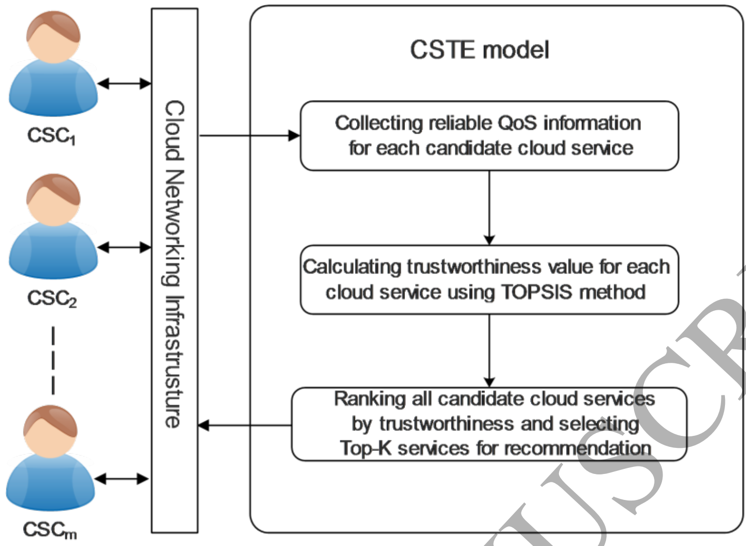


Fig. 1. CSTE (Cloud Service Trustworthiness Evaluation) model

M_{s_k} as shown in Fig.2. According to the basic idea, a third-party monitor procedure is employed to collect QoS attributes information at regular times after CSCs perform a cloud service. When we acquire all the collected information from the monitor, we need to preprocess data beforehand in order to apply entropy weight method in next steps. Here we calculate the average values of each QoS attribute according to the number of CSCs for every candidate cloud service. Suppose there are m similar cloud services provided by different CSPs and every service has n attributes, s_k is the kth cloud service of m cloud services, n_k is the number of CSCs using the specific cloud service s_k, where $1 \leq k \leq m$. In the CSC-QoS matrix, each row represents a CSC, each column denotes a certain attribute of the invoked service s_k. An item c_{ij} in the matrix M_{s_k} denotes a value of the jth QoS attribute (a_j) acquired from the ith CSC (c_i) for the cloud service s_k. Let s_{ij} be the average value of the attribute a_j of a certain cloud service s_i, then it can be calculated according to formula (1):

$$s_{ij} = \frac{\sum_{k=1}^{n_i} c_{kj}}{n_i}, \qquad (i = 1, 2, ..., m; j = 1, 2..., n) \tag{1}$$

Step 1: establishing a matrix of cloud services and QoS attributes

In this step, we establish a service-attribute matrix M_1 of cloud services and their QoS attributes for all alternatives with the same functionality. As shown in Fig.3, where each row represents a certain cloud service, and each column represents a common QoS attribute of all candidate cloud services, an item in M_1 means an average value of a certain QoS attribute of a cloud service. Let s_i $(1 \leq i \leq m)$ be the ith cloud service to be evaluated and a_j $(1 \leq j \leq n)$ be the jth attribute belonging to the service s_i, then the item s_{ij} in M_1 denotes the average value of the jth attribute (a_j) of the ith cloud service (s_i) according to the number of CSCs. It can be calculated in terms of formula (1). In fact, we can calculate all the entries in M_1 based on the above Step 0 and formula (1).

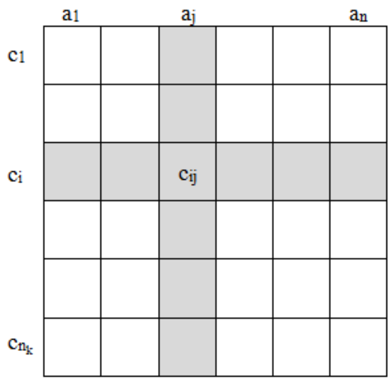


Fig. 2. The matrix M_{s_k} of CSCs and QoS attributes for a certain cloud service s_k

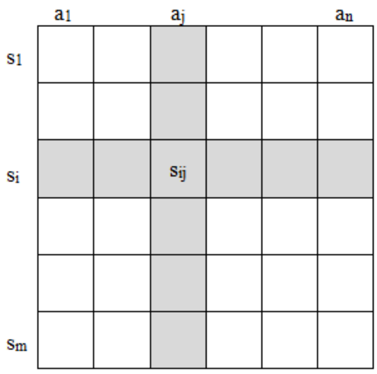


Fig. 3. The matrix M_1 of cloud services and their QoS attributes values

Step 2: normalizing the matrix of cloud services and QoS attributes

In this step, we need to normalize the QoS attributes values in the matrix M_1 derived from Step 1. Due to the heterogeneity of QoS attributes for all candidate cloud services, their values may have different measurement units, ranges, and even opposite meanings, which will cause inconsistency in comparison. A necessary approach to solve this problem is to use normalization.

Although TOPSIS method allows us to use any normalization algorithm to normalize the raw data, some comparative studies of different normalization procedures for TOPSIS have been done [41, 42, 43]. All these studies conclude that vector normalization is the best choice compared to any linear normalization. For instance, Chakraborty et al. [41] conclude that the use of vector normalization in TOPSIS method is justified by their experiment results, it is the most consistent among the common normalization procedures in ranking and can handle weight sensitivity quite well. Çelen [42] also reveals that vector normalization procedure generates the most consistent results which is in line with the conclusion in [41]. Vafaei et al. [43] compared four sum-based normalization techniques for TOPSIS method and found one of the best is vector normalization. Inspired by the above research work, we employ vector normalization algorithm in our TOPSIS evaluation scheme.

Before we normalize the matrix M_1 shown in Fig.3, we need to distinguish QoS attributes into two categories i.e. positive attributes and negative attributes [3], as mentioned in Section 4. For positive attributes, the larger the QoS attribute value is, the better the quality of the corresponding cloud service is. For example, a cloud service can acquire a good reputation for its better quality deriving from larger Throughput or higher Availability. But with negative attributes, the opposite is true. The smaller the value of a negative attribute is, the better the corresponding cloud service is. Take Response Time as an example, a cloud service with a smaller value of Response Time are usually considered to have better quality and hence it can acquire a good reputation. In order to eliminate the inconsistency between positive attributes and negative attributes, we need to separately normalize them in terms of vector normalization algorithm.

Now we normalize the positive attributes in M_1 of Step 1 according to formula (2):

$$ns_{ij} = \frac{s_{ij}}{\sqrt{\sum_{i=1}^{m}(s_{ij})^2}}, \qquad (i = 1, 2, ..., m;\ j = 1, 2..., n) \tag{2}$$

Next we need to normalize the negative attributes in M_1 of Step 1 according to formula (3):

$$ns_{ij} = \frac{\dfrac{1}{s_{ij}}}{\sqrt{\sum_{i=1}^{m}\dfrac{1}{(s_{ij})^2}}}, \qquad (i = 1, 2, ..., m;\ j = 1, 2..., n) \tag{3}$$

After this step, we get a new normalized matrix, denoted as M_2. Now each entry in M_2 has the same metric scale and belongs to $[0, 1]$.

Step 3: determining the combined weight of each QoS attribute

The combined weight in this study includes two parts: objective weight and subjective weight. How to acquire the two parts to form the combined weight is significant for further application of TOPSIS.

To differentiate the importance degree of various QoS attributes for all candidate cloud services, we introduce entropy in this study. Generally, entropy is a quantified way of uncertainty in terms of probability theory. It refers to the average amount of information generated by a stochastic source of data in information theory [44]. The data source with lower probability values is considered to carry more information than that with higher probability values. That is to say, the former data are more important than the latter data and deserve higher weight. Based on this rationale, we can conclude that the bigger the difference between the values of a quality attribute, the more useful this information is to select between different cloud services. Therefore, we employ entropy to differentiate the importance degree between different QoS attributes and transform it into entropy weight directly for ease of use.

Subjective weight derives from the individual trust preference of a CSC. Every CSC may have different priority to each attribute, i.e. weight. The weight values of all the attributes from a certain CSC constitute a vector, i.e. trust preference vector mentioned above. It can vary greatly among different CSCs. For example, a CSC with strict demand for time can give a higher weight to Response Time than another CSC who cares more about Reliability. It is possible to occur that two trust preference vectors from them look like this respectively: $(0.6, 0.1, 0.1, 0.1, 0.1)$, $(0.1, 0.6, 0.1, 0.1, 0.1)$, if there are five QoS attributes in total, and they are Response Time, Reliability, Throughput, Availability and Accuracy in sequence.

In this step, we first calculate objective entropy weight according to the following substeps $(1) - (4)$. Then we determine subjective weight from trust preference according to the substep (5).

Finally, we acquire the combined weight which can be used in TOPSIS method straightforward according to the substep (6).

