MMH : An Effective Clustering Algorithm for Trustworthy Cloud Service Provider Selection

Article · May 2016

CITATIONS
2

READS
182

2 authors:

Monoj Kumar Muchahari
Kaziranga University
7 PUBLICATIONS **55** CITATIONS
SEE PROFILE

Smriti Kumar Sinha
Tezpur University
35 PUBLICATIONS **185** CITATIONS
SEE PROFILE

MMH : An Effective Clustering Algorithm for Trustworthy Cloud Service Provider Selection

Monoj Kumar Muchahari
Dept. of Computer Science and Engineering
Tezpur University
Napaam, Sonitpur, Assam-784 028, INDIA

Smriti Kumar Sinha
Dept. of Computer Science and Engineering
Tezpur University
Napaam, Sonitpur, Assam-784 028, INDIA

***Abstract*— Cloud computing is a new paradigm in computing that delivers pay per use resources as services. Right after the advent of cloud computing, widespread acceptance is witnessed. But issues of trust and security became the primary concern of cloud service users, hindering the cloud adoption. Different trust management systems are being proposed based on feedback from cloud service consumers. Now, trust datasets, based on cloud service users' feedback are available for research. In this paper, we use unsupervised learning method to cluster unlabeled trust data into trustworthy and non-trustworthy cloud service providers. Experimental results and validations of the proposed algorithm, using different cluster indices with two synthetic and two real-life trust datasets exhibits favourable performance.**

Keywords- Cloud computing; Trust; Unsupervised Learning; Clustering; Correlation Measure

I. INTRODUCTION

Cloud computing is a new paradigm of computing. [1] defines cloud computing as a system, where the resources of a data center are shared using virtualization technology, which also provide elastic, on-demand and instant services to its customers and charges customer usage as utility bill. According to [2], cloud computing provide virtualization-based services and applications operating on distributed network. Because of imperative characteristics like on-demand service, rapid elasticity, pay per service, etc., cloud computing is adopted by many organizations. Cloud Computing today, is being favoured for use of mobile devices to remotely execute services on clouds with reduced energy consumption [3] and also is becoming a paramount scientific application platform [4].

Even though there are many lucrative features of cloud, it still faces hurdles to adoption, growth, policy and business [5], [6]. Though cloud is gaining popularity for its dynamic capabilities and business benefits but trust on Cloud Service Providers (CSPs) by potential Cloud Service Consumers (CSCs) is a great concern. Questions about the trustworthiness of CSPs by large institution managers and stakeholders of information technology companies has perilously affected the cloud migration [7]. CSCs fear to give away their sensitive data to providers whom they cannot trust [8], [9]. Also, multi-tenant environment of cloud engender complex trust issues. Largely, users take into account of perceived security or trust while deciding whether or not to adopt new information technologies and services [10] and cloud computing being perceived as new [11], is also affected. Due to the lack of resource control, transparency, service level agreement, portability, performance, customer support, privacy [12] and hazy security assurance, CSCs hesitate to trust CSPs. Phaphoom et al. [13] in their survey, discuss some major technical barriers to cloud adoption. [14] with respect to cloud storage believe that CSPs cannot be trusted as they might hide data loss incidents to sustain reputation. Also, composition of Cloud infrastructure is often imperceptible for CSCs [15]. A substantial amount of research have been carried out for trust management in cloud in recent years. In the present scenario, the paramount concern for enterprises is the problem related to trust in cloud computing [16].

Trust can be built based on past experiences and feedbacks of prospective CSPs from consumers and also various other CSPs. CSCs' feedbacks can ensure the dependability of cloud resources [17]. Habib et al. [18] believe CSPs should be evaluated based on fine-grained QoS parameters together with CSCs' feedbacks, recommendations, and further specific parameters related to the cloud computing environment. Indeed, user's feedbacks rating proves as a good way to adjudge reputability. Eventually, Cloud Commons[1] intents to combine consumer feedbacks with technical measurements for assessing and comparing the trustworthiness of cloud providers [19]. All these suggests a publicly available cloud dataset. Noor et al. [20] feel the need of publicly available cloud service dataset for use in cloud computing research.

Dataset is mostly used in training and evaluating new systems. Verification of publication, longitudinal research, interdisciplinary use of data and valorization are the factors which describes the importance of dataset [21]. For these reasons researchers use either their own methods to build trust datasets or use datasets from available sources though all of them being unlabeled data. Unsupervised learning is regarded apt for finding concealed structure in unlabeled data.

[1] www.cloudcommons.com

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

In this paper, we introduce a data-driven learning approach to selection of trustworthy CSPs using unlabeled feedback dataset. After aggregating different feedbacks from CSCs of respective CSPs into single object, we label them accordingly with user defined thresholds and then those single objects of CSPs are grouped to their corresponding clusters.

The basic outline of this paper is as follows. Section 2 provides a brief overview of the related works. We present our motivation in section 3 and the problem definition in section 4. Then, section 5 comprises of description of the proposed method. Section 6 discusses the experiments and results of the proposed clustering algorithm used to cluster the dataset. Finally, we summarize and conclude in section 7.

II. Related Work

Various methods of trust management for cloud, based on feedback ratings have been introduced in the recent past. In [17], a resource broker for cloud resources is being used to evaluate the resources' trustworthiness by taking into account the security levels, user's feedback values and the performance criteria. A distributed framework that helps a CSC in assigning weight to feedbacks of different raters of prospective CSPs for accurate and fair assessment of service reputation is presented in [22]. But eventually, their model might face problem with the increase in size of feedback repository. Trust model named Application-oriented Remote Verification Trust Model (ARVTM) [23], dynamically alters the users' trust value to guarantee the security of information resources with the trust feedback mechanism to determine whether to or not to provide the requested resource or service. The trust collection part of these models consist of credential database and application information database. A new trust management architecture by Muchahari et al. [24], uses credible feedbacks of CSCs and CSPs to calculate trust level from a repository.

Noor et al. [25] introduce distributively managed trust feedbacks, collected from CSCs for their method. The proposed trust management method of Chong et al. [26], make use of feedbacks ratings acquired from trading partners after the fruitful completion of the transaction. User feedbacks have also been used in cloud for privacy management [27].

But most of the work emphasize on the issue of credibility of the feedbacks. Feedbacks repository is obvious to increase in view of the current trend of cloud computing growth. Dealing with different parameters or attributes of cloud trust is also necessary. Clustering approach can be used to address issues in multivariate trust analysis in cloud computing. To the best of our knowledge, nobody has tried to apply clustering on cloud feedbacks trust data.

III. Motivation

In a dynamic and risky environment like cloud, trust though a human notion, plays a mandatory role [28]. Number of CSPs with varying offers are increasing, making the selection of a reliable provider a daunting task. But despite the benefits of trust management, considerable issues related to general trust assessment mechanisms, distrusted feedbacks, poor identification of feedbacks, privacy of participants and the lack of feedbacks integration still need to be addressed [29].

Feedback-based datasets usually contain numeric ratings given by users on some items or services from past experiences. Trust values can be calculated as cumulative or average results of these user ratings. But this type of calculation can become prey to non-legitimate and insufficient feedbacks. Increasing number of CSPs and CSCs will increase the number of feedbacks, increasing the size and complexity of dataset. To overcome such an issue, we need an effective mechanism to calculate trust value from the feedbacks ratings.

Unsupervised method seems apt for differentiating between trustworthy and untrustworthy CSPs from those dataset. Cluster analysis, an unsupervised learning method is for grouping objects of similar kind into their respective group. Clustering algorithms are for revealing effective but unknown classes of items, in other words it deals with finding meaningful structure in a compilation of unlabeled data. Clustering is necessary when no labeled data are available regardless of whether the data are binary, categorical, numerical, interval, ordinal, relational, textual, spatial, temporal, spatio-temporal, image, multimedia, or mixtures of the above data types [30]. Clustering techniques have been used in improving recommendation accuracy in social networks [31]. Gupta et al. [32] applied cluster-based methods for web trust scrutiny. Similar work can be seen in [33]. Pitsilis et al. [34] in their paper through experiments using clustering algorithms revealed that the trust information which people express explicitly for the people they know can be useful for improving the accuracy of their recommendations. In [35], a clustering approach is proposed and studied in two application scenarios: academic venue recommendation based on collaboration information and trust-based recommendation. Now unlabeled trust dataset being available motivates us to propose an unsupervised method to cluster similar trust valued CSPs based on consumers' feedbacks. Moreover different correlation measures can be used to group CSPs having similar feedback rating values. It is possible that the rating values for different parameters of a CSP might have identical values. If we compute correlation between any two objects where one object contains identical values and the other object contains non-identical values then the widely used correlation measures like Pearson [36], Spearmen [37] and Kendall [38] may yield undefined correlation value as the example shown in Table 1. In such a situation clustering methods cannot work efficiently. To overcome this correlation based clustering problem, we proposed a new correlation measure called M_{dev} that handles the undefined correlation values. Effective handling of voluminous trust data accumulated over time without prior knowledge to support trustworthy CSP selection is the main motivation of this work.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

Table 1: Comparison of M_{dev}, Pearson, Spearman and Kendall Correlation

Sl. No.	Row Values	M_{dev}	Pearson	Spearman	Kendall
1	1 2 1 4 5 5 3 1 2 1 5 5 5 5 5 5 5 5 5 5 5 5	0.4600	NaN	NaN	NaN
2	1 2 1 4 5 5 3 1 2 1 5 1 1 1 3 5 1 1 1 1 1 1	0.5400	0.5179	0.4749	0.4446
3	1 2 1 4 5 5 3 1 2 1 5 5 5 5 5 5 5 5 5 5 5 5	0.4600	NaN	NaN	NaN
4	1 2 1 4 5 4 3 1 4 1 5 5 5 5 5 5 5 5 5 5 5 5	0.3500	NaN	NaN	NaN
5	1 2 5 4 5 5 2 3 1 1 5 5 5 5 5 5 5 5 5 5 5 5	0.5400	NaN	NaN	NaN
6	5 2 2 3 5 4 4 3 4 1 2 5 5 5 5 5 5 5 5 5 5 5	0.3600	NaN	NaN	NaN
7	1 2 1 4 5 5 3 1 2 1 5 1 2 1 4 5 5 3 1 2 1 5	1	1	1	1
8	1 2 1 4 5 4 3 1 4 1 5 1 1 1 3 5 1 1 1 1 1 1	0.5443	0.5138	0.5168	0.4788
9	5 2 1 3 5 5 2 3 1 1 5 5 5 5 5 5 5 5 5 5 5 5	0.5601	NaN	NaN	NaN
10	1 2 1 4 5 4 3 1 4 1 5 1 2 1 4 5 5 3 1 2 1 5	0.8638	0.9160	0.9415	0.8889
11	1 2 1 4 5 5 3 1 2 1 5 3 3 3 3 3 3 3 3 3 3 3	0.1925	NaN	NaN	NaN
12	2 1 5 3 1 1 3 3 3 2 5 5 5 5 5 5 5 5 5 5	0.3900	NaN	NaN	NaN
13	2 1 5 3 1 1 3 3 3 2 1 1 1 1 1 1 1 1 1 1	0.5000	NaN	NaN	NaN
14	3 5 4 3 3 2 1 5 5 5 5 5 5 5 5 5 5 5 5 5	0.5900	NaN	NaN	NaN
15	3 5 4 3 3 2 1 5 5 5 1 1 1 1 1 1 1 1 1 1	0.3407	NaN	NaN	NaN
16	2 4 2 4 5 3 3 1 4 5 4 4 4 4 4 4 4 4 4 4	0.4429	NaN	NaN	NaN
17	5 5 4 5 5 5 5 5 2 5 4 4 1 5 1 4 2 2 3 2 1 4 4 2 2 1 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5	0.4400	NaN	NaN	NaN
16	5 5 4 5 5 5 5 5 2 5 4 4 1 5 1 4 2 2 3 2 1 4 4 2 2 1 5 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0.2200	0.2002	0.2225	0.2029
19	5 5 5 5 5 5 2 5 2 1 4 4 5 5 3 4 5 5 5 2 5 5 4 4 1 1 3 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	0.1848	NaN	NaN	NaN
20	5 5 4 5 5 5 5 2 5 2 1 4 4 3 4 3 4 2 2 1 2 5 5 4 5 1 4 5 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4	0.3050	NaN	NaN	NaN

IV. Problem Definition

Feedback ratings of different trust parameters on CSPs given by CSCs are not enough to make the right selection. The task becomes more difficult with the increasing size and complexity of label devoid datasets. A data-driven approach should be suitable in making decisions for selection of trustworthy CSPs. Clustering approach can be applied when no labeled data are available. In feedback-based trustworthy CSP selection, the problem is to identify two or more clusters from a given dataset $D = \{O_1, O_2, O_3, \dots, n\}$. An object $O_i \in D$; $\forall i = 1, 2, 3, \dots n$, consist of m numbers of trust parameters that is $O_i = \{P_1, P_2, P_3, \dots P_m\}$ where each parameter P_j; $\forall j = 1, 2, 3, \dots, m$ represents feedbacks of CSPs. We divide D into k number of clusters $C_1, C_2, \dots, C_k$ to represent trust levels {Trustworthy, Untrustworthy …}. The objective is to develop a trustworthy CSP selection method using unsupervised learning approach to achieve best possible selection.

V. Proposed Method

A partitional clustering algorithm [39], named Min-Max Head (MMH) clustering is proposed to group different objects based on closeness to the dynamically calculated cluster heads. A new correlation measure called Mean-Dev M_{dev} Correlation is used to compute the closeness between a cluster head and an object. As the dataset consists of several feedback ratings for each CSP, we introduce an algorithm called Feedback Rating Aggregation (FRA) to aggregate the feedbacks of each CSPs, form single object for the respective CSPs and label them with respect to a user defined threshold. We have an Initial Cluster Head Computation (ICHC) algorithm for generating the initial cluster heads with which the very first object is compared for clustering using MMH algorithm for trustworthiness. The required number of initial cluster heads will be equal to the number of clusters to be formed. We keep fixed maximum and minimum parameter values and compute the evenly distributed mean values to form the remaining in between cluster heads. The framework of the proposed method is shown in Figure 1. To describe the different algorithms, we use symbols as shown in Table 2.

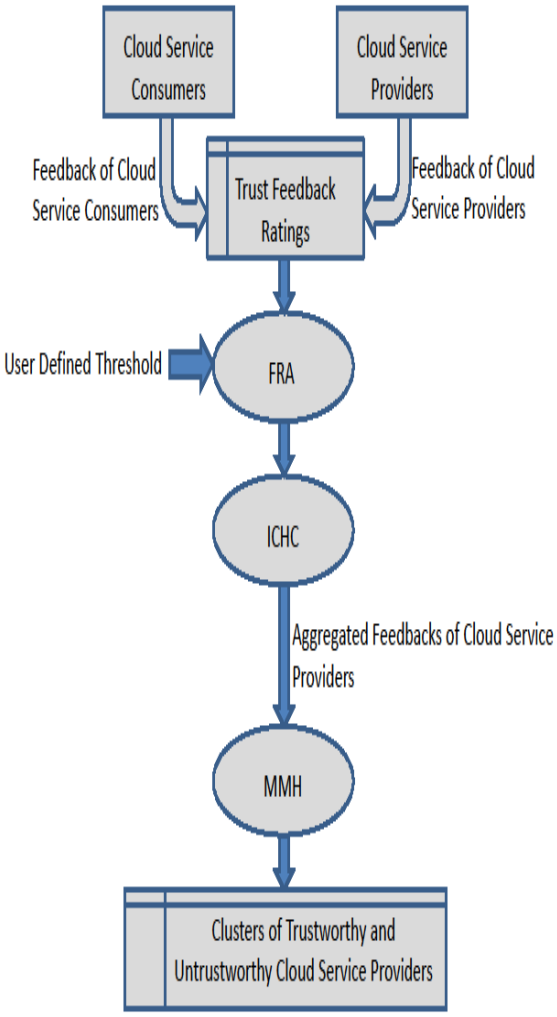

Figure 1: Proposed Framework

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

Table 2: Symbol table

Symbol	Meaning
D	Dataset
k	Number of clusters
n	Total number of objects of D
IH_k	k number of initial class heads
D_{ai}	*ith* attribute of D
O_i	Objects from D where $i = 1, 2, \ldots n$
C_k	*kth* number of clusters
CV_k	Correlation of $(O_i; IH_k)$ for *kth* cluster
C_m	Mean of objects in cluster C
CH_k	*kth* cluster head

A. M_{dev} Correlation

The feedback rating values of any rating datasets are generally within the range 1-5, 1-7 or 1-10, which is small. Due to small range the possibility of having identical feedback rating value for all the parameter is high. So, we need an effective and precise correlation measure to identify the highly correlated objects that have subtle difference among the parameter values. We propose an effective correlation measure that computes the number of exactly matching pair of parameter values and the deviation of each pair of parameter values from the mean value of their respective objects. To correlate any object and the cluster heads, we separate similar and dissimilar parameter values and calculate the deviation of the dissimilar values from mean then add it to the similar values using equation (1).

$$M_{dev}(a,b) = \begin{cases} 1, if\ S_{ab} = 0 \\ \left| \frac{S_t}{N_p} + \left(\frac{S_{ab}}{(N_p - S_t) \times M_{ab}} \right) \right|, otherwise \end{cases} \quad (1)$$

$$S_{ab} = \sum_{i=1}^{n} (|a_i - b_i|) \quad (2)$$

$$M_{ab} = \sum_{i=1}^{n} \left(|m^a - a_i| + |m^b - b_i| \right) \quad (3)$$

where, a_i and b_i are the ith parameter value of objects a and b respectively for n number of parameters. m^a and m^b are the mean of parameter values of a and b respectively. N_p is the number of parameters of an object and S_t is the number of common parameter values for both a and b. S_{ab} is the sum of absolute L_1 distance of every pair of object and M_{ab} total of sum of absolute difference between means and two respective objects given by equation (2) and (3) respectively.

1) *Properties of M_{dev}*: The proposed correlation measure satisfies the following properties.

   a) *Identity:* The M_{dev} correlation between two identical objects is always 1 i.e., $M_{dev}(a,a) = 1\ and\ M_{dev}(b,b) = 1$

   Proof: We know,

   $$S_{aa} = \sum_{i=1}^{n} (|a_i - a_i|) = 0$$

   Similarly,

   $$S_{bb} = 0$$

   Hence,

   $$M_{dev}(a,a) = 1\ and\ M_{dev}(b,b) = 1$$

   b) *Limit:* The M_{dev} correlation value range from 0 to 1.

   Proof: Equation $\left| \frac{S_t}{N_p} + \left(\frac{S_{ab}}{(N_p - S_t) \times M_{ab}} \right) \right|$, being absolute will be always positive (or zero). So,

   $$\left| \frac{S_t}{N_p} + \left(\frac{S_{ab}}{(N_p - S_t) \times M_{ab}} \right) \right| \geq 0$$

   If $a = b\ then\ S_{ab}\ hence, \frac{S_{ab}}{(N_p - S_t) \times M_{ab}} = 0$

   So, $\frac{S_t}{N_p} = 1$

   Therefore, M_{dev} correlation value range from 0 to 1.

   c) *Non-negativity:* The M_{dev} correlation yields positive correlation values.

   Proof: In equation $\left| \frac{S_t}{N_p} + \left(\frac{S_{ab}}{(N_p - S_t) \times M_{ab}} \right) \right|$, S_t and N_p cannot be negative but the value of S_{ab} and M_{ab} can be negative. As, we have taken the absolute value for both S_{ab} and M_{ab}, so they will be always positive. Hence,

   $$\left| \frac{S_t}{N_p} + \left(\frac{S_{ab}}{(N_p - S_t) \times M_{ab}} \right) \right| \geq 0$$

   d) *Symmetricity:* For any two objects a and b, $M_{dev}(a,b) = M_{dev}(b,a)$

   Proof: For any two objects a and b,

   $$\begin{aligned} S_{ab} &= \sum_{i=1}^{n} (|a_i - b_i|) \\ &= \sum_{i=1}^{n} (|b_i - a_i|) \\ &= S_{ba} \end{aligned}$$

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

Similarly,

$$M_{ab} = \sum_{i=1}^{n}(|m^a - a_i| + |m^b - b_i|)$$
$$= \sum_{i=1}^{n}(|m^b + b_i| + |m^a - a_i|)$$
$$= M_{ba}$$

So,

$$M_{dev}(a,b) = 1, if\ S_{ab} = 0$$
$$M_{dev}(b,a) = 1, if\ S_{ba} = 0$$

Since, $S_{ab} = S_{ba} \neq 0$

$$M_{ab} = M_{ba} \neq 0$$

Hence, $M_{dev}(a,b) = M_{dev}(b,a)$

e) *Triangular Inequality:* For any three objects *a*, *b* and *c*, to prove that

$$|M_{dev}(a,b)| \leq |M_{dev}(a,c)| + |M_{dev}(b,c)|$$

Proof: For generality, we consider the following three cases:

i. Case 1: M_{dev} (a, b), M_{dev} (a, c) and M_{dev} (b, c) are all positive.

ii. Case 2: M_{dev}(a, b), M_{dev}(a, c) and M_{dev}(b, c) are all negative

iii. Case 3: Either of M_{dev}(a, b), M_{dev}(a, c) and M_{dev}(b, c) is negative and the others are negative

i. Case 1:

Since,

$$M_{dev}(a,b) > 0, M_{dev}(a,c) > 0, M_{dev}(b,c) > 0$$

So,

$$S_{ab} > 0, M_{ab} > 0;\ S_{ac} > 0, M_{ac} > 0;\ S_{bc} > M_{bc}$$

And,

$$(M_{dev}(a,c) + M_{dev}(b,c)) > 0$$

Hence,

$$|M_{dev}(a,b)| < |M_{dev}(a,c) + M_{dev}(b,c)|$$

ii. Case 2

Since,

$$M_{dev}(a,b) < 0, M_{dev}(a,c) < 0, M_{dev}(b,c) < 0$$

So,

$$S_{ab} < 0, M_{ab} < 0;\ S_{ac} < 0, M_{ac} < 0;\ S_{bc} < M_{bc}$$

And

$$M_{dev}(a,b) < 0;\ (M_{dev}(a,c) + M_{dev}(b,c)) \leq 0$$

Therefore,

$$|M_{dev}(a,b)| \leq |M_{dev}(a,c) + M_{dev}(b,c)|$$

iii. Case 3:

Let,

$$M_{dev}(a,b) > 0, M_{dev}(a,c) < 0, M_{dev}(b,c) > 0$$

So,

$$S_{ab} > 0, M_{ab} > 0;\ S_{ac} < 0, M_{ac} < 0;\ S_{bc} > 0, M_{bc} > 0$$

And,

$$\left(M_{dev}(a,c) + M_{dev}(b,c)\right) \geq 0$$

Therefore,

$$|M_{dev}(a,b)| \leq |M_{dev}(a,c) + M_{dev}(b,c)|$$

Similarly, for either two objects being negative, we can say that

$$|M_{dev}(a,b)| < |M_{dev}(a,c) + M_{dev}(b,c)|$$

B. FRA Algorithm

The main purpose of FRA algorithm is to aggregate the multiple feedback ratings from CSCs for different CSPs. There are multiple feedbacks for *n* unique CSPs in dataset *D*. Prior to clustering of *n* CSPs, we aggregated those feedbacks into single object and labelled them based on a user-defined thresholds θ to form dataset D_a. Threshold θ can be any rating value from the dataset D_a. We compute *M*, which is the sum of maximum number of occurrence of all unique parameter P_j values divided by total number of parameters *m*. To label the CSPs, we consider the value of $\theta=1$ to represent a trustworthy CSP and the value of $\theta=2$ as untrustworthy. Labels can be more in numbers with respect to θ.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

Algorithm 1: FRA Algorithm

Data: D: original dataset, θ: user defined threshold
Result: D_a: aggregated dataset
begin
  Find total n number of unique CSPs in D;
  foreach $CSP_i, \forall i = 1, 2, \ldots, n$ **do**
    Find total number of feedbacks of CSP_i for m number of parameters;
    foreach *parameter* $P_j, j = 1, 2, \ldots, m$ **do**
      Find unique values from P_j;
      Find the number of occurrence of each unique P_j values;
      D_a^j= Put the maximum occurrence of a unique P_j values;
    end
    Compute $M = \frac{\sum_{j=1}^{m}(D_a^j)}{m}$;
    if $M \geq \theta$ **then**
      $D_a^{j+1} = 1$ (Trustworthy), $(j+1)$ is the label for CSP_i;
    else
      $D_a^{j+1} = 2$ (Untrustworthy), $(j+1)$ is the label for CSP_i;
    end
  end
end

C. ICHC Algorithm

ICHC algorithm is meant for computation of initial cluster heads. The maximum and the minimum value of all the feedback ratings *m* of *ith* attribute of dataset *D* is taken as respectively the first IH_1 and the last IH_k heads. Number of cluster IH_k is equal to the number of cluster *k*. Depending on *k*, if the number is more than two then the in-between value becomes the intermediate head(s).

Algorithm 2: ICHC Algorithm

Data: D, k
Result: IH_k clusters
begin
  Take $\bigcup_{i=1}^{m} \max(D_{a_i})$ as IH_1;
  Take $\bigcup_{i=1}^{m} \min(D_{a_i})$ as IH_k;
  if $k > 2$ **then**
    Find intermediate number of clusters j between IH_1 and IH_k, $j = k - 2$;
    for $l = 1$ *to* j **do**
      $IH_{l+1} = \left(\frac{IH_l + IH_{l+2}}{2}\right)$;
    end
  end
end

D. MMH Clustering Algorithm

In MMH clustering, correlation values are calculated between every object of the dataset and the cluster heads. Generated initial cluster heads IH_k is compared with the very first CSP object to group into *k* clusters. Every object O_i having maximum correlation value CV_k is put to their respective clusters C_k. After successive iterations, cluster heads CH_k are updated by the mean values C_m of the corresponding cluster. If correlation value is similar, $CV_j = CV_{j+1}$ then Euclidean distance is calculated and objects are put accordingly in their appropriate clusters based on minimum Euclidean distance.

Algorithm 3: MMH Clustering Algorithm

Data: D, k
Result: k clusters
begin
  Compute initial k cluster heads IH_k (Using algorithm 2);
  forall *objects* $O_i \in D$ **do**
    forall *cluster* C_k **do**
      if *number of object in* $C_k == null$ **then**
        CV_k=Compute correlation (O_i, IH_k) (Using equation 1);
        $IH_k = null$;
      else
        C_m=Compute mean of objects in C_k;
        $CH_k = C_m$;
        CV_k=Compute correlation (O_i, CH_k)(Using equation 1);
      end
      if $CV_j \neq CV_{j+1}, \forall j = 1, \ldots, k$ **then**
        Put O_i to cluster C_k for CV_j is maximum;
      else if $CV_j \geq max(\forall CV_k), j \neq k$ **then**
        if $IH_k == null$ **then**
          CV_j=Compute Euclidean distance $(O_i, CH_k) * 10$;
        else
          CV_j=Compute Euclidean distance $(O_i, IH_k) * 10$;
        end
        Put O_i to cluster C_k for CV_j is maximum;
      end
    end
  end
end

E. Complexity Analysis

The overall complexity of the proposed method is computed from the computational steps of MMH clustering and ICHC algorithm. The ICHC algorithm $O(n)$ times to determine k cluster heads from *n* objects of *D*. Similarly, MMH algorithm takes $O(n \times k)$ times to generate the final k clusters. As $O(n) + O(n \times k) \cong O(n \times k)$ and hence, the overall complexity of the proposed method is $O(n \times k)$.

VI. Experiments and Results

Experiments were carried out on a machine with 2.8GHz Intel i5 processor having 4GB main memory on 64bit Windows 8.1 operating system. We implemented our algorithm using Matlab 2015a software.

A. Dataset Used

Cloud trust dataset for research are now available [40], [41]. For various trust and reputation models, datasets from Epinions[2] and Ciao[3] are mostly used. Product review sites are capable of equipping the researchers with practical platform to study online trust [42]. We used two datasets namely Trust Feedbacks Dataset[4] by "Cloud Armor Project" and "rating" taken from the well-known product review website Ciao.

Trust Feedbacks Dataset by [43], was collected for their proposed 'Cloud Armor trust management service'. The publicly available 'Trust Feedbacks Dataset' have 10,000+ feedbacks based on cloud Quality of Service(QoS) attributes, given by more than 7,000 anonymized consumers to 113 real-world cloud services. The QoS attributes are availability,

[2] http://www.epinions.com
[3] http://ciao.co.uk
[4] http://cs.adelaide.edu.au/~cloudarmor/ds.html

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

security, response time, accessibility, price, speed, storage space, features, ease of use, technical support, customer service and level of expertise where all of them are of numeric type ranging value from 0 to 5.

In Ciao, users write reviews to rate items and establish trust networks with their like-minded users. “Rating” dataset of Ciao, also used in [44], [45] includes rating information with five columns like userid, productid, categoryid and rating. To mould the dataset according to our need, we considered userid as CSCs, productid as CSP, categoryid as cloud QoS attributes. We merged the rows with same userid and productid with different categoryid to form a single row of CSC, CSP and 28 different attributes.

The statistics of the datasets are shown in Table 3. As both the datasets have some missing values for the QoS attributes, we put random value ranging from 1 to 5 values against those missing attribute values.

Table 3: Statistics of Armor and Ciao dataset

Armor		**Ciao**	
No. of CSCs	7312	No. of users	7375
No. of CSPs	117	No. of products	106797
No. of QoS attributes	12	No. of category	28
Rating range	0-5	Rating range	0-5

For experimental purpose we used two synthetic dataset namely *Synthetic 1* and *Synthetic 2*. Both *Synthetic 1* and *Synthetic 2* were generated using random integer number function in Matlab with rating value range of 1-3 and 1-5 respectively. Statistics of both the synthetic datasets are given in Table 4 and the data distribution patterns are shown in Figure 2 and Figure 3.

Table 4: Statistics of synthetic dataset

	Synthetic 1	**Synthetic 2**
No. of CSCs	10080	20160
No. of CSPs	120	235
No. of QoS attributes	11	25
Rating range	1-5	1-5

Figure 2: Histogram of Synthetic 1

Figure 3: Histogram of Synthetic 2

B. Results

The performance of our method is evaluated on two real world datasets, viz., ``Cloud Armor", ``Ciao" and two synthetic dataset “*Synthetic 1*”, “*Synthetic 2*”.

Accuracy percentage is calculated by using equation (4)

$$Accuracy\ Percentage = \frac{S_c}{O_n} \times 100 \quad (4)$$

where S_c is the similarity count between actual and computed object levels and O_n is the total number of objects.

The method uses a user defined threshold θ to differentiate between trustworthy and untrustworthy CSPs. With varying number of threshold, θ= *{2, 2.5, 3, 3.5, 4}* and for two number of clusters, we could get high accuracy rate of 84.61% when θ=*2, 2.5* and 93.31% when θ=*2, 2.5* respectively for both the dataset as shown in Table 5.

Table 5: Accuracy result of Armor and Ciao

Dataset	**No. of Cluster**	**Threshold(θ)**	**Accuracy (%)**
Armor	2	2	84.61
	2	2.5	84.61
	2	3	83.76
	2	3.5	76.06
	2	4	51.28
Ciao	2	2	93.31
	2	2.5	93.31
	2	3	93.12
	2	3.5	47.81
	2	4	47.81

For analysing the quality of clusters, different internal and external clustering indices such as Silhouette Index [46], Davies-Bouldin Index [47], Rand Index [48] and Jaccard Index [49] are considered. Indices shown in Figure 4 and Figure 5, indicates that the ``Cloud Armor" sample is on or very close to the decision boundary between two neighboring clusters. With respect to internal indices, value of Silhouette is 0.36 and that of Davies-Bouldin index is 1.69 that suggests similar quality of two clusters against θ= *{2, 2.5, 3, 3.5, 4}*. For external indices,

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

maximum value of both Rand and Jaccard index is 0.73 suggesting 2 clusters to be optimal for $\theta = \{2, 2.5, 3\}$.

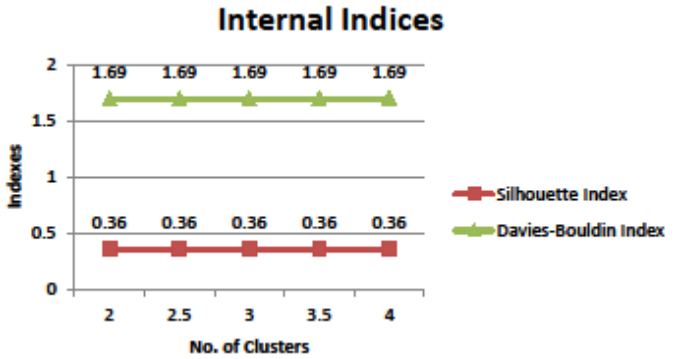

Figure 4: Maximum of Silhouette and minimum of Davies-Bouldin is best

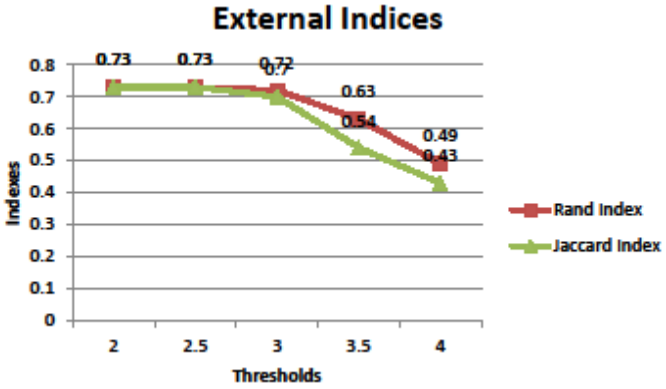

Figure 5: Maximum value for Rand and Jaccard is best

As shown in Table 6, the synthetic dataset "*Synthetic 1*" shows 80.83% accuracy with threshold $\theta=2$ for two clusters whereas "S*ynthetic 2*" yields 100% accuracy for $k=5$ clusters in threshold range $4.5 < \theta < 5$. For "*Synthetic 2*", all internal and external indices suggests 5 clusters as the optimal number as shown in Figure 6 and Figure 7.

Table 6: Accuracy result of Synthetic 1 and Synthetic 2

Dataset	No. of cluster	Threshold (θ)	Accuracy (%)
Synthetic 1	2	2	80.83
	2	2.5	75.50
	2	3	56.66
	2	3.5	34.16
	2	4	20.00
Synthetic 2	2	1.5<θ<2.5	91.84
	2	2.5<θ<3.5	92.76
	2	3.5<θ<4.5	91.48
	2	4.5<θ<5	100.00

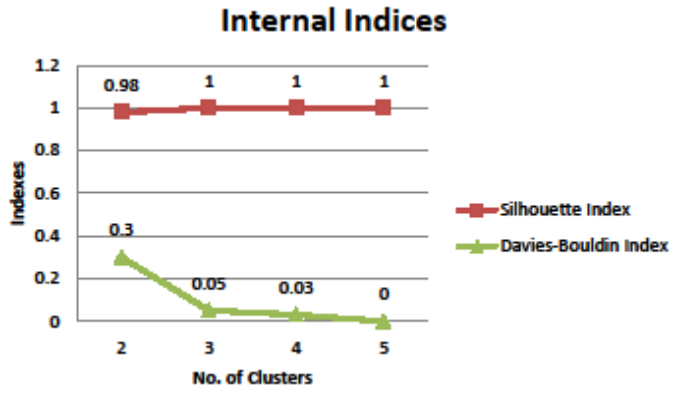

Figure 6: Silhouette and Davies-Bouldin indices of Synthetic 2

Figure 7: Rand and Jaccard indices of Synthetic 2

C. *Comparison of Clustering Algorithms*

The proposed algorithm is compared with k-means, Partitioning Around Medoids(PAM), Clustering Large Applications based upon Randomised Search (CLARANS), Hierarchical-Complete Linkage(H-CL) and Hierarchical-Average Linkage (H-AL) algorithm. The popularity and applicability are the reasons for selecting these algorithms. The accuracy of these three algorithms for $k=2$ and $\theta = \{2, 2.5, 3, 3.5, 4\}$ is given in Table 7. For "Cloud Armor", MMH shows better accuracy in all the considered algorithms in most of the threshold values. Dataset "*Synthetic 1*" being right skewed, MMH shows better accuracy for $\theta = \{2, 2.5\}$.

Table 7: Comparison of MMH, k-means, PAM, CLARANS, H-CL and H-AL algorithm using dataset Armor and Synthetic 1

			Accuracy (%) using clustering algorithm					
Dataset	No. of Cluster(k)	Threshold(θ)	MMH	k-means	PAM	CLARANS	H-CL	H-AL
Cloud Armor	2	2	84.61	74.35	23.93	36.75	14.52	12.82
	2	2.5	84.61	74.35	23.93	43.58	14.52	12.82
	2	3	83.76	78.63	16.23	33.33	18.80	20.51
	2	3.5	76.06	86.32	15.38	34.18	24.78	26.49
	2	4	51.28	61.53	40.17	45.29	49.57	51.28
Synthetic 1	2	2	80.83	65.00	56.66	47.50	25.83	16.66
	2	2.5	73.50	73.33	56.66	47.50	25.83	20.83
	2	3	66.66	77.50	51.66	53.33	35.00	25
	2	3.5	34.16	43.33	34.16	45	57.50	54.16
	2	4	20	39.16	20	53.33	70	71.66

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

D. Discussion

The performance of the MMH clustering algorithm is found satisfactory on both real-life and synthetic datasets. The method groups the CSPs into multiple clusters based on the user defined threshold θ. From the experimental results, we observed that when the value of θ is between 2 to 3 then the method yields high accuracy for two number of clusters and the cluster with the maximum value of cluster head being the trustworthy cluster. All indices showed almost similar cluster quality and compactness in "Cloud Armor" dataset for different thresholds. Similar to the real datasets, on "*Synthetic 1*" the method gives high accuracy within the threshold range of 2 to 3. In case of "*Synthetic 2*", if we consider two to five number of cluster then the method gives 100% accuracy for cluster five and the indices also suggests five as the optimal number of clusters. We may assume the five number of clusters to be "very untrustworthy", "untrustworthy", "moderate", "trustworthy", and "very trustworthy" as considered in [50]. The proposed M_{dev} correlation used in MMH algorithm yields preferred outcome. From the comparison of the proposed MMH algorithm with k-means, PAM, CLARANS, H-CL and H-AL algorithm, we found that in most of the cases, our clustering approach gives better accuracy on "Cloud Armor" and "*Synthetic 1*" dataset.

VII. Conclusion and Future Work

Trust is regarded as the prime obstacle for adoption and growth of cloud computing. CSCs' experiences against each QoS parameter can be used to appraise the CSPs [51]. Feedback rating based trust management have been explored by many researchers. But increasing number of CSPs and CSCs have increased the number of feedbacks which escalated the size and complexity of dataset. In this paper, we present a new MMH clustering algorithm using M_{dev} correlation measure to aid selection of trustworthy CSPs using feedback ratings. The proposed FRA algorithm is used to aggregate the multiple feedback ratings from CSCs for different CSPs. The clusters are formed based on the correlation values with respect to cluster heads, initial cluster heads being computed by ICHC algorithm. Based on different user defined thresholds, we could derive satisfactory results in differentiating between trustworthy and untrustworthy CSPs. Clustering with MMH gives better result than well-known k-means, PAM, CLARANS and hierarchical algorithm. In future, we intent to come up with a better method for labelling the cloud trust data.

References

[1] M. T. Khorshed, A. Ali and S. A. Wasimi, "A survey on gaps, threat remediation challenges and some thoughts for proactive attack detection in cloud computing," *Future Generation Computer Systems,* vol. 28, no. 6, pp. 833--851, 2012.

[2] R. Rezaei, T. K. Chiew, S. P. Lee and Z. S. Aliee, "A semantic interoperability framework for software as a service systems in cloud computing environments," *Expert Systems with Applications,* vol. 41, no. 13, pp. 5751--5770, 2014.

[3] Y. L. Lai, "Analyzing strategies of mobile agents on malicious cloud platform with Agent-Based Computational Economic Approach," *Expert Systems with Applications,* vol. 40, no. 7, pp. 2615--2620, 2013.

[4] W. Wang, G. Zeng, D. Tang and J. Yao, "Cloud-DLS: Dynamic trusted scheduling for Cloud computing," *Expert Systems with Applications,* vol. 39, no. 3, pp. 2321--2329, 2012.

[5] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica and others, "A view of cloud computing," *Communications of the ACM,* vol. 53, no. 4, pp. 50--58, 2010.

[6] S. Qamar, N. Lal and M. Singh, "Internet ware cloud computing: Challenges," *arXiv preprint arXiv:1004.1746,* 2010.

[7] A. Jula, E. Sundararajan and Z. Othman, "Cloud computing service composition: A systematic literature review," *Expert Systems with Applications,* vol. 41, no. 8, pp. 3809--3824, 2014.

[8] H. Mouratidis, S. Islam, C. Kalloniatis and S. Gritzalis, "A framework to support selection of cloud providers based on security and privacy requirements," *Journal of Systems and Software,* vol. 86, no. 9, pp. 2276--2293, 2013.

[9] M. D. Ryan, "Cloud computing security: The scientific challenge, and a survey of solutions," *Journal of Systems and Software,* vol. 86, no. 9, pp. 2263--2268, 2013.

[10] W.-W. Wu, "Developing an explorative model for SaaS adoption," *Expert systems with applications,* vol. 38, no. 12, pp. 15057--15064, 2011.

[11] A. Abdelmaboud, D. N. Jawawi, I. Ghani, A. Elsafi and B. Kitchenham, "Quality of service approaches in cloud computing: A systematic mapping study," *Journal of Systems and Software,* vol. 101, pp. 159--179, 2015.

[12] G. Drosatos, P. S. Efraimidis, I. N. Athanasiadis, M. Stevens and E. D'Hondt, "Privacy-preserving computation of participatory noise maps in the cloud," *Journal of Systems and Software,* vol. 92, pp. 170--183, 2014.

[13] N. Phaphoom, X. Wang, S. Samuel, S. Helmer and P. Abrahamsson, "A survey study on major technical barriers affecting the decision to adopt cloud services," *Journal of Systems and Software,* vol. 103, pp. 167--181, 2015.

[14] Y. Yu, J. Ni, M. H. Au, H. Liu, H. Wang and C. Xu, "Improved security of a dynamic remote data possession checking protocol for cloud storage," *Expert Systems with Applications,* vol. 41, no. 17, pp. 7789--7796, 2014.

[15] Z. Li, H. Zhang, L. O'Brien, R. Cai and S. Flint, "On evaluating commercial Cloud services: A systematic review," *Journal of Systems and Software,* vol. 86, no. 9, pp. 2371--2393, 2013.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

[16] K. M. Khan and Q. Malluhi, "Establishing trust in cloud computing," *IT professional*, vol. 12, no. 5, pp. 20--27, 2010.

[17] P. D. Manuel, S. Thamarai Selvi and M.-E. Barr, "Trust management system for grid and cloud resources," in *Advanced Computing, 2009. ICAC 2009. First International Conference on*, 2009.

[18] S. M. Habib, S. Ries and M. Muhlhauser, "Cloud computing landscape and research challenges regarding trust and reputation," in *Ubiquitous Intelligence & Computing and 7th International Conference on Autonomic & Trusted Computing (UIC/ATC), 2010 7th International Conference on*, 2010.

[19] S. M. Habib, S. Ries and M. Muhlhauser, "Towards a trust management system for cloud computing," in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2011 IEEE 10th International Conference on*, 2011.

[20] T. H. Noor, Q. Z. Sheng, A. H. Ngu and S. Dustdar, "Analysis of Web-Scale Cloud Services," *Internet Computing, IEEE*, vol. 18, no. 4, pp. 55--61, 2014.

[21] R. Dekker, "The importance of having data-sets," in *2006 IATUL Conference*, 2006.

[22] J. Abawajy, "Establishing trust in hybrid cloud computing environments," in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2011 IEEE 10th International Conference on*, 2011.

[23] X. Zhang, H. Liu, B. Li, X. Wang, H. Chen and S. Wu, "Application-oriented remote verification trust model in cloud computing," in *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*, 2010.

[24] M. K. Muchahari and S. K. Sinha, "A new trust management architecture for cloud computing environment," in *Cloud and Services Computing (ISCOS), 2012 International Symposium on*, 2012.

[25] T. H. Noor and Q. Z. Sheng, "Trust as a service: a framework for trust management in cloud environments," in *Web Information System Engineering--WISE 2011*, Springer, 2011, pp. 314--321.

[26] S. K. Chong, J. Abawajy, M. Ahmad and I. R. A. Hamid, "Enhancing Trust Management in Cloud Environment," *Procedia-Social and Behavioral Sciences*, vol. 129, pp. 314--321, 2014.

[27] M. Mowbray and S. Pearson, "A client-based privacy manager for cloud computing," in *Proceedings of the fourth international ICST conference on COMmunication system softWAre and middlewaRE*, 2009.

[28] M. K. Muchahari and S. K. Sinha, "A Survey on Web Services and Trust in Cloud Computing Environment," in *National Workshop on Network Security 2013, Tezpur University*, Tezpur, 2013.

[29] T. H. Noor, Q. Z. Sheng, S. Zeadally and J. Yu, "Trust management of services in cloud environments: Obstacles and solutions," *ACM Computing Surveys (CSUR)*, vol. 46, no. 1, p. 12, 2013.

[30] T. Warren Liao, "Clustering of time series data - a survey," *Pattern recognition*, vol. 38, no. 11, pp. 1857--1874, 2005.

[31] T. DuBois, J. Golbeck, J. Kleint and A. Srinivasan, "Improving recommendation accuracy by clustering social networks with trust," *Recommender Systems \& the Social Web*, vol. 532, pp. 1--8, 2009.

[32] M. Gupta, Y. Sun and J. Han, "Trust analysis with clustering," in *Proceedings of the 20th international conference companion on World wide web*, 2011.

[33] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng and T. Wu, "Rankclus: integrating clustering with ranking for heterogeneous information network analysis," in *Proceedings of the 12th International Conference on Extending Database Technology: Advances in Database Technology*, 2009.

[34] G. Pitsilis, X. Zhang and W. Wang, "Clustering recommenders in collaborative filtering using explicit trust information," in *Trust Management V*, Springer, 2011, pp. 82--97.

[35] M. C. Pham, Y. Cao, R. Klamma and M. Jarke, "A Clustering Approach for Collaborative Filtering Recommendation Using Social Network Analysis," *J. UCS*, vol. 17, no. 4, pp. 583--604, 2011.

[36] K. Pearson, "Note on regression and inheritance in the case of two parents," in *Proceedings of the Royal Society of London*, 1895.

[37] A. Lehman, JMP for basic univariate and multivariate statistics: a step-by-step guide, SAS Institute, 2005.

[38] M. G. Kendall, Rank correlation methods, Griffin, 1948.

[39] R. Xu, D. Wunsch and others, "Survey of clustering algorithms," *Neural Networks, IEEE Transactions on*, vol. 16, no. 3, pp. 645--678, 2005.

[40] T. H. Noor, Q. Z. Sheng, A. Alfazi, A. H. Ngu and J. Law, "CSCE: A Crawler Engine for Cloud Services Discovery on the World Wide Web," in *Web Services (ICWS), 2013 IEEE 20th International Conference on*, 2013.

[41] T. H. Noor, Q. Z. Sheng and A. Alfazi, "Reputation Attacks Detection for Effective Trust Assessment Among Cloud Services," in *Trust, Security and Privacy in Computing and Communications (TrustCom), 2013 12th IEEE International Conference on*, 2013.

[42] P. De Meo, E. Ferrara, D. Rosaci and G. M. Sarne}, "Trust and compactness in social network groups," *Cybernetics, IEEE Transactions on*, vol. 45, no. 2, pp. 205--216, 2015.

[43] T. H. Noor, Q. Z. Sheng, A. H. Ngu, A. Alfazi and J. Law, "Cloud Armor: a platform for credibility-based trust management of cloud services," in *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, 2013.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500

[44] J. Tang, H. Gao and H. Liu, "mTrust: discerning multi-faceted trust in a connected world," in *Proceedings of the fifth ACM international conference on Web search and data mining*, 2012.

[45] J. Tang, H. Gao, H. Liu and A. Das Sarma, "eTrust: Understanding trust evolution in an online world," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012.

[46] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53--65, 1987.

[47] N. R. Pal and J. Biswas, "Cluster validation using graph theoretic concepts," *Pattern Recognition*, vol. 30, no. 6, pp. 847--857, 1997.

[48] W. M. Rand, "Objective criteria for the evaluation of clustering methods," *Journal of the American Statistical association*, vol. 66, no. 336, pp. 846--850, 1971.

[49] R. Real and J. M. Vargas, "The probabilistic basis of Jaccard's index of similarity," *Systematic biology*, pp. 380--385, 1996.

[50] M. Tavakolifard, S. J. Knapskog and P. Herrmann, "Trust transferability among similar contexts," in *Proceedings of the 4th ACM symposium on QoS and security for wireless and mobile networks*, 2008.

[51] S. M. Habib, S. Hauke, S. Ries and M. Muhlhauser, "Trust as a facilitator in cloud computing: a survey," *Journal of Cloud Computing*, vol. 1, no. 1, pp. 1--18, 2012.

https://sites.google.com/site/ijcsis/
ISSN 1947-5500