Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org

Volume 23, Issues 4, pp. 303–319, DOI 10.12694/scpe.v23i4.2034

CLOUD BROKER RECOMMENDATION FRAMEWORK TO PROVIDE TRUSTWORTHY CLOUD SERVICES TO THE END USER

M. MARIMUTHU[*], J. AKILANDESWARI [†], VARASREE B[‡], RAJESH KUMAR GUNUPUDI[§] AND SOMULA RAMASUBBAREDDY[¶]

Abstract. In recent years, many cloud services have become available on the Website. Discovering suitable cloud services for the end user is incredibly complex and difficult. The cloud brokerage service is an application that aids in providing solutions for this problem. It recommends suitable cloud service providers to the end users depending on their relevant requirements. The Internet provides access to a wide variety of cloud brokers. As a result, choosing a cloud broker or service provider is both time-consuming and tedious. It is now becoming a necessity to choose a proper cloud brokerage service based on trust. Research works found in the literature address some of the issues and provide feasible solutions by proposing frameworks, optimizations and rule based algorithms. However, those works focus solely on delivering a trustworthy service to the end user through application of techniques and algorithms. There is no proper framework model in place to provide suitable and trustworthy recommended services to the users. This article provides a detailed description of the frameworks that are offered by the researchers, including issues and proposes a trustworthy recommendation framework (TRF) to provide trustworthy services to the end user. This article also presents a Trustworthy Recommended Weighted value (TRWv) approach for determining trustworthy services, and it is discovered that the proposed method achieves high accuracy (91.3%) when compared to similar works.

Key words: cloud broker, cloud service provider, end user, recommendation framework, logistic regression

AMS subject classifications. 62J05

1. Introduction. In cloud computing, diverse computer system resources (networks, servers, storage, applications and services) are shared and configured automatically with less administrative effort as and when required by the end user. It is a new technology that provides end users with utility services as resources on demand. Computing resources is symbolized by on-demand self-service, extensive network connectivity, rapid elasticity, resource pooling and measured service. First and foremost, identify the most suitable cloud deployment model or architecture to implement cloud services [1]. Platform as a service (PaaS), Infrastructure as a service (IaaS), and Software as a service (SaaS) are the various types of cloud computing services. It is conceivable to contract IT resources such as networks, servers, storage, virtual machines and operating systems, from a cloud service provider (CSP) via IaaS. Platform as a service provides a reliable environment which enables the user to build, evaluate, distribute, and also to perform various administrative tasks with respect to an application. On demand, subscription-based access to software programmers via the Internet is referred to as SaaS.

The deployment paradigms are configured as public cloud, private cloud, hybrid cloud, and community cloud [2, 3]. Public clouds, typically disseminate computing resources such as servers and storage through the Internet, are initiated and operated by the trusted intermediate CSP. The cloud infrastructure is accessible to the general public. Private cloud is a cloud storage service, which is owned and monitored solely by a single organization via private network and it is physically hosted at an onsite premises. Some organizations pay for intermediate service providers to set-up their private clouds. A hybrid cloud is consists of two or more distinct clouds (private, public, and community) that are combined together which permits transfer of data and applications between them. A hybrid cloud increases the organization's flexibility by enabling the migration of

[*]Assistant Professor, Department of Computer Science and Engineering, Sona College of Technology (mari.btech@gmail.com).
[†]Professor, Department of Information Technology, Sona College of Technology (akilandeswari@sonatech.ac.in)
[‡]Information Technology, Institute of Aeronautical Engineering, Hyderabad, India.(b.varasree@iare.ac.in)
[§]Information Technology, VNRVJIET, Hyderabad, India.(gunupudirajesh@gmail.com)
[¶]Information Technology, VNRVJIET, Hyderabad,India.(svramasubbareddy1219@gmail.com)

data and applications between private and public clouds. Community cloud infrastructure is used by several organizations to achieve a common goal.

Cloud Service providers (CSPs) are vendors that provide Information Technology (IT) as a service via the Internet. Many CSPs are available online. Therefore finding a CSP is an exhausting task for end users [1, 4]. Majority of CSPs provide identical functionality, which creates the service selection problems for end users. Selecting the best service provider requirements is a challenging and time consuming job. To solve this issue, Cloud Broker (CB) plays a vital role in providing an optimal CSP to the end user. The CB has been considered as a major concern for emerging cloud technology. A CB is a firm that oversees the usage, efficiency, and distribution of cloud services, as well as negotiates contracts among CSPs and end customers. CB acts as a go-between for CSPs and end users. It provides three categories of services namely

Service Intermediation: Enables value-added services or enhances functionality, such as controlling cloud access.

Service Aggregation: A CB combines and integrates multiple services into one or more new services. It facilitates data integration and ensures data security while transporting data between a end user and several CSPs.

Service Arbitrage: In order to create new services, a CB combines various kinds of services. It is similar to aggregation, except aggregation is not fixed. It has the ability to choose the CSPs based on the data features.

CBs provide a variety of services, including application-to-application (A2A), business-to-business (B2B), and trade partner relationship management. The desired services are not provided by all CSPs to end users. The user's perspective is fully focused on selecting the precise CB, major concerns in selecting the precise cloud broker are trustworthiness and recommended services as trusted ones. [5] It is also an efficient method for assessing and establishing a relationship between a service provider and user [6, 7, 41].

Making a decision or recommending a cloud services in an inter-cloud environment is not an ordinary task. The unavailability of precise information like QoS features, TMP and trustworthy further added to the issue. In a cloud environment, the resource selection procedure is incredibly difficult and less reliable due to the immense complexity and constraints of existing methodologies [8]. In this context, manual approaches will be inefficient and time-consuming; therefore, automated recommendation systems are required to assist the consumer in choosing the best cloud services [9, 10].