(1) Calculating the characteristic proportion of cloud services

For a common QoS attribute of all m candidate cloud services, its m values may vary between two extreme situations: totally same and totally different. This can spell different probability levels among various values for the certain attribute. Here we use characteristic proportion to denote the ratio of one attribute value to the sum of all values for a certain attribute. Here characteristic proportion is equivalent to relative frequency [45]. It implies the probability of this attribute value from a certain cloud service in all candidate cloud services. Let p_{ij} be the characteristic proportion of the jth QoS attribute of the ith cloud service s_i, then we can calculate it according to formula (4):

$$p_{ij} = \frac{ns_{ij}}{\sum_{i=1}^{m} ns_{ij}}, \qquad (i = 1, 2, ..., m;\ j = 1, 2..., n) \tag{4}$$

Since ns_{ij} belongs to [0, 1], the range of p_{ij} is [0, 1].

(2) Calculating the entropy value for each QoS attribute

As can be concluded from the above, the probabilistic nature of entropy implies that the bigger the variation among values of a certain QoS attribute for all candidate cloud services is, the higher this attribute should be weighted than others. How to define entropy and apply it into TOPSIS method is a key step. The measure of information entropy associated with each possible data value is the negative logarithm of the probability mass function for the value [44]. Inspired by the definition of information entropy and based on the result in substep (1), then we can define and calculate entropy for each QoS attribute according to formula (5):

$$e_j = \frac{-1}{ln(m)} \sum_{i=1}^{m} p_{ij} \cdot ln(p_{ij}), \qquad (j = 1, 2..., n) \tag{5}$$

In the above formula (5), e_j denotes the entropy value of the jth QoS attribute for all services. The range of e_j is [0, 1]. For a certain QoS attribute a_j, the greater the difference among its values ns_{ij} for different i is, the smaller its entropy value e_j is. According to information theory, for a certain QoS attribute, the smaller its entropy value is (i.e. the more diverse values are), the larger amount of information reflected by this attribute. On the contrary, when the entropy value of a certain QoS attribute is smaller, the amount of information reflected by this attribute is larger.

(3) Calculating the difference coefficient for each QoS attribute

For ease of understanding and distinction, we introduce another concept named as difference coefficient. Just as its literal meaning suggests, difference coefficient reflects the variation degree among the values of a certain QoS attribute. Let d_j indicate the difference coefficient of the jth QoS attribute, then we define d_j according to formula (6):

$$d_j = 1 - e_j, \qquad (j = 1, 2..., n) \tag{6}$$

It is evident that the larger the value of d_j is, the larger the variation among values of the jth QoS attribute a_j is, i.e. the larger the amount of information offered by this attribute is. In other words, it means the jth QoS attribute a_j is more objective and valuable than others when distinguishing different cloud services.

(4) Calculating the entropy weight for each QoS attribute

In this step, we determine the entropy weight for each QoS attribute. In fact, entropy weight is derived from the entropy and it is employed to derive the objective weight of each QoS attribute for all candidate cloud services in this study. It is very helpful in ranking all candidate cloud services. Let ew_j be the entropy weight of the jth QoS attribute, we can calculate w_j according to formula (7):

$$ew_j = \frac{d_j}{\sum_{j=1}^{n} d_j}, \qquad (j = 1, 2..., n) \tag{7}$$

where n is the number of all QoS attributes.

Compared with the weighting method determined subjectively by experts, it is not difficult to understand that entropy weight is an objective weighting method. Next we discuss the subjective weight.

(5) Determining subjective weight from trust preference vector

As mentioned above, trust preference reflects different subjective trust requirement of requesting CSCs to various QoS attribute. It is denoted by an n-dimensional trust preference vector, here n is equal to the number of QoS attributes. Each entry in this vector determines the CSC's priority to each of the attributes, i.e. subjective weight. The sum of each entry in this vector is 1. In practice, there are many ways and tools available to capture customer requirements and preferences such as questionnaire, individual inquiring, expert interviews or surveys [46]. We propose to employ individual inquiring when implementing the evaluation scheme in practice, integrating a pair-wise comparison approach [47, 48] to gather the raw data and then derive the trust preference vector based on eigenvector method. The basic process is shown in Fig.4.

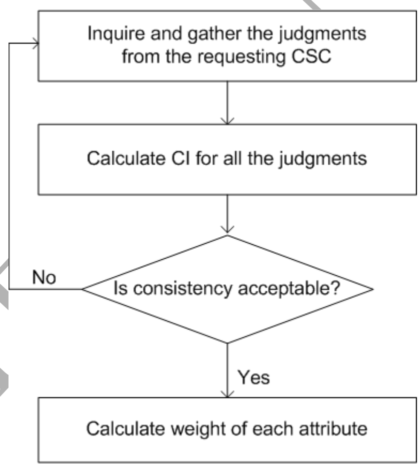


Fig. 4. The process of obtaining a trust preference vector

According to the process, we first ask the requesting CSC to provide judgments by comparison between every two QoS attributes for all n attributes of the same kind of cloud services in terms of the designed scale as the same as in Table 1 of Reference [47]. Then we can get the pair-wise comparison matrix A in equation (8), in which the element a_{ij} of the matrix is the relative importance of the ith attribute with respect to the jth attribute. Reciprocals are automatically assigned to each element. Next is to calculate CI (Consistency Index) [48] after calculating the maximum eigenvalue based on eigenvector method. Since judgments in matrix A may not be consistent, we need to make consistency test. If the matrix A is consistent, then the eigenvector with maximum eigenvalue is regarded as the trust preference vector. If not, judgments forming the raw data should be improved by repeating the inquiring and gathering. The corresponding formulas can refer to Reference [48].

$$A = (a_{ij})_{n\times n} = \begin{bmatrix} 1 & a_{12} & \cdots & a_{1n} \\ \dfrac{1}{a_{12}} & 1 & \cdots & a_{2n} \\ \vdots & \vdots & \cdots & \vdots \\ \dfrac{1}{a_{1n}} & \dfrac{1}{a_{2n}} & \cdots & 1 \end{bmatrix} \tag{8}$$

(6) Calculating the combined weight for each QoS attribute

How to combine objective and subjective aspects when evaluating cloud service trustworthiness is an essential challenge. To solve this problem, we introduce the combined weight integrating objective and subjective aspects into TOPSIS method. It is calculated in terms of formula (9):

$$cw_j = ew_j \times sw_j\ , \qquad (j = 1, 2..., n) \tag{9}$$

where cw_j represents the combined weight, it is the product of objective entropy weight ew_j and subjective weight sw_j from a requesting CSC to different QoS attributes.

Step 4: weighting the normalized matrix of cloud services and QoS attributes

Now we weight the normalized matrix M_2 and get a new matrix M_3 according to formula (10):

$$ws_{ij} = cw_j \times ns_{ij}\ , \qquad (i = 1, 2, ..., m;\ j = 1, 2..., n) \tag{10}$$

where ws_{ij} denotes the weighted normalized value of the jth QoS attribute of the ith cloud service, cw_j is the combined weight, and ns_{ij} is the normalized QoS attribute value of the jth QoS attribute for the ith cloud service in Step 3.

Step 5: determining the positive ideal solution and the negative ideal solution

$$\begin{aligned} ws^+ &= \{ws_j^+,\ j = 1, 2, ..., n\} \\ ws^- &= \{ws_j^-,\ j = 1, 2, ..., n\} \end{aligned} \tag{11}$$

where ws^+ and ws^- in formula (11) denote the positive ideal solution and the negative ideal solution, respectively. $ws_j^+ = \max\{ws_{ij}, i = 1, 2, ..., m\}$, representing the best value of the jth QoS attribute in all m cloud services, whereas $ws_j^- = \min\{ws_{ij}, i = 1, 2, ..., m\}$, representing the worst value of the jth QoS attribute in all m cloud services. Based on the previous steps and definitions, we can see that ws^+ refers to the ideal cloud service with all the best values of QoS attributes, whereas ws^- refers to the worst one with all the worst values of QoS attributes. In fact, the two services that ws^+ and ws^- refer to may not exist in the real list of cloud services.

In order to avoid the inverse order problem of TOPSIS, we can adopt absolute positive ideal solution and negative ideal solution, respectively. That is to say, we can specify fixed value for positive ideal solution and negative ideal solution respectively according to the actual situation instead of maximum and minimum values when updating candidate cloud services every time.

Step 6: calculating the distance from each cloud service to the positive ideal solution and the negative ideal solution respectively

$$
\begin{aligned}
ds_i^+ &= \sqrt{\sum_{j=1}^{n}(ws_{ij} - ws_j^+)^2}\,, \qquad (j = 1, 2, ..., n) \\
ds_i^- &= \sqrt{\sum_{j=1}^{n}(ws_{ij} - ws_j^-)^2}\,, \qquad (j = 1, 2, ..., n)
\end{aligned}
\tag{12}
$$

where ds_i^+ in formula (12) represents the distance between the ith cloud service and the positive ideal solution, while ds_i^- represents the distance between the ith cloud service and the negative ideal solution.

Step 7: calculating the trustworthiness score of each cloud service

Based on the above steps, now we can calculate the closeness degree to the positive ideal solution and the negative ideal solution for each cloud service, which is called cloud service trustworthiness in this study. Let T_i be the trustworthiness value of the ith cloud service s_i, then we can calculate T_i according to formula (13):

$$
T_i = \frac{ds_i^-}{ds_i^+ + ds_i^-}\,, \qquad (i = 1, 2, ..., m) \tag{13}
$$

where ds_i^+ and ds_i^- derive from the calculation results in terms of formula (12) in Step 6. It is obvious that the higher the value T_i of a cloud service is, the more trustworthy this service is.

5.4. Ranking candidate cloud services

After calculating the trustworthiness evaluation results of each cloud service according to Section 5.3, we can rank all the candidate cloud services in terms of their trustworthiness values and list Top-k cloud services for the requesting CSC. Here k can be determined by the CSC, or be set a default value such as 10.

In conclusion, our proposed CSTE model utilizes a novel TOPSIS evaluation scheme to calculate the trustworthiness value of cloud services in terms of QoS attributes information deriving from reliable source. In our model, we not only propose but also quantify the most common cloud service QoS attributes adopted by most trust models concerning quantitative and qualitative types. The most important of all, we adapt the way of weight assignment in classical TOPSIS method and change it into a new hybrid type incorporate not only the objective entropy weight that can decrease the subjectivity of QoS information, but also the subjective weight from a CSC's trust preference.

6. Experiments and analysis

In this section, we first depict the experimental objectives and data used in the experiments. Then we perform experiments to verify the proposed evaluation scheme. Experiment 1 mainly illustrates the validity of TOPSIS method using entropy weight. Experiment 2 shows the advantage of introducing subjective trust preference. The results of two experiments together demonstrate the feasibility, effectiveness and benefit of the proposed evaluation scheme.

6.1. Experimental objectives and data description

The proposed evaluation scheme can be applied on a set of services for which no assessment of quality attributes is available, which can also be considered as the reason to introduce it in cloud service marketplaces. To find out how trustworthy and satisfactory a cloud service is can help CSCs to determine whether it is reliable to interact directly with the corresponding CSP. In this set of simulation experiments, our aim is to research the feasibility, effectiveness and benefit of the proposed scheme. To achieve these goals, we perform two experiments and simulate the proposed scheme based on the existing research work. Experiment 1 is chiefly designed to verify the feasibility and effectiveness of TOPSIS method employing entropy weight. It also presents the application process of our proposed scheme in evaluating cloud service trustworthiness. Experiment 2 concerns a case study on the concept of Satisfaction Degree based on the existing work. It is to clarify the benefit of combining the subjective trust preference into the trustworthiness evaluation scheme, by comparing satisfaction degrees before and after using subjective trust preference.

Since no common QoS dataset on cloud service is available nowadays and most cloud services are Web services in essence, we choose the QWS (Quality of Web Service) dataset [49, 50] to perform our experiments. The QWS dataset consists of various Web services collected using Web Service Crawler Engine (WSCE) from the real Web. These Web services were obtained from public sources on the Web including Universal Description, Discovery, and Integration (UDDI) registries, search engines, and service portals. In order to make comparison, we employ version 1.0 of QWS dataset in our experiments since only this version of dataset provides the ratings and classification. The public dataset of version 1.0 consists of 364 Web services each with a set of nine quality attributes that have been measured using commercial benchmark tools. Each service was tested over a ten-minute period for three consecutive days by the creators of the QWS dataset.

6.2. Experiment 1: verifying the feasibility, effectiveness of the proposed evaluation scheme

In order to facilitate the following experiments, we first implement an automated tool with a simple graphic interface for the evaluation process using Tkinter in Python 2.7.5. The operating system environment is Microsoft Windows 7 (32 bit), Service Pack 1. Using this tool, after we import the experimental data to be evaluated and specify the trust preference vector, we can click the Evaluate button to acquire the desired evaluation results, which can be saved as a file by clicking the Save button. The results of an application example are shown in Fig.5.

Based on the automated evaluation tool, we perform two kinds of experiments. The first kind of experiments is to evaluate the services with same functionality, which are searched in terms of keywords from the corresponding website. The second kind of experiments are based on services acquired by random sampling to extend the scope of samples. Finally, the evaluation results from these two kinds of experiments are verified by Kendall's Tau rank correlation and its significance test.

6.2.1. The first kind of experiments: evaluating the searched services with similar functionality

For this kind of experiments, we first choose 10 groups of services with same functionality from the QWS dataset (version 1.0) for our case study by utilizing the search function provided by the related website (`http://www.uoguelph.ca/~qmahmoud/qws/index.html#demo`), then evaluate these groups of services applying our proposed scheme respectively. During

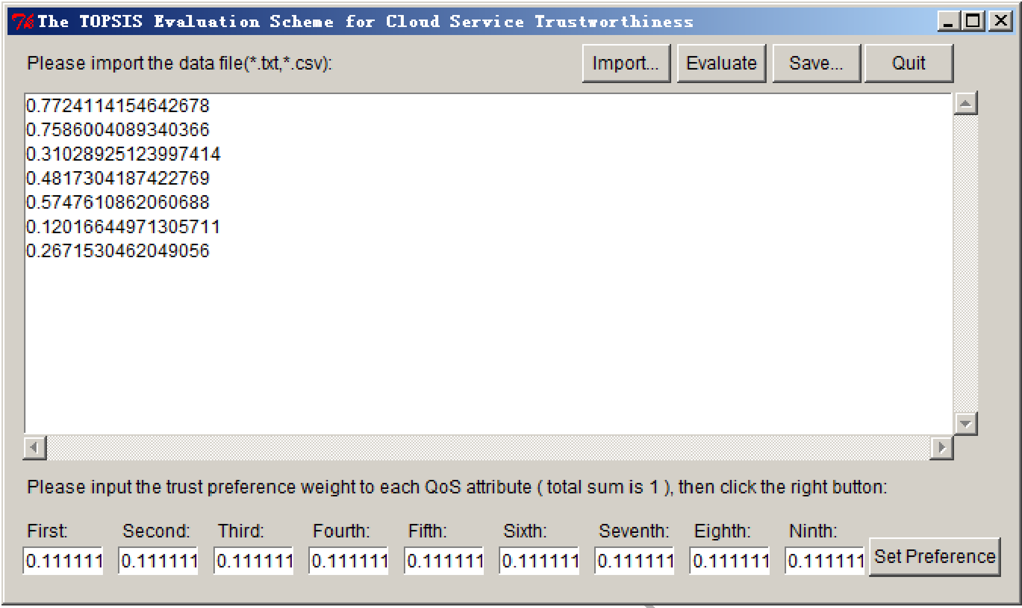


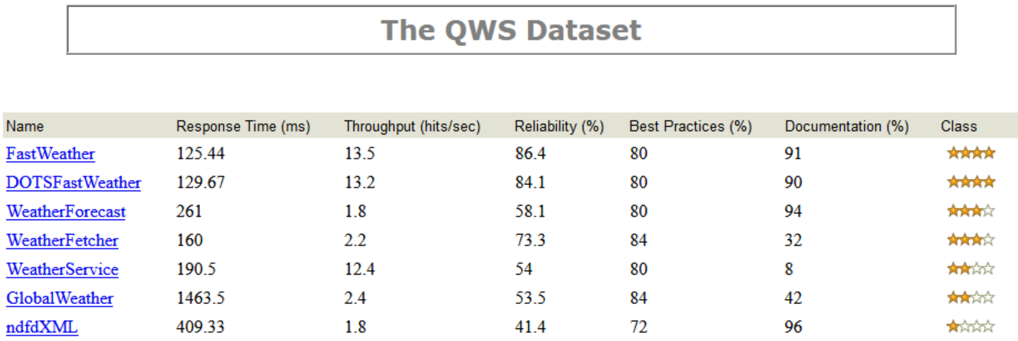
Fig. 5. An example: the evaluation results of TOPSIS evaluation tool for the weather services

this course, we take one group as an example to testify the evaluation process of our proposed scheme. Finally, we acquire the final ranking results.

When clicking the link "Search Services Demo" on the website, we can go to the search page, which provides us with search function using keywords through the contents of the dataset to display the similar Web services. In order to acquire services with same functionality, we tried various keywords and finally selected 10 groups of services, each type including at least three services. These keywords are fax, email, time, news, phone, weather, stock, sms, quote, and test. As the literal meanings of these keywords show, these 10 groups of Web services separately offer the function related to fax, email, time, news, phone, weather, stock, sms, quote, and test. The number of services included in these groups is 3, 4, 4, 5, 7, 7, 7, 8, 9, 9 respectively. The Web services in each type are regarded as different ones but with same functionality as their keywords show. Next we will take the keyword "weather" as an example to demonstrate the evaluation process.

Fig.6 shows the desired search list when we input the keyword "weather". It consists of seven weather Web services and each service lists only five of nine quality parameters here (column 2 to 6): Response Time, Throughput, Reliability, Best Practices and Documentation. The first column is service name, and the last one is classification that represents four different levels of QoS according to overall ratings in essence. It is not difficult to find that the more stars the service has, the better its quality is. The total nine QoS attributes and their descriptions are shown in Table 2. For ease of use, each attribute in the first column is followed by its abbreviation in bracket. Among these attributes, only Response Time and Latency are negative attributes, others are all positive attributes, as shown in the last column.

To evaluate these services and compare our evaluation results with the existing ratings of



The QWS Dataset

Name	Response Time (ms)	Throughput (hits/sec)	Reliability (%)	Best Practices (%)	Documentation (%)	Class
FastWeather	125.44	13.5	86.4	80	91	
DOTSFastWeather	129.67	13.2	84.1	80	90	
WeatherForecast	261	1.8	58.1	80	94	
WeatherFetcher	160	2.2	73.3	84	32	
WeatherService	190.5	12.4	54	80	8	
GlobalWeather	1463.5	2.4	53.5	84	42	
ndfdXML	409.33	1.8	41.4	72	96	

Fig. 6. Seven weather Web services searched from QWS dataset

Table 2
QoS attributes, descriptions and units of Web services in QWS dataset (version 1.0)

Attributes	Descriptions	Units	Positive/Negagive
Response Time (RT)	Time taken to send a request and receive a response	ms	Negative
Availability (Ava)	Number of successful invocations / total invocations	%	Positive
Throughput (TP)	Total number of invocations for a given period of time	hits/sec	Positive
Successability (Suc)	Number of response / number of request messages	%	Positive
Reliability (Rel)	Ratio of the number of error messages to total messages	%	Positive
Compliance (Com)	The extent to which a WSDL document follows WSDL specification	%	Positive
Best Practices (BP)	The extent to which a Web service follows WS-I Basic Profile	%	Positive
Latency (Lat)	Time taken for the server to process a given request	ms	Negative
Documentation (Doc)	Measure of documentation (i.e. description tags) in WSDL	%	Positive

classification, we need to find all the attributes information and their initial ratings from the QWS dataset. The search results of the seven weather services are shown in Table 3. For ease of description, we have numbered the seven weather services from s_1 to s_7, and we use the abbreviation of each attribute. The last two columns are the overall ratings and corresponding rankings for the seven weather services, respectively.

Next we will perform the TOPSIS evaluation process according to the steps in Section 5.3.

Step 1: establish the matrix of candidate services and their QoS attributes

As mentioned in Section 6.1, each service was tested over a ten-minute period for three

Table 3
The complete attribute information of seven weather services, original ratings and derived rankings

Services	RT	Ava	TP	Suc	Rel	Com	BP	Lat	Doc	Ratings	Rankings
s_1	125.44	100	13.5	86	86.4	78	80	125.33	91	86	1
s_2	129.67	100	13.2	86	84.1	78	80	129.56	90	85	2
s_3	261	100	1.8	71	58.1	78	80	229.5	94	71	3
s_4	160	100	2.2	71	73.3	78	84	74	32	71	3
s_5	190.5	86	12.4	71	54	89	80	188.5	8	64	5
s_6	1463.5	100	2.4	71	53.5	78	84	1410.5	42	61	6
s_7	409.33	49	1.8	27	41.4	89	72	401.5	96	55	7

consecutive days by the creators of the QWS dataset. Therefore, we have enough reason to consider that the quality parameters of seven weather Web services are the average values of many times. Then we can skip Step 0 in Section 5.3, and directly build the matrix M_1 of seven weather services and their QoS attributes according to Step 1 in Section 5.3. It is easy to see that the first 10 columns in Table 3 constructs the matrix M_1, as shown in Table 4.

Table 4
The matrix M_1 of seven weather services and its QoS attributes

Services	RT	Ava	TP	Suc	Rel	Com	BP	Lat	Doc
s_1	125.44	100	13.5	86	86.4	78	80	125.33	91
s_2	129.67	100	13.2	86	84.1	78	80	129.56	90
s_3	261	100	1.8	71	58.1	78	80	229.5	94
s_4	160	100	2.2	71	73.3	78	84	74	32
s_5	190.5	86	12.4	71	54	89	80	188.5	8
s_6	1463.5	100	2.4	71	53.5	78	84	1410.5	42
s_7	409.33	49	1.8	27	41.4	89	72	401.5	96

Step 2: normalize the matrix of all candidate services and their QoS attributes

As mentioned before, various QoS attributes are usually different in units, ranges and even meanings. To eliminate the inconsistency between them in order to compare them together fairly, we need to normalize the raw data. Vector normalization has been justified to be an ideal normalization algorithm for TOPSIS method. In this step, we employ vector normalization to separately normalize the positive attributes and the negative attributes in matrix M_1 according to formula (2) and (3) of Step 2 in Section 5.3. The normalized matrix M_2 of seven weather services and its QoS attributes are shown in Table 5.

Step 3: determine the combined weight for each QoS attribute

In this step, we first calculate the objective entropy weight for each attribute of seven weather services according to formula (4) - (7) of Step 3 in Section 5.3, then combine them with the subjective weight of trust preference from the requesting CSC to various QoS attributes according to formula (9) and finally get the combined weight. Table 6 shows the characteristic proportion results according to formula (4), which reflects the probability of a certain QoS attribute for each candidate service and will be used for the subsequent entropy calculation. Table 7 mainly shows the different weight results according to formula (5) - (9) respectively, where the entropy

Table 5
The normalized matrix M_2 of seven weather services and its QoS attributes

Services	RT	Ava	TP	Suc	Rel	Com	BP	Lat	Doc
s_1	0.5491	0.4089	0.5879	0.4553	0.4925	0.3627	0.3776	0.4207	0.4713
s_2	0.5312	0.4089	0.5748	0.4553	0.4794	0.3625	0.3776	0.4070	0.4661
s_3	0.2639	0.4089	0.0784	0.3759	0.3312	0.3627	0.3776	0.2298	0.4868
s_4	0.4305	0.4089	0.0958	0.3759	0.4178	0.3627	0.3964	0.7125	0.1657
s_5	0.3616	0.3517	0.5400	0.3759	0.3078	0.4138	0.3776	0.2797	0.0414
s_6	0.0471	0.4089	0.1045	0.3759	0.3050	0.3627	0.3964	0.0374	0.2175
s_7	0.1683	0.2004	0.0784	0.1429	0.2360	0.4138	0.3398	0.1313	0.4972

value of each attribute denotes its variation degree. The bigger the entropy value is, the smaller the variation between values of the corresponding attribute is. Correspondingly, the difference coefficient is smaller and the entropy weight is smaller. In essence, the entropy weight from a certain QoS attribute reflects its importance degree in evaluation. There is a positive relationship between them.

Table 6
The characteristic proportion for each QoS attribute of seven weather services

Services	RT	Ava	TP	Suc	Rel	Com	BP	Lat	Doc
s_1	0.2335	0.1575	0.2854	0.1781	0.1917	0.1373	0.1429	0.1897	0.2009
s_2	0.2259	0.1575	0.2791	0.1781	0.1866	0.1373	0.1429	0.1835	0.1987
s_3	0.1122	0.1575	0.0381	0.1470	0.1289	0.1373	0.1429	0.1036	0.2075
s_4	0.1831	0.1575	0.0465	0.1470	0.1626	0.1373	0.15	0.3212	0.0706
s_5	0.1538	0.1354	0.2622	0.1470	0.1198	0.1567	0.1429	0.1261	0.0177
s_6	0.0200	0.1575	0.0507	0.1470	0.1187	0.1373	0.15	0.0169	0.0927
s_7	0.0716	0.0772	0.0381	0.0559	0.0918	0.1567	0.1286	0.0592	0.2119

Table 7
Results of different weights for each QoS attribute of seven weather services

Parameters	RT	Ava	TP	Suc	Rel	Com	BP	Lat	Doc
Entropy	0.9183	0.9887	0.8262	0.9780	0.9845	0.9991	0.9994	0.8856	0.9135
Difference coefficient	0.0817	0.0113	0.1738	0.0220	0.0155	0.0010	0.0006	0.1144	0.0865
Entropy weight	0.1613	0.0223	0.3430	0.0434	0.0306	0.0019	0.0011	0.2258	0.1707
Subjective weight	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111	0.1111
Combined weight	0.0179	0.0025	0.0381	0.0048	0.0034	0.0002	0.0001	0.0251	0.0190

In order to underline the objectivity and validity of entropy weight method and meanwhile simplify the verification process in this experiment, we ignore the differences in subjective trust preference of a CSC to various QoS attributes. For this purpose, we suppose all the values in subjective trust preference vector from the requesting CSC are equal. As shown in Table 7, the subjective weight of each attribute is set to be the same value ($\frac{1}{9} \doteq 0.1111$), which means each attribute acquires the same importance degree from the requesting CSC. In this case, each quality attribute is just determined by entropy weight in this experiment. From Table 7, we can see that the entropy weight value of TP (i.e. Throughput) is 0.3430, which is the biggest among nine attributes, representing this attribute is the most important of all; while BP (i.e. Best Practices) gets the smallest entropy weight value 0.0011, representing it is the least important of all.

The values of the combined weight in Table 7 are the product of each entropy weight and each subjective weight respectively, which will be employed as the weight in TOPSIS method for subsequent calculation.

Step 4: weight the normalized matrix of candidate services and QoS attributes

Applying the combined weight in table 7 into TOPSIS method, we get the weighted normalized matrix M_3 of cloud services and QoS attributes according to formula (10), as is shown in Table 8.

Table 8
Results of weighted normalized matrix M_3 with the combined weight

Services	RT	Ava	TP	Suc	Rel	Com	BP	Lat	Doc
s_1	0.0098	0.0010	0.0224	0.0021	0.0017	7.52E-05	4.64E-05	0.0106	0.0089
s_2	0.0095	0.0010	0.0219	0.0022	0.0016	7.52E-05	4.64E-05	0.0102	0.0088
s_3	0.0047	0.0010	0.0030	0.0018	0.0011	7.52E-05	4.64E-05	0.0058	0.0092
s_4	0.0077	0.0010	0.0037	0.0018	0.0014	7.52E-05	4.87E-05	0.0179	0.0031
s_5	0.0065	0.0009	0.0206	0.0018	0.0011	8.59E-05	4.64E-05	0.0070	0.0008
s_6	0.0008	0.0010	0.0040	0.0018	0.0010	7.52E-05	4.87E-05	0.0009	0.0041
s_7	0.0030	0.0005	0.0030	0.0007	0.0008	8.59E-05	4.18E-05	0.0033	0.0094

Step 5: determine the positive ideal solution and the negative ideal solution of all candidate services

In this step, our purpose is to determine the positive ideal solution and the negative ideal solution respectively according to formula (11), i.e. the best service and the worst service in theory. The corresponding results are shown in Table 9.

Table 9
Results of positive ideal solutions (PIS) and negative ideal solutions (NIS)

PIS/NIS	RT	Ava	TP	Suc	Rel	Com	BP	Lat	Doc
PIS	0.0098	0.0010	0.0224	0.0022	0.0017	8.59E-05	4.87E-05	0.0179	0.0094
NIS	0.0008	0.0005	0.0030	0.0007	0.0008	7.52E-05	4.18E-05	0.0009	0.0008

Step 6: calculate the distance from each cloud service to the positive ideal solution and the negative ideal solution respectively

Based on the above steps, we can separately calculate the distance from each cloud service to the positive ideal solution and the negative ideal solution according to formula (12). The results are shown in Table 10.

Step 7: calculate the trustworthiness score of each cloud service

Now we can calculate the trustworthiness score of each service according to formula (13). The corresponding results are shown in Table 10, where the trustworthiness values are the same as the results calculated by our evaluation tool, as shown in Fig.5 of Section 6.2.1.

The larger the trustworthiness value is, the better the quality of the weather service is, and vice versa. In terms of the trustworthiness values, we can easily get the ranking results, as shown in the last column of Table 10.

The rest nine groups of services can be evaluated and ranked in the same way. The experiment results will be analyzed in Section 6.2.3 with that of the second kind of experiments.

Table 10
Distances between each service and ideal solutions, trustworthiness values and rankings

Services	ds^+	ds^-	Trustworthiness	Rankings
s_1	0.0073	0.0249	0.7724	1
s_2	0.0077	0.0242	0.7586	2
s_3	0.0235	0.0106	0.3103	5
s_4	0.0199	0.0185	0.4817	4
s_5	0.0144	0.0195	0.5748	3
s_6	0.0271	0.0037	0.1202	7
s_7	0.0253	0.0092	0.2672	6

6.2.2. The second kind of experiments: evaluating the services derived from random sampling

In order to extend the scope of experiment samples and the number of samples, we design the second kind of experiments. To achieve this goal, we first acquire 30 groups of samples from QWS dataset (version 1.0) by random sampling. The number of services for each group is varied from 5 to 150 in increment of 5. The total number of all these samples is 2325 and their distribution is shown as Fig.7. Based on the above samples, we then evaluate them one group by one using the automated tool. Since the process is the same as in Section 6.2.1, no more details here.

6.2.3. Results analysis of experiment 1

After we rank the Web services in the above two kinds of experiments based on the proposed evaluation scheme, we make analysis employing Kendall's Tau rank correlation test [51, 52]. Kendall's Tau correlation coefficient τ can evaluate the similarity between two rankings given to the same set of objects. It depends on the relative proportion of the number between concordant pairs and discordant pairs from the two ordered sets (i.e. two rankings) on the same set of objects. The range of τ is [-1, 1]. If there are repeated orders (i.e. ties) in the compared rankings, we use formula (3.3) in Reference [52] to calculate τ. Otherwise, we use formula (4) in Reference [51]. To get the final results of all groups easily, we implement the algorithm in Python.

Since the rankings with which we deal in practice are usually based on a set of individuals which themselves are only samples from a much larger population, it is necessary to test the significance of observed rank correlations in the special sense of the statistical theory of sampling [52]. It is the same situation for the rankings in the two kinds of experiments. Next we will make analysis based on the one-tail test of τ with significance level $\alpha = 0.05$ and $\alpha = 0.01$, respectively.

For the first kind of experiments, we first calculate Kendall's Tau rank correlation coefficient between our rankings and the original rankings derived from the QWS dataset for 10 groups of

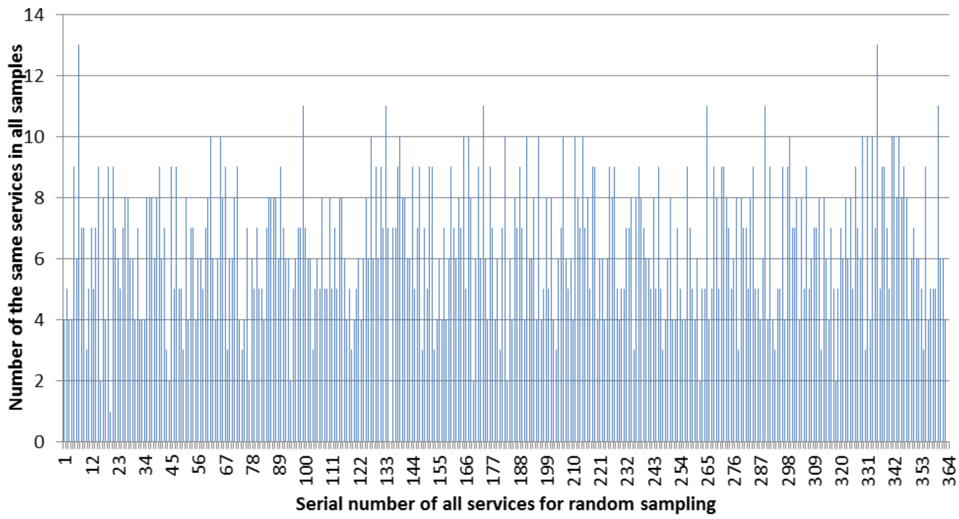


Fig. 7. Samples distribution for 30 groups of random sampling from QWS dataset 1.0

searched services, then decide if the correlation between every two rankings for each group is significant at two different significance levels. Take the seven weather services as an example, as there exist repeated orders (i.e. ties) in the original rankings of seven weather services (as shown in Table 3), we first select formula (3.3) in Reference [52] to calculate τ and get the final calculation result $\tau \doteq 0.6343$, then compare the calculated τ with that in Table 2 of Reference [51]. Since $0.6190 \leq \tau \leq 0.8095$, we can infer that the correlation between our rankings and the original rankings from QWS ratings is significant at $\alpha = 0.05$ but not at $\alpha = 0.01$. Similarly, we perform the same operations to the other nine service groups mentioned in Section 6.2.1, i.e. fax, time, email, news, phone, stock, sms, quote, and test. The results of calculated τ and its contrast values at significance level $\alpha = 0.05$ are shown in Table 11.

Table 11
Results of Kendall's Tau rank correlation coefficient for 10 groups of services

Type	fax	email	time	news	phone	weather	stock	sms	quote	test
Number	3	4	4	5	7	7	7	8	9	9
τ	1	0.6667	1	1	0.7143	0.6343	0.7143	0.4728	0.3662	0.6480
$\alpha = 0.05$	1	1	1	0.8000	0.6190	0.6190	0.6190	0.5714	0.5000	0.5000

According to Table 11, we can see that in all 10 types of services, seven types have great similarity compared with the original ranking results at significance level $\alpha = 0.05$. However, it is far from the perfect similarity. Meanwhile, the number of samples for each group is very small.

For the second kind of experiments, as described in Section 6.2.2, we enlarge the scope of samples by random sampling and the number of each group (denoted as N) is varied from 5 to 150 in increment of 5 step by step. For the 30 groups of samples, after evaluating them one group

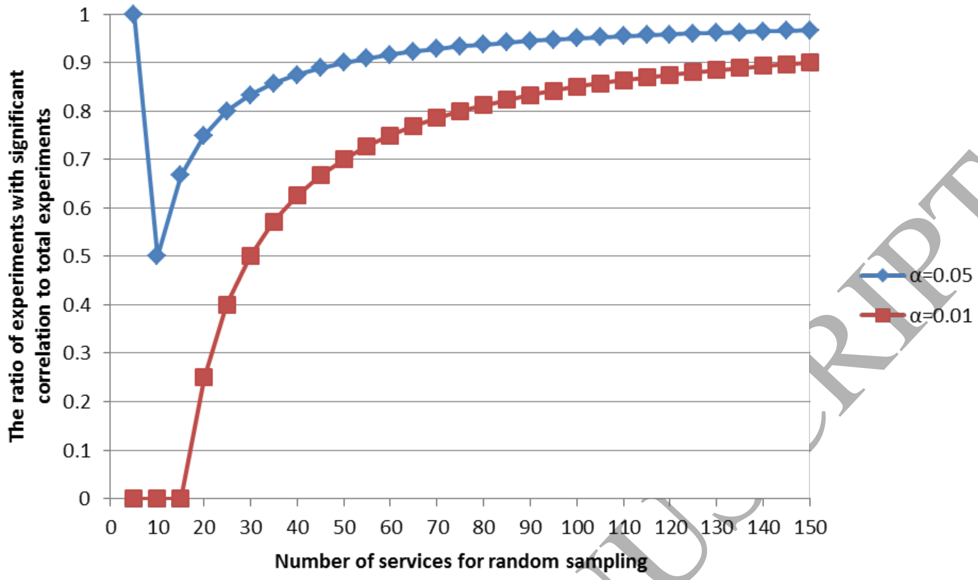


Fig. 8. Kendall Tau correlation test at different significance levels for 30 groups of random sampling

by one using the automated tool, we employ IBM SPSS Statistics 20 to make the one-tail test of τ for each group of rankings with significance level $\alpha = 0.05$ and $\alpha = 0.01$, respectively. The experiment results are shown in Fig.8.

From Fig.8, we can find that the proposed TOPSIS evaluation scheme can acquire more significant correlation and stability with the increase of sample size N. Apparently, when $N \leq 15$, i.e.for small size of samples, the correlation between our rankings and the original rankings from QWS dataset is not significant. This is also consistent with the results of the first kind of experiments. However, when $N \geq 20$, the correlation between two rankings is becoming more and more significant.

Based on the above analysis, we can conclude that the proposed TOPSIS evaluation scheme using entropy weight for cloud service trustworthiness is feasible and effective to a great extent, especially for larger size of samples ($N \geq 20$).

6.3. Experiment 2: demonstrating the benefit of subjective trust preference in terms of CSC's Satisfaction Degree

Customer satisfaction level is one of the most important factors in the evaluation of the quality and service during development of a new product [46]. Su et al. [32] have proved the positive relationship between the Satisfaction Degree and the similarity between a consumer's priority to different QoS attributes and the priority of reference service in their experiments. To demonstrate the benefit of the subjective trust preference in our proposed evaluation scheme, we introduce the same concept proposed by Su et al. [32] and convert it into another appropriate form through mapping in this study. At last, we make comparisons by case study to verify the benefit of combining CSC's subjective trust preference in our evaluation scheme.

The definition of Satisfaction Degree in Reference [32] is shown in formula (14):

$$SatDegree_i = Sim \times \frac{\sum_{j=1}^{n} R_j}{n \times 100}, \qquad (i = 1, 2, ..., m) \tag{14}$$

$SatDegree_i$ refers to the satisfaction degree of the ith service from the perspective of a certain consumer. R_j represents the rating score for the jth attribute of the ith service ranging in $[0, 100]$, which is offered by the requested provider and initially derives from the referenced consumer whose priority is closest to the requesting consumer compared with other consumers who have used the service. The sum of all R_j divided by $n \times 100$ is to average and scale the result into a value ranging in $[0, 1]$. Sim in essence refers to the similarity of priorities between the requesting consumer and the referenced consumer. It is defined using dot product by formula (15) [32]:

$$Sim = \frac{\sum_{k=1}^{n} (CW_k \times RW_k)}{\sqrt{(\sum_{k=1}^{n} (CW_k)^2) \times (\sum_{k=1}^{n} (RW_k)^2)}}, \qquad (i = 1, 2, ..., m) \tag{15}$$

where CW_k and RW_k represent the priority of the kth attribute of the requesting consumer and the referenced consumer respectively.

Similarly, we apply the concept of Satisfaction Degree into our study. In theory, the parameter CW_k and RW_k in formula (15) should be mapped to the weight of the kth attribute respectively derived from the requesting CSC and the referenced CSC, who has the greatest similarity with the requesting CSC in trust preference among all the consumers who have used the service provided by the requested CSP. Considering the problem of data source and the reason that the ratings from the referenced consumer are directly provided by the requested provider in essence in [32], we take a simplified and reasonable approach that does not affect the conclusions, replacing the referenced CSC mentioned above with the requested CSP in the following descriptions.

Since our aim is to compare the difference of Satisfaction Degree before and after we introduce the trust preference, we distinguish two situations for the requested CSPs in this experiment: before using the trust preference, we regard each weight to each attribute provided by all CSPs as $\frac{1}{n}$, here n is the number of attributes for the same kind of services; after using the trust preference, we regard it to be the same as that of the requesting CSC because it is completely consistent with the subjective need of the requested CSC when evaluating the trustworthiness of each service provided by all the requested CSPs.

Due to the difference of QoS information source, we still need to modify formula (14) through mapping. Because the ratings from the referenced consumer are usually equivalent to the appropriate evaluation to the attributes of QoS [32], it is reasonable to employ the real values of QoS attributes instead of ratings in order to ensure the reliability and objectivity. Based on the above analysis, we convert the expression $\frac{\sum_{j=1}^{n} R_j}{n \times 100}$ in formula (14) into the equivalent $\frac{\sum_{j=1}^{n} ls_{ij}}{n}$ in function, where ls_{ij} is the value of the jth attribute for the ith service after linear normalization using formula (16) and (17) to distinguish between positive attributes and negative attributes for the data in Matrix M_1. Then the mapping formula of Satisfaction Degree is shown in formula (18):

$$positive\ attributes: \qquad ls_{ij} = \frac{s_{ij}}{\max\limits_{i=1,2,...,m}(s_{ij})}, \qquad (i = 1, 2, ..., m; j = 1, 2, ..., n) \tag{16}$$

$$negative\ attributes: \qquad ls_{ij} = \frac{\min\limits_{i=1,2,...,m}(s_{ij})}{s_{ij}}, \qquad (i = 1, 2, ..., m; j = 1, 2, ..., n) \tag{17}$$

$$SatDegree_i = Sim \times \frac{\sum_{j=1}^{n} ls_{ij}}{n}, \qquad (i = 1, 2, ..., m) \tag{18}$$

In order to examine the effect of trust preference on Satisfaction Degree, we separately calculate its value before we introduce trust preference and afterwards for each of the seven weather services in experiment 1. One important step is to determine the value of Sim. As mentioned above, before introducing the trust preference, we have enough reason to believe that each QoS attribute value provided by every CSP should be assigned the average weight (i.e. the same weight value $\frac{1}{9}$) by the requested CSPs because the subjective requirement from the requesting CSC is ignored, no matter what the trust preference of the requesting CSC is. Under such circumstance, the value of Sim is the distance between the real trust preference vector and the average weight vector in essence. Afterwards, we can believe that the similarity is always equal to 1 because the combined weight completely incorporates the subjective trust preference of the requesting CSC.

Next we take 10 CSCs as an example and separately assign the subjective weights of their trust preference to nine common QoS attributes of the seven weather services. Considering the diversity and possibility of subjective trust preference, the subjective weights from 10 CSCs are simulated and generated by random number generator, as shown in Table 12. Then we separately calculate 10 values of Sim according to formula (15), and the subsequent Satisfaction Degrees according to formula (18) for each service which represent the Satisfaction Degrees before introducing trust preference. The results are shown in Fig.9. Similarly, we can calculate the Satisfaction Degrees of 10 CSCs after introducing subjective trust preference. Since the needs of 10 CSCs are satisfied after considering trust preferences (i.e. all the Sim values equal to 1), their Satisfaction Degrees completely depend on the average value of the scaled QoS attributes in this case. After that, we can calculate the average value of Satisfaction Degrees for each of the seven weather services before and after using trust preference respectively. The final comparison results of average Satisfaction Degrees are shown in Fig.10.

Table 12
Subjective weights of trust preference randomly generated for 10 CSCs

CSCs	RT	Ava	TP	Suc	Rel	Com	BP	Lat	Doc
1	0.1503	0.3024	0.1126	0.1962	0.1168	0.0247	0.0381	0.0217	0.0373
2	0.2665	0.3398	0.1327	0.0237	0.0822	0.0404	0.0377	0.0246	0.0523
3	0.4936	0.1862	0.1226	0.0550	0.0086	0.0314	0.0221	0.0318	0.0488
4	0.4304	0.0224	0.0226	0.2447	0.0703	0.0415	0.02810	0.03810	0.1020
5	0.1392	0.4272	0.0523	0.1464	0.0878	0.0341	0.0263	0.0190	0.0678
6	0.0815	0.1244	0.2149	0.1749	0.0682	0.0303	0.1051	0.0145	0.1864
7	0.0323	0.0865	0.1904	0.2183	0.0148	0.2042	0.0847	0.02306	0.1457
8	0.2887	0.0579	0.1312	0.1875	0.0677	0.0749	0.0470	0.0403	0.1048
9	0.3747	0.0198	0.1659	0.0934	0.0019	0.0121	0.0262	0.1258	0.1803
10	0.0995	0.2789	0.2271	0.0410	0.0051	0.0564	0.0795	0.0916	0.1208

Now we can make analysis based on the above results. As is shown in Fig.9, the relationship of positive correlation between Sim and SatDegree in definition (18) is well depicted. It is easy to conclude that the greater the similarity of the trust preferences between the requesting CSC and the requested CSP (i.e. the referenced CSC in theory), the greater the Satisfaction Degree of the

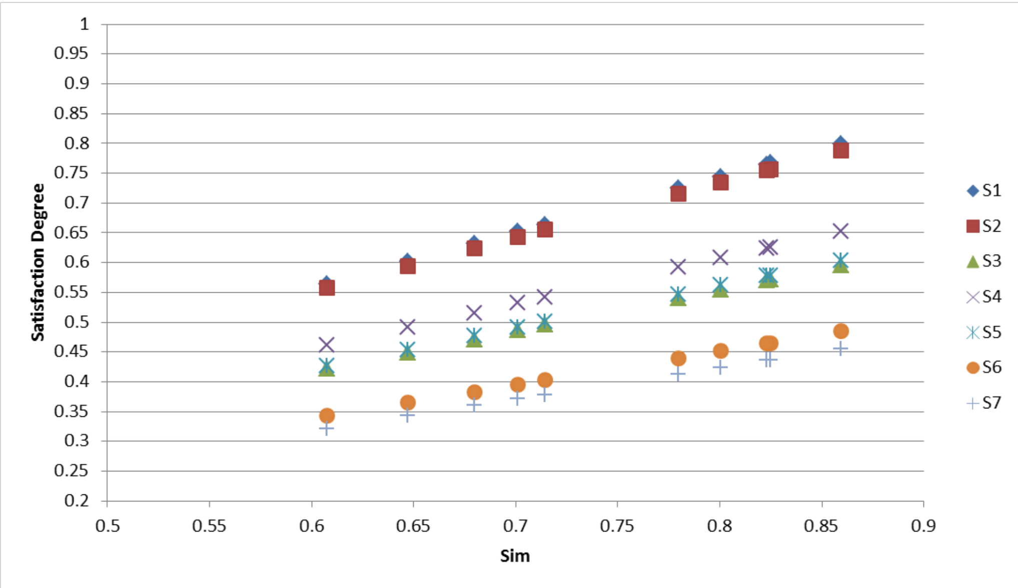


Fig. 9. The relationship between Sim and Satisfaction Degree

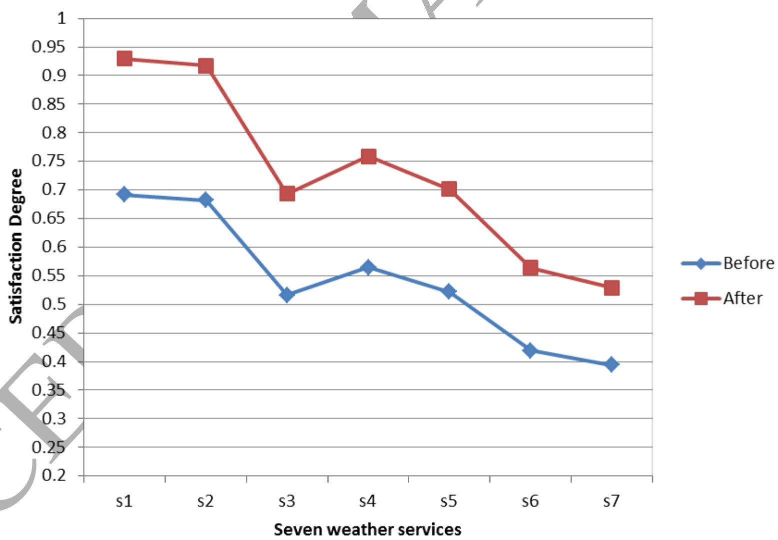


Fig. 10. Average Satisfaction Degrees of 10 CSCs for seven weather services before and after using trust preference

requesting CSC is. In addition, it implies that Satisfaction Degree can vary with different ratings for QoS attributes of the candidate services (reflecting different attribute values in essence), even though the similarities of trust preferences between the same requesting CSC and different referenced CSCs are the same. According to Fig. 10, we can find that the average Satisfaction Degree of each weather service is always higher than before after using subjective trust preference. In theory, the trust preference of the referenced CSC is closer to the need of the requesting CSC, the Satisfaction Degree is higher.

In conclusion, trust preference reflects the subjective aspects when evaluating the overall trustworthiness in essence. The results in this experiment demonstrate that after employing the subjective trust preference of CSCs in the proposed TOPSIS evaluation scheme, the results of service trustworthiness are better and more satisfactory for CSCs than before.

7. Conclusion and future work

Cloud computing has been paid more attention as a growing computing paradigm. Besides its several benefits, trust issue greatly affects the success of cloud services adoption. In order to help CSCs to choose the trustworthy cloud service, we propose the common objective QoS attributes of cloud services in this study. Most of all, inspired by a couple of existing work, we build a CSTE model and put forward a novel TOPSIS evaluation scheme for cloud service trustworthiness combining both objective and subjective aspects.

Our main work is different from existing work in several ways. First, we consider the objectivity of QoS attributes values from two aspects: one is the QoS information source of cloud services, the other is the objective weight of different QoS attributes. Second, we consider the subjectivity nature of trust that reflects the individual trust preference of CSCs and describe it as a weight vector from the perspective of CSCs. Most important of all, we put forward the combined weight integrating both objective and subjective aspects mentioned above, and successfully apply it into TOPSIS method that effectively solves the weight assignment problem and form a novel evaluation scheme for cloud service trustworthiness. Two sets of experiments based on QWS dataset together demonstrate the feasibility, effectiveness, and high Satisfaction Degree of our proposed scheme.

However, the proposed evaluation scheme for cloud service trustworthiness in this study still has several challenges to confront in future work. One is the dynamic of trustworthiness evaluation. In online situation, it is necessary to focus on the dynamic of cloud service applications[3]. Another is the optimization of TOPSIS method. In order to attain better performance in trustworthiness evaluation, it is necessary to improve TOPSIS method or find a more effective way. Furthermore, we will investigate a more practicable way to determine the trust preference vector in future research, for example, based on the data derived from the social network of CSCs to obtain the trust preference.

ACKNOWLEDGEMENTS

This work has been greatly supported by the National Natural Science Foundation of China (No. 91118002) and Key Laboratory of Trustworthy Distributed Computing and Service, Ministry of education, School of Software, Beijing University of Posts and Telecommunications. The QWS dataset is supported by the original authors. The quality of this paper has been greatly improved thanks to the anonymous reviewers' insightful comments and constructive suggestions.

References

[1] T. H. Noor, Q. Z. Sheng, Credibility-based trust management for services in cloud environments, in: Service-Oriented Computing - International Conference, ICSOC 2011, Paphos, Cyprus, December 5-8, 2011 Proceedings, 2011, pp. 328–343.

[2] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, I. Brandic, Cloud computing and emerging it platforms: Vision, hype, and reality for delivering computing as the 5th utility, Future Generation Computer Systems 25 (6) (2009) 599 – 616. doi:https://doi.org/10.1016/j.future.2008.12.001.
URL http://www.sciencedirect.com/science/article/pii/S0167739X08001957

[3] S. Ding, S. Yang, Y. Zhang, C. Liang, C. Xia, Combining qos prediction and customer satisfaction estimation to solve cloud service trustworthiness evaluation problems, Knowledge-Based Systems 56 (3) (2013) 216–225.

[4] T. H. Noor, Q. Z. Sheng, Trust as a service: A framework for trust management in cloud environments, in: Web Information System Engineering - WISE 2011 - International Conference, Sydney, Australia, October 13-14, 2011. Proceedings, 2011, pp. 314–321.

[5] A. Abdelmaboud, D. N. A. Jawawi, I. Ghani, A. Elsafi, B. Kitchenham, Quality of service approaches in cloud computing: A systematic mapping study, Journal of Systems and Software 101 (C) (2015) 159–179.

[6] S. M. Habib, S. Ries, M. User, P. Varikkattu, Towards a trust management system for cloud computing marketplaces: using caiq as a trust information source, Security and Communication Networks 7 (11) (2014) 2185–2200.

[7] T. H. Noor, Q. Z. Sheng, A. H. H. Ngu, S. Dustdar, Analysis of web-scale cloud services, IEEE Internet Computing 18 (18) (2014) 55–61.

[8] R. K. L. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Q. Liang, B. S. Lee, Trustcloud: A framework for accountability and trust in cloud computing, in: Services, 2011, pp. 584–588.

[9] S. M. Habib, V. Varadharajan, A framework for evaluating trust of service providers in cloud marketplaces, in: ACM SAC, 2013, pp. 1963–1965.

[10] V. V. Rajendran, S. Swamynathan, Hybrid model for dynamic evaluation of trust in cloud services, Wireless Networks 22 (6) (2016) 1807–1818.

[11] L. Mui, M. Mohtashemi, A. Halberstadt, A computational model of trust and reputation for e-businesses, in: Hawaii International Conference on System Sciences, 2002, p. 188.

[12] A. Jsang, R. Ismail, The beta reputation system, in: Bled Conference on Electronic Commerce, 2002, pp. 1–14.

[13] M. K. Denko, T. Sun, I. Woungang, Trust management in ubiquitous computing: A bayesian approach, Computer Communications 34 (3) (2011) 398–406.

[14] J. H. Cho, A. Swami, I. R. Chen, A survey on trust management for mobile ad hoc networks, IEEE Communications Surveys and Tutorials 2 (2) (2010) 562–583.

[15] L. Xiong, L. Liu, Peertrust: Supporting reputation-based trust for peer-to-peer electronic communities, IEEE Transactions on Knowledge and Data Engineering 16 (7) (2004) 843–857.

[16] C. Tian, B. Yang, J. Zhong, X. Liu, Trust-based incentive mechanism to motivate cooperation in hybrid p2p networks, Computer Networks 73 (C) (2014) 244–255.

[17] Y. Ben Saied, A. Olivereau, D. Zeghlache, M. Laurent, Trust management system design for the internet of things: A context-aware and multi-service approach, Computers & Security 39 (39) (2013) 351365.

[18] T. Noor, Q. Sheng, L. Yao, S. Dustdar, A. Ngu, Cloudarmor: Supporting reputation-based trust management for cloud services, IEEE Transactions on Parallel and Distributed Systems 27 (2) (2016) 367–380.

[19] S. Ding, C. Y. Xia, C. Y. Xia, K. L. Zhou, S. L. Yang, J. S. Shang, Decision support for personalized cloud service selection through multi-attribute trustworthiness evaluation, Plos One 9 (6) (2014) e97762.

[20] M. Behzadian, S. Khanmohammadi Otaghsara, M. Yazdani, J. Ignatius, A state-of the-art survey of topsis applications, Expert Systems with Applications 39 (17) (2012) 13051–13069.

[21] A. Jsang, S. L. Presti, Analysing the relationship between risk and trust, Lecture Notes in Computer Science lncs 2 (2004) 135–145.

[22] S. Adali, Trust as a Computational Concept, Springer New York, 2013.

[23] D. M. Rousseau, S. B. Sitkin, R. S. Burt, C. Camerer, Introduction to special topic forum: Not so different after all: A cross-discipline view of trust, Academy of Management Review 23 (3) (1998) 393–404.

[24] D. Gambetta, Can we trust trust?, Trust Making and Breaking Cooperative Relations 5 (4) (1988) 213–237.

[25] H. T. Nguyen, W. Zhao, J. Yang, A trust and reputation model based on bayesian network for web services, in: IEEE International Conference on Web Services, 2010, pp. 251–258.

[26] X. Zhu, F. Wang, H. Wang, C. Liang, R. Tang, X. Sun, J. Li, Topsis method for quality credit evaluation: A case of air-conditioning market in china, Journal of Computational Science 5 (2) (2014) 99–105.

[27] R. J. R. Raj, T. Sasipraba, Web service selection based on qos constraints, in: Trendz in Information Sciences and Computing, 2010, pp. 156–162.

[28] S. K. Garg, S. Versteeg, R. Buyya, A framework for ranking of cloud computing services, Future Generation Computer Systems 29 (4) (2013) 1012–1023.

[29] S. M. Habib, S. Ries, P. Varikkattu, Towards a trust management system for cloud computing marketplaces: using caiq as a trust information source, Security and Communication Networks 7 (11) (2015) 2185–2200.

[30] N. Limam, R. Boutaba, Assessing software service quality and trustworthiness at selection time, IEEE Transactions on Software Engineering 36 (4) (2010) 559–574.

[31] S. Nepal, M. Pathan, Security, Privacy and Trust in Cloud Systems, Springer Berlin Heidelberg, 2014.

[32] X. Su, M. Zhang, Y. Mu, Q. Bai, A robust trust model for service-oriented systems, Journal of Computer and System Sciences 79 (5) (2013) 596–608.
[33] X. Li, J. Du, Adaptive and attribute-based trust model for service level agreement guarantee in cloud computing, Iet Information Security 7 (1) (2013) 39–50.
[34] M. Alhamad, T. Dillon, E. Chang, A trust-evaluation metric for cloud applications, International Journal of Machine Learning and Computing 1 (2011) 416–421.
[35] S. M. Habib, Trust establishment mechanisms for distributed service environments, PhD Thesis.Technische Universität (2014) 1–206.
[36] L. Xiong, L. Liu, A reputation-based trust model for peer-to-peer ecommerce communities, in: IEEE International Conference on E-Commerce, 2003, pp. 275–284.
[37] Y. Wang, J. Vassileva, A review on trust and reputation for web service selection, in: International Conference on Distributed Computing Systems Workshops, 2007, pp. 25–25.
[38] Z. Zheng, X. Wu, Y. Zhang, M. R. Lyu, J. Wang, Qos ranking prediction for cloud services, IEEE Transactions on Parallel and Distributed Systems 24 (6) (2013) 1213–1222.
[39] T. H. Noor, Q. Z. Sheng, S. Zeadally, J. Yu, Trust management of services in cloud environments:obstacles and solutions, Acm Computing Surveys 46 (1) (2013) 1–30.
[40] A. Iosup, S. Ostermann, M. N. Yigitbasi, R. Prodan, T. Fahringer, D. Epema, Performance analysis of cloud computing services for many-tasks scientific computing, IEEE Transactions on Parallel and Distributed Systems 22 (6) (2011) 931–945.
[41] S. Chakraborty, C. H. Yeh, A simulation comparison of normalization procedures for topsis, in: International Conference on Computers & Industrial Engineering, 2009, pp. 1815–1820.
[42] A. Çelen, Comparative analysis of normalization procedures in topsis method: With an application to turkish deposit banking market, Informatica 24 (2) (2012) 185–208.
[43] N. Vafaei, R. A. Ribeiro, L. M. Camarinha-Matos, Importance of data normalization in decision making: case study with topsis method, in: INT. CONFERENCE ON DECISION SUPPORT SYSTEMS TECHNOLOGIES AN EWG-DSS CONFERENCE. THEME: BIG DATA ANALYTICS FOR DECISION-MAKING, 2015, pp. 1–6.
[44] R. B. Ash, Information theory, Interscience Publishers, 1965.
[45] D. L. Mon, C. H. Cheng, J. C. Lin, Evaluating weapon system using fuzzy analytic hierarchy process based on entropy weight, Fuzzy Sets & Systems 62 (2) (1994) 127–134.
[46] L. Li, F. Liu, C. Li, Customer satisfaction evaluation method for customized product development using entropy weight and analytic hierarchy process, Computers & Industrial Engineering 77 (C) (2014) 80–87.
[47] T. L. Saaty, How to make a decision: The analytic hierarchy process, European Journal of Operational Research 48 (1) (1994) 9–26.
[48] O. Ibrahim, M. Nilashi, K. Bagherifard, N. Hashemi, N. Janahmadi, M. Barisami, Application of ahp and k-means clustering for ranking and classifying customer trust in m-commerce, Australian Journal of Basic & Applied Sciences 5 (12) (2011) 1441–1457.
[49] E. Al-Masri, Q. H. Mahmoud, Discovering the best web service, in: Proceedings of the 16th International Conference on World Wide Web, WWW '07, ACM, New York, NY, USA, 2007, pp. 1257–1258. doi:10.1145/1242572.1242795.
URL http://doi.acm.org/10.1145/1242572.1242795
[50] E. Al-Masri, Q. H. Mahmoud, Qos-based discovery and ranking of web services, in: International Conference on Computer Communications and Networks, 2007, pp. 529–534.
[51] H. Abdi, The kendall rank correlation coefficient, Encyclopedia of Measurement and Statistics 11 (2007) 508–510.
[52] M. G. Kendall, Rank correlation methods, C. Griffin, 1970.

Lilei Lu is an Associate Professor at the Department of Computer Science, Tangshan Normal University, Hebei, China. She received the M.S. in Software Engineering from Peking University, China. Currently she is a Ph.D. candidate at School of Software, Beijing University of Posts and Telecommunications, Beijing, China. Her research interests include trustworthy service, cloud computing and information security.

Yuyu Yuan is a Professor of Beijing University of Posts and Telecommunications, Beijing, China. She respectively received the M.S. and Ph.D. at University of Electronic Science and Technology of China and Research Institute for Fiscal Science, Ministry of Finance, China. Her main research interests include software quality, trustworthy service and software testing. She is also an expert in International Standards Workgroup.