An effective trust management framework provides optimal as well as trustworthy services to end users. However, in a cloud context, trust assessment is one of the most challenging tasks [11]. Most of the frameworks, algorithms, approaches and methodologies determine the trustworthiness of the cloud brokerage service or service provider to provide trusted services. The selection of cloud services falls into one of four categories: decision-making, prediction-based, approach-based, and optimization techniques [12]. Approach based methodologies are Quality of service (QoS) ranking algorithms or models based on subjective or objective assessment or integrating both. Decision making is the approach for identifying the best CSP depending on the end user's needs. Optimization is a technique used to identify the appropriate CSPs. Linear programming and predictive techniques are used to forecast events. If the cloud user does not have any idea about the service, the feedback derived from the historical data that depends on the service provider is used. By using this approach, the best service provider can be predicted [13, 14]. Machine learning algorithms, statistical algorithms and data mining algorithms are the few examples of prediction. There are other methodologies to identify the best CSP for the end user. First order logic, Computation techniques, Fuzzy set theory, Fuzzy ontology prioritized aggregation operator and ranking oriented prediction are some of them.

All the research articles only focused on trustworthy service using the above techniques, but none of the research articles concentrate much on the framework. The framework provides a complete idea of the trustworthy services to the end user and, by using this framework model, can incorporate any of the technologies listed above. This research article provides a detailed survey of the framework models available, and proposes a new Trustworthy Recommendation Framework (TRF) model for selection of services in the cloud environment using the Trustworthy Recommended Weighted value (TRWv) approach, which also provides trustworthy services to end users and is suitable for real-time environments.

The remaining sections of the article are organized as follows: Section 2 presents the overall perspective of relevant work on the framework model. The proposed trustworthy recommendation framework model to enable a recommended trustworthy cloud services to the end user is outlined in Section 3. Section 4 elaborates the results that are compared with other framework models available in the literature. The conclusion and future developments of the research study are addressed in Section 5.

2. Related Work. Research has shown that the provision of trust services to end users remains a challenge because of their privacy, protection, reliability, availability and dynamically distributed environment. Most of the articles used service provider capabilities and end-user reviews to identify trustworthy cloud services. Similar services are identified for end users in a cloud environment based on the availability of such services in a dynamic nature or depending on the quality of services identified for the end user. In this section of the literature, a comprehensive description of the trust framework model available in the current scenario has been presented and, finally, feedback on the framework model has been given.

Somu et al. [15] proposed a TrustCom – a novel trust assessment framework for identifications of the trustworthy services. Selection of services is determined by the level of trust, the accuracy of trust value based on Trust Measure Parameters (TMPs) including availability, accountability, and cost. Hypergraph-based computation model was utilised to combine Rough set theory (RST) and existing features for feature selection when processing huge datasets because RST has constraints with respect to all subsets for picking the optimal feature subset. The cloud services are ranked based on the trust computation value. Noor et al. [16] presented a web service (WS)-based trust management in cloud environment. Many solutions have been proposed to identify the trust review/feedback, however establishing the legitimacy of user feedback has been neglected. Dynamic cloud environment is also very difficult to predict service availability. The author introduced an Adaptive credibility model (framework) that is used to figure out credible and malicious feedback. Once again, trust depends on the availability of services in the cloud environment.

Devi et al. [17] proposed a Linear Programming (LP) model which is used to order the cloud services dynamically. Due to the availability of many CSPs, it is difficult to choose the optimal cloud service. The quantitative and qualitative approaches are used to predict the service provider for end users based on QoS requirements. As per requirements given from the end user, the weightage is assigned. So the maximum weight is assigned to the highest order of the requirements to evaluate the rank based on objective method. The ranking score of the providers is used to determine the best CSP.

Qu et al. [18] proposed a CCCloud framework model, where service selection depends on subjective and objective assessment. Subjective assessments are obtained from end users and objective assessments are obtained from third parties about service providers' performance or capabilities. They conclude the trustworthy by comparing and aggregating both assessments of the overall performance of CSP services. The objective assessment is dynamically changed based on the similarity of the context and the weighted value assigned to the cloud services. It not only provides the credibility of a user i.e. legitimate or not, but it also prevents user collisions in cloud services. Jayapriya et al. [19] proposed the CorQosCloud framework architecture, where quality of service (QoS) being a significant factor for choosing the services in a cloud environment. Service recommendation is done by rating and ranking. In this article, along with active user, the correlated QoS ranking algorithm was also presented to get the exact feedback about the service to achieve accuracy of the service.

Kanwal et al. [20] proposed a Trust Evaluation Model (TEM) to calculate the trust scores. In cloud federation, CSPs are dynamically collaborating to share their Virtual Machine (VM) infrastructure facility, due to the demand of quality of service during in load balancing. In order to check the data privacy and security author propose the TEM, because lack of trust between CSPs. It establishes a trust relationship between CSPs among SLAs [21]. Liu et al. [22] proposed Trust-aware recommender systems to provide a reliable trust aware QoS cloud services to the end users in a cloud environment. It combines the clustering-based algorithm and trust-aware method to provide the active user with a more customized QoS forecasting and trustworthy cloud services are recommended. It also overcome the data sparsity problem, task similarity among a similar users may result in differing prediction outcomes [23].

Habib et al. [24] proposed Trust aided unified evaluation framework. In this article both trust and reputation are considered as they relate to CSPs. To measure the trustworthiness of the cloud services, leveraging the trust and reputation systems that relate to QoS cloud related parameters are identified. The primary goal of the research is to provide an estimation of the future behaviour of the service provider. Noor et al. [25]

Table 2.1: Summary of Trust framework model

S.No	Author	Framework	Observations
1.	Devi et al. 2020 [17]	Linear Programming (LP) model	In the detailed explanation of the SLA repository, it is not clear how the weight is assigned to the end user requirements. Because in a real cloud environment, the needs vary for each end user.
2.	Liu et al. 2019 [22]	Trust-aware recommender systems	Experiment setup has only 2 data sets to implement the approach to achieve a better performance. Identification of the task similarity and data credibility problem in a cloud environment is not clearly mentioned.
3.	Al-Faifi et al. 2018 [32] and Wang et al. 2019 [33]	MCDM algorithm	MCDM algorithm is used to find out the optimal services for end users using clustering techniques. Performance wise it provides optimal services to the end user but reputation and trust are not focused.
4.	Smithamol et al. 2018 [30]	Trust management mechanism (TMM)	To evaluate the trustworthiness of cloud services in a real-time cloud environment, such as a multi-cloud environment, performance must be monitored in a dynamic manner, i.e. the availability of cloud services vary in real time context.
5.	Somu et al. 2017 [15]	TrustCom – a novel trust assessment framework (RSHT)	TThe identification of the subset of trust measure parameters requires the least amount of time. The efficiency of trustworthy services is carried out at fixed intervals, and the performance of the framework is similar to existing feature selection techniques.
6.	Jayapriya et al. 2016 [19]	CorQosCloud framework	The accuracy of the framework model compares with a few other frameworks, but the accuracy is purely dependent on the availability of services. Time and location are also major components of the accuracy that is not described in detail.
7.	Qu et al. 2015 [18]	CCCloud framework	Subjective and objective assessments are not extracted from the real dataset prior to implementing the cloud service selection model; only 50 percent of it is extracted from the cloud services and the remaining 50 percent is partially generated. So the result cannot be accurate, and the performance is carried out without the assessment context.
8.	Noor et al. 2014 [25]	Generic analytical framework (TCSR)	Framework compares the trust management prototype based on the assessment criteria such as (security, privacy, personalization etc.). But the criteria are varied in a distributed cloud environment due to the availability of services.
9.	Kanwal et al. 2014 [20]	Trust Evaluation Model (TEM)	Qualities of Service (QoS) parameters are determined by security and privacy and it is extracted from SLAs. The trust score are exchanged between CSPs to enhance cloud federation.
10.	Zheng et al. 2013 [28]	QoS ranking prediction framework (KRCC)	Due to the availability of cloud services in different contexts, such as time and location, the accuracy of the ranking model should be enhanced. Cloud services are ranked based on past usage of consumers.

proposed d a trust management service in the cloud environment. Various trust management techniques are available, which are categorized into Policy, Recommendation, Repudiation and Prediction. Generic analytical frameworks, for each level a set of dimensions are identified to evaluate and analyse the trust. It also compares various trust management methods depending on the evaluation criteria.

Kumar et al. [26] talks about the cloud service selection in spite of the availability of the services across a wide range of cloud platforms. Many service providers have the same functionality, and therefore choosing an appropriate service provider is one of the most important tasks. A non-functional QoS requirement has been taken to the cloud service ranking in terms of Fuzzy logic. Noor et al. [27] addresses the credibility of the user feedback and also recommends to the end user a CSP that completely depends on trust. Zheng et al. [28] proposed a CloudRank QoS ranking prediction framework. The selection of cloud services are determined by a QoS ranking algorithm through a set of functionality. It is quite difficult to identify the cloud services in a real time cloud environment. The consumer experience is included in this framework. Depending on the needs or requirements, the prediction framework provides an optimal cloud services to the end user. The framework not only provides optimal services, it also reduces the time and efforts consumed process of choosing optimal services for the end user.

Li et al. [29] discusses in the same issue, but the proposed T-broker serves as an intermediary between the user and the cloud environment. Depending on the trusted attributes, it provides the trustworthy cloud services to the end user. It has a lightweight feedback mechanism to improve the efficiency or accuracy of the cloud services that are offered by the end user. Smithamol et al. [30] presented a trust management mechanism for multi cloud environments. The trust management framework supports a multi-cloud environment with distributed cloud service availability in order to determine the CSP's trustworthiness. The framework mainly focuses on accuracy and efficiency [31].

Al-Faifi et al. and Wang et al. [32, 33] addresses the selection of cloud services depends on the hybrid multi criteria decision method. Using clustering techniques, similar features that are offered by service providers are identified. Ranking of the service providers is obtained after applying the multi-criteria decision making (MCDM) algorithm. The following Table 2.1 provides a brief overview of the trust framework model present in the literature.

As per the above mentioned literature, the selection of cloud services in a multi cloud environment is decision making/ranking/recommendation based. Due to the availability of services in a cloud context, the trust value could be measured [34]. But, measuring the trust value is not a simple task because of lack of subjective and objective assessment. Identifying or extracting real-time features from CSPs and end users to measure trust is extremely difficult. Even identifying the services for end user requirements is a difficult task; the requirements vary for each individual. In cloud federation [35], an enormous pool of services is constructed. Many of the pool's services are of the same nature, but they are accessed in different ways and have distinctive features. Consumers' must identify and pick the desired service from the available options. Due to the same functionality of services that are given differently, it is difficult for end users to choose the best option. This existing research is only suitable for consumers who wish to identify cloud services that are identical to the method they already knew or engage, but does not take into account those who look for the optimal services without background knowledge [36, 37, 38].

In order to solve these issues and assist end users in their cloud service selection, a systematic recommended framework model is needed. The framework proposed incorporates model creation and Trustworthy Recommended Weighted value (TRWv). The model creation will be done by using the ML algorithm (Logistic Regression) and TRWv done by service provider capabilities and end user feedback ratings. TRF is an intelligent automated framework that can tackle the issue of identifying the best cloud services among numerous services available in the Internet. By making recommendations for necessary services, it can be utilized to effectively deliver customized services to end users. When a end user seeks a recommendation for a cloud service, the system identifies the most relevant services based on the cloud consumer's past interactions. The proposed trustworthy framework and methods provide conceptual recommendations for creating and developing relevant recommended systems for end users.

3. Proposed Recommendation Framework. A strong design framework makes available all feasibility to the end users, offering accurate services also a good elucidation to the trustworthiness. The framework integrates the capabilities of the service provider and extracts the consumer feedback, reviews, ratings and recommendations from past historical data. A framework itself has various interfaces such as service provider capability, consumer realistic experience and automated tools (using a machine learning approach), incorporating the whole interface which is connected to a single model.

Machine Learning (ML) is a technique that offers the ability to learn itself, discover hidden insights, train and enhance from experience without explicit programming [39]. A machine learning approach has been used in this article to build a TRF model, where more number of features and a huge dataset is included in this study. The ML technique helps in predicting the services for new features requirements which are given by an end-user, because the target features are of categorical type. The target output has 10 entities namely CSP1, CSP2 ... CSP10, based on the end user requirements the output may fall into any of the categories. The overall proposed trustworthy recommendation framework Figure 3.1 provides trusted services to end users as

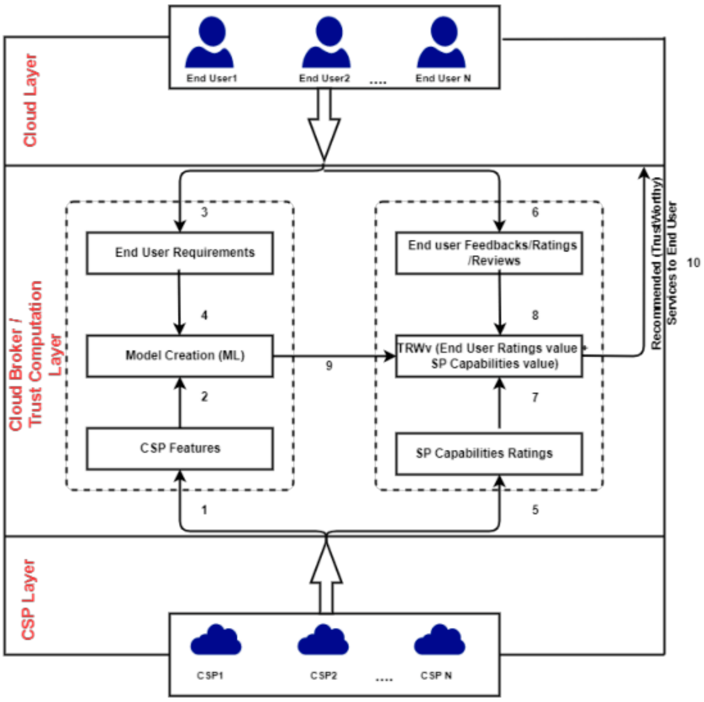

Fig. 3.1: Trustworthy Recommendation Framework

compared with other framework models.

Identifying CSP features or selecting features has been a difficult task. This article focuses only on the recommendation framework to provide optimal services to the end user. Machine learning model creation and trust value computation using Trustworthy Recommended Weighted value (TRWv) are the major parts of the recommendation framework. About 250 features have been identified to incorporate the ML model. It includes a range of services, such as computing, storage, networking, management, security and performance. These services are given by the CSPs. For the implementation of the TRWv approach, end-user ratings/reviews and service provider capabilities have been taken as trust measure parameters. It has a total of 76 features, 52 end-user rating features and 24 service provider features, such as SLAs (Service Level Agreement), Audits and Compliance, Measuring and Ratings, Security and Performance. The entire dataset or features are available in the GitHub repository [40].

The proposed framework has three layers such as 1.Cloud Layer 2. Cloud Broker/Trust Computation Layer 3. Cloud Service Provider Layer. Cloud Layer and CSP layer are interconnected with each other. In the cloud environment, all the detailed information about the end user is given by the cloud layer. To offer a trust service to the end user, the cloud layer is used to obtain the user requirements and service provider ratings and the information is provided to the trust computation layer. CSP layer provides a complete solution/idea about the service provider's availability on the Internet. To enable the services, all service providers have to register in the CSP registry. With the help of service provider (CSP-Registry) registry, the entire CSPs details are extracted from the Internet. The CSP layer was used to extract the service provider features and capabilities that are required for ML model creation and trust computation.

The Cloud broker/Trust computation layer acts as a middle layer in the proposed framework. Machine learning model creation and trust computation are done in this phase. The Trustworthy Recommended Weighted value (TRWv) is the primary component of the trust computation layer that determines whether the ML model returns the services which is trustworthy or not. The process is repeated until trusted services are obtained.

After identifying the features of each service, the ML model is built using a machine learning algorithm

(Logistic regression). The end user requirements are given to the ML model to obtain trustworthy services. After obtaining the end user requirements, the ML model will provide the services to the trust value computation approach TRWv. The TRWv combines SPVA and EVA. End-user ratings value average is referred to as EVA, while service provider capabilities value average is referred to as SPVA. The CSPs capabilities and end user feedback/reviews/ratings about each services provider available in the cloud layer and CSP registry are extracted, and a rating is given out of 10. The end user has to rate each service which are given by the service provider. Depending on the service provider's capabilities they can assign their ratings for their own services. The trust value was computed for the all the services with TRWv (Trustworthy Recommended Weighted value) by combining CSP capabilities and end user ratings. In order to measure the trust value, 70% of the weightage is given to the end user ratings and 30% weightage is given to the service provider capabilities. The trust value computation algorithm will check each service, whether the returned services trust threshold value is equal or greater than to 7. If the service value is greater than 7, the services are returned to end user through the cloud layer. If the service trust score is not satisfied by the threshold, the ML model accesses the next service. These services are known as recommended or trustworthy services. The results suggest that the trustworthy recommendation framework increases the accuracy of trust services and is more accurate in determining trustworthy services then other related frameworks that are discussed in literature section.

The trustworthy recommendation proposed framework contribution is outlined in the following steps:

1. Identify each service feature.
2. The model was created using a Logistic regression algorithm for each service along with service provider features.
3. Extract end user requirements from the cloud layer.
4. End user requirements have been given to the ML model.
5. Identify SP capabilities ratings in CSP layer.
6. Extract end users feedback/reviews/ratings.
7. SP capabilities values are given to TRWv.
8. End users values are given to TRWv. TRWv compute the trustworthiness of a service by integrating the value given by the CSP and user.
9. Depends on their end user requirements the model was given the services, which are returned to trust computation.
10. Identify whether the services are trustworthy or not using Trustworthy Recommended Weighted value (TRWv) approach/method that is given to end users. If the services are not trustworthy again these services are returned to ML model, until they get a trusted service.

The implementation of the trustworthy recommendation model is given as step by step pseudo code in Algorithm 1, which is represented as Trustworthy Recommended Weighted (TRW)_cloud broker. It begins with the creation of the model, progresses to the testing of the model, and concludes with the identification of trustworthy services for the end user.

The TRW_cloud broker algorithm has three parts namely,

1. Model creation
2. Testing the models
3. Identify the trustworthy services

It starts with reading a dataset from a service that is used to build a machine learning model. The dataset is represented in the form of a matrix with features (F) and rows (N), i.e., the number of samples. After identifying the independent (input) and dependent variables (output), need to perform exploratory data analysis (EDA) and data preprocessing to enhance the dataset. The better understanding of the data is necessary in order to remedy issues of missing values, spelling errors, and standardizing values such that they transform the data well enough for model creation. To apply the ML model, the data has to be divided into training and testing sets. The training set is used to build a predictive model, and the testing set is used to make a prediction. To implement a machine learning model, the LogisticRegression algorithm has been chosen because it is a classification algorithm used to allocate data to a discrete set of classes and is also suitable for multi-linear function classification problems. Problem formulation of LogisticRegression is given in equation (3.1) as

$$f(x_1, x_2) = b_0 + b_1x_1 + \; + b_nx_n \tag{3.1}$$

Algorithm 1: Trustworthy Recommended Weighted (TRW)_cloud broker

```
Model Creation
Input: file directory of dataset (filepath)
Output: model
foreach filepath in do
    dataset ← DataframeofCSV
    dataset ← NULLvaluesreplacedas0
    X ← dataset.iloc[:, : −1].values
    y ← dataset.iloc[:, −1].values
    X ← Xislabelencoded
    X_train,X_test,y_train,y_test ← train_test_split(X, y, test_size = 0.2, random_state = 0)
    classifier ← LogisticRegression(random_state = 0)
    classifier.fit(x_train,y_train)
    model ← y_pred ← classifier.predict(X_test)
end
Testing the Model
Input: file directory user input
Output: services
foreach filepath in do
    dataset ← DataframeofCSV
    dataset ← NULLvaluesreplacedas0
    initialize i to 0
    X ← dataset.iloc[:, : −1].values
    TRWv(services ← model[i].predict(X))
    accuracy_score(y_test,y_pred)
end
Determine the Trustworthy Services
Input: EVA, SPVA
Output: Trustworthy Recommended Weighted value (TRWv)
Determine End-user feedback/ratings value average (EVA)
Determine Cloud service providers capabilities value average (SPVA)
ENTRIES (X): ← Enduser and CSPs ratings for each service
EVA = X.mean()
SPVA = X.mean()
TRWv = EVA [ ] *70% + SPVA [ ]*30%
Input: Trustworthy Recommended Weighted value (TRWv) for each service
Output: to determine Trustworthy service
initialize i to 0
Def TRWv(predictedservices):
    if TRWv[services][i] >=7.0 then
    return services
    else
        model[i].predict(services)
end
```

where b_0, $b_1 \dots b_n$ are estimators or predicted weights and $x_1 \dots x_n$ are number of inputs. Predicted probability of logistic regression function in equation (3.2)

$$p(x_1, x_2) = 1/(1 + exp(-f(x_1, x_2))) \tag{3.2}$$

It's usually close to 0 or 1. The expected probability that the outcome for a given x being 1 is typically

Table 3.1: Dataset for Proposed Model

Cloud Service Provider (CSP)	Model Creation		Trustworthy Features	
	Services(S)	Features Count	Trustworthy Features	Features Count
CSP1 CSP2 CSP3 CSP4 CSP5 CSP6 CSP7 CSP8 CSP9 CSP10	Services1 Services2 Services3 Services4 Services5 Services6	174	CSP Capability /Performance End users Ratings	22 54
10CSP		174F		76F

interpreted as the function $p(x)$. As a result, 1-$p(x)$ represents the probability that the output will be zero. To obtain the optimal weights, Maximize the log-likelihood function (LLF) for all occurrences $i = 1 \ldots n$. Maximum likelihood estimation is the method, which is represented by the following equation (3.3).

$$LLF = \sum_i (ylog(p(x)) + (1-y)log(1-p(x))) \tag{3.3}$$

Logistic regression is used to determine the best predicted weights b_0, $b_1 ... b_n$, and the function $p(x)$ is as near as feasible to all real answers y_i , i = 1...n, where n is the number of occurrences. The LLF for the corresponding observation is equivalent to log(1- p(x_i)) when y_i=0. Whenever p(x_i) reaches y_i = 0, log(1- p(x_i)) becomes 0.

In our dataset, it has feature values of between 0 and 1. Although it has 60 services, 10 cloud service providers (CSP1, CSP2, CSP3 ... CSP10), each CSP offers 6 different services such as computing, storage,networking, management, security and performance which are denotes as Services1, Services2 and so on, 300 end users, and 250 features are taken to implement the problem. The entire dataset of the proposed framework model is shown in Tables 3.1. Even though it is a multi-linear function classification problem, each iteration acts as a single linear classification problem. Because with each iteration, it finds the most suitable service provider for the services based on the end user's requirements.

Once the model creation is successfully done, it will predict the result by checking the testing data. The model creation is done by 174 features, which includes the services features. Depending on the end user requirements, the TRWcloud broker algorithm predicts the outputs, which are shown in Figures 3.2 and 3.3. The end user has given the 174 features a value in the form of 0 or 1 while testing the model. The input samples are given in Figure 4.1 (ML Model Creation Features), which represents the end user input which varies from others in the value of 0 or 1. The predicted outputs are in the form of single array entities, which are shown in equation (3.4) for user Input1. It is plotted in the graph using a Python simulation, which is also presented in Table 3.3.

$$[CSP2, CSP5, CSP10, CSP4, CSP8, CSP1] \tag{3.4}$$

The proposed model output is interpreted as follows: The following CSPs are considered as trustworthy, recommended cloud services depending on the end user requirements in the form of features. These are the services that are returned by the TRW_cloud broker, which are represented in Figure 3.2 and Table 3.2 as, CSP2 provides Services1 (computing), CSP5 provides Services2 (storage), CSP10 provides Services3 (networking), CSP4 provides Services4 (management), CSP8 provides Services5 (security) and CSP1 provides Services6 (performance). Similarly, Figure 3.3 is predicted by the TRW_cloud broker algorithm for other user input2. The last part of the algorithm is to find out the trustworthiness of the services, which are returned by the service providers. Trustworthy Recommended Weighted value (TRWv) is used to find the trust value for each service. If the TRWv values for the services are greater than the threshold value, they are expected to be trusted as services. Otherwise, depending on the end user requirements, the ML model has to find the next

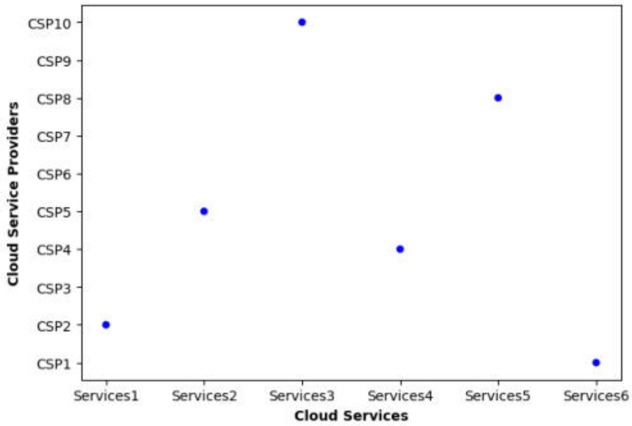

Fig. 3.2: Proposed Model Prediction for user Input1

Table 3.2: Predicted Output for user Input1

CSPs	Services1	Services2	Services3	Services4	Services5	Services6
CSP1						
CSP2						
CSP3						
CSP4						
CSP5						
CSP6						
CSP7						
CSP8						
CSP9						
CSP10						

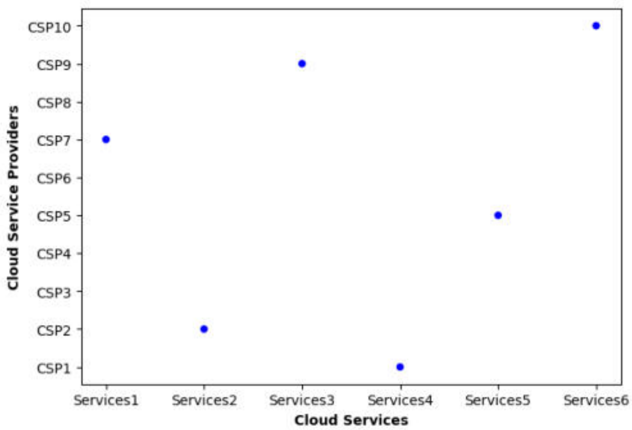

Fig. 3.3: Proposed Model Prediction for user Input2

cloud service until it gets a trusted service. The following sections elaborate on the features and accuracy of the results compared with other formworks that are discussed in the Section 2.

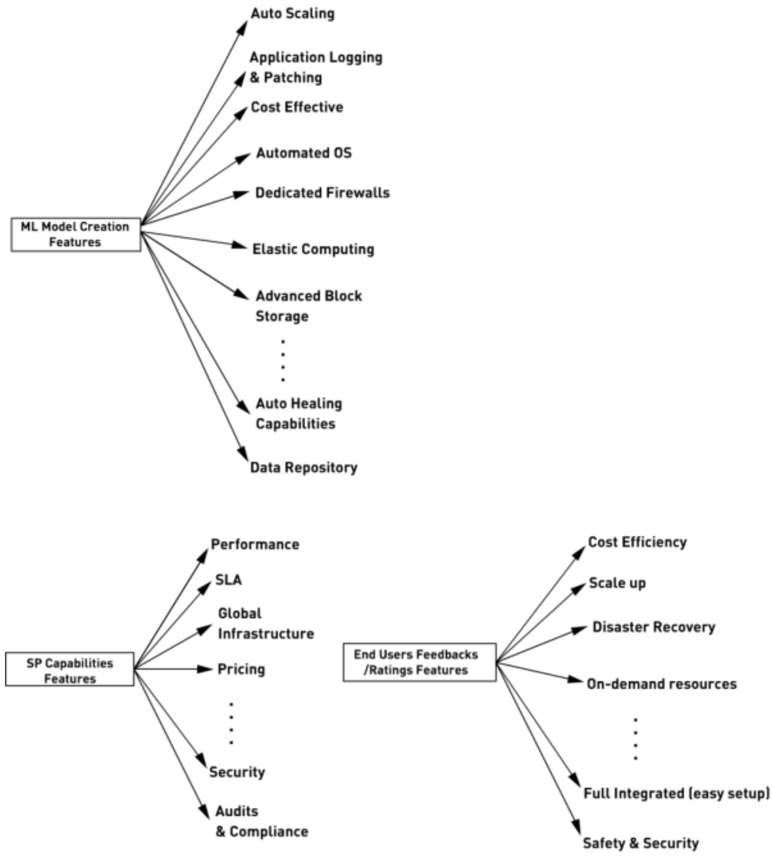

Fig. 4.1: Proposed framework features

4. Feature comparison with other frameworks. In the literature section, various frameworks proposed by the authors have already been discussed along with their advantages and observations. All the research articles that have shown performance/accuracy/efficiency are good compared to a few other frameworks. As a results of the availability of cloud services in a real multi-cloud environment with dynamic nature, context and location, time and trust measure parameters. However, in our proposed framework, identified trust measure parameters or features are evaluated by industry experts as well as a large number of features have been included during the research Figure 4.1. Table 4.2 presents a list of trustworthy features considered in our approach and excluded from previous research in the literature.

It includes ML model creation features, CSP capabilities features and end user feedback/reviews/ratings on parameters such as auto scaling, application logging patching, cost effective, automated OS, firewalls, elastic computing, advanced block storage and data repository. For SP capabilities performance, SLA, global infrastructure, pricing, security and audits and compliance etc. are identified. Similarly end user ratings have the following features such as cost efficiency, scale up, disaster recovery, on demand resources, full integrated and safety and security. All features are extracted in a real environment, and therefore the accuracy of the result is high compared to other frameworks. The framework integrated both the machine learning model and the Trustworthy Recommended Weighted value (TRWv) approach to provide trustworthy services to end-users. The following Table 4.1 highlights the features or trust parameters used in this article and compared them with other frameworks, to show how trustworthiness is improved in the proposed trustworthy recommendation framework.

Table 4.1: Comparison with other frameworks

Authors	Features	Techniques	Accuracy and efficiencey	Accuracy and No of User's involvement and Services	Weightage
Liu et al. 2019 [22]	response time	Trust-aware recommender systems	Mean Absolute Error	339 users 5825 web services	equal priority is given to CSP and users
Smithamol et al. 2018 [30]	response time, latency, and failure. throughput, CPU percentage, and network bandwidth.	Combining subjective and objective assessment	throughput and response time	142 users 4500 services	70% importance to objective trust, whereas 30% is subjective trust.
Somu et al. 2017 [15]	QoS attributes trust measure parameter (TMP) include 12 features such as availability, security , price etc.	Rough set: hypergraph-based trust measure parameter selection technique	To enhance the accuracy of the services, computation time play a vital role	10080 feedbacks that relates to QoS parameters. 7000 customers and 114 cloud services	Service selection techniques are verified in terms of size, ranking and time complexity.
Jayapriya et al. 2016 [19]	response time from web services and throughput from users	Correlated QoS Ranking Methodology	response time and availability of services Normalized Discounted Cumulative Gain (NDCG)	300 distributed users 500 web services	Priority is given to similar users.
Qu et al. 2015 [18]	3 subjective attributes such as privacy, response time etc. and 2 objective attributes response time and CPU performance	Creditability evaluation approach	Location and time	300 ordinary customers and 59 real cloud services	Priority is given to identifying the users' credibility to assess the cloud service truthfully
Kanwal et al. 2014 [20]	confidentiality, integrity, access control and authentication	Trust Evaluation Model (TEM)	Trust-score	2 web services 3 CSP (cloud node) 6 users 30 questionnaires	equal priority is given to CSP and users (Average of both scores has taken into the final trust value)
Zheng et al. 2013 [28]	response time, throughput, failure probability etc.	Kendall Rank Correlation Coefficient (KRCC)	QoS ranking prediction (performance)	300 distributed users 500 web services	Similar users are given weightage to rank services.
Proposed TRF	250 features to build a TRF (52 features from the client side and 24 features from the server side)	Machine learning model and TRWv approach	confusion matrix	300 users 60 services 10 CSPs	70 % priority to users and 30 % priority given to CSPs.

5. Results and Discussion. This section, elaborately discussed TRWv value and accuracy compared with other frameworks. To calculate or determine the TRWv, end-user ratings and service provider capabilities ratings are taken as an input. To measure EVA and SPVA, the end-user rating value average and service provider capabilities value average are considered. Using equations (5.1) and (5.2), each service's EVA and SPVA can be calculated. For example, CSP1 EVA and SPVA values for a service1 are 9.26 and 9.62 respectively. Using equation (5.3), the TRWv value for the CSP1 is computed as 9.368. Similarly, using the above equations, the CSPs for other services can be calculated.

Table 4.2: Trustworthy Features

Trustworthy Features/Factors		Our Methodology	Other work in the literature
Trustworthy Features (Service Provider Capability and CC reviews)	Retention Time, Audits and Compliance (HIPAA,PCI,ISO,CSA), Global Infrastructure, Data Center Zones and Location, Disaster recovery, Features for enterprise, BI analytics support,Innovation features, Increased efficiency and flexibility,Comprehensive and reliable, Networking setup, Indispensable speed, Performance above expectation, Cloud managed stocks and inventory, Security and compliance experts, Infrastructure support, hosting, Migration support, Cloud platform for audits, VPN hosting, Cost saving and scalability, Cloud ecosystem, Native integration with common OS and Forecasting	Yes	No

Table 5.1: TRWv value for each service

CSPs	Services1	Services2	Services3	Services4	Services5	Services6
CSP1	9.368	7.602	9.252	9.174	8.74	9.064
CSP2	9.064	9.416	9.76	9.176	8.46	9.064
CSP3	9.062	9.064	7.94	9.3	6.46	9.7
CSP4	8.902	8.322	9.7	9	8.21	9.52
CSP5	9.934	9.4	9.856	9.52	7.514	9.188
CSP6	9.23	9.096	9.398	9.188	8.862	9.062
CSP7	9.062	9.3	7.94	9.3	8.46	7.628
CSP8	6.828	8.342	8.608	8.714	7.628	9.332
CSP9	9.332	8.96	6.392	7.98	7.92	9.062
CSP10	9.368	8	8.622	9.2	8.826	6.94

The TRWv for a services are computed for all the CSP and given in Table 5.1, which represents the TRWv value or score for all the services that are available in the dataset (60 cloud services, 10 CSP). From Table 5.1, the proposed framework provides trusted services to the end user because the threshold values are greater than 7.0 for most services, except a few. The untrusted services are provided by CSP3 (services5), CSP8 (services1), CSP9 (services3), and CSP10 (services6), and the proposed framework repeats the process until the end user receives a trustworthy service.

$$EVA = (Sum\ of end\ user\ ratings)/(No\ of\ Features) \tag{5.1}$$

$$SPVA = (CSPs\ Capabilities\ ratings)/(No\ of\ features) \tag{5.2}$$

$$TRW_v\ score = EVA\ * 70\% + SPVA * 30\% \tag{5.3}$$

A confusion matrix (CM) is used to determine the accuracy of the suggested proposed model. The outcome of the CM is represented as Table 5.2. From the above results, true negative is 120, true positive is 148, false positive is 22 and false negative is 16. Using the below equations (5.4, 5.5, 5.6 and 5.7), accuracy and F1 score of the proposed model is computed.

$$Accuracy = (TP + TN)/N = 0.913 \tag{5.4}$$

$$F1score = 2 * (recall * precision)/(recall + precision) \tag{5.5}$$

Table 5.2: Confusion Matrix

	Predicted (False)	Predicted (True)	Total
Actual (False)	TN = 120	FP=22	142
Actual (True)	FN= 16	TP=148	158
	136	164	

Table 5.3: Accuracy comparison with other framework

S.No	Author	Framework	Accuracy
1.	Liu et al. 2019 [22]	Trust-aware recommender systems	0.908
2.	Smithamol et al. 2018 [30]	TMM	0.90
3.	Somu et al. 2017 [15]	RSHT	0.88
4.	Jayapriya et al. 2016 [19]	CorQoSCloudRank	0.90
5.	Qu et al. 2015 [18]	CCCloud	0.904
6.	Noor et al. 2014 [25]	TCSR	0.88
7.	Zheng et al. 2013 [28]	KRCC	0.894
8.	Proposed - Trustworthy Recommendation Framework(TRF)	LogisticRegression and TRWv approach	0.913

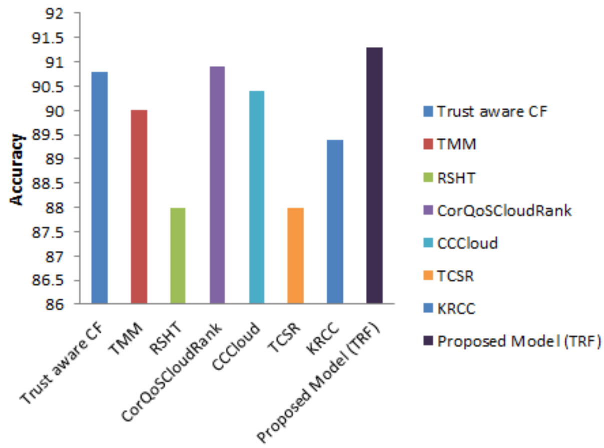

Fig. 5.1: Accuracy comparisons

$$Recall = TP/(TP + FN) = 0.923 \quad (5.6)$$

$$Precision = TP/(TP + FP) = 0.904 \quad (5.7)$$

$$F1\ score = 0.9134 \quad (5.8)$$

The proposed framework model's Accuracy and F1 score are respectively 91.3 and 91.34. The following Table 5.3 presents the accuracy comparison with other framework models and Figure 5.1 depicts it graphically.

The confusion matrix is used to evaluate the classification accuracy in order to validate the model. If the accuracy is 90% or more, there is a good confidence in the results that the model gives. Otherwise, parameter

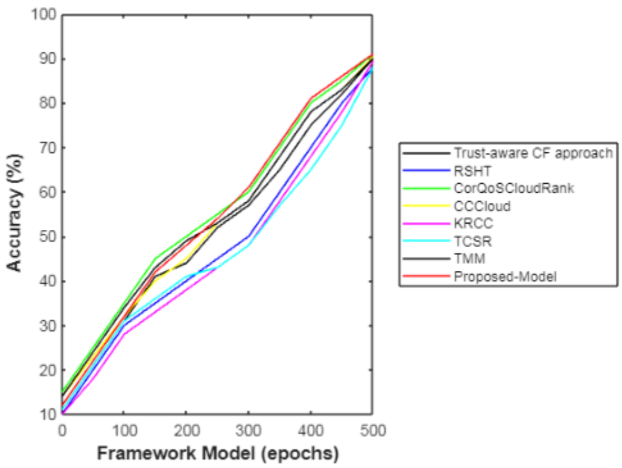

Fig. 5.2: Proposed framework Accuracy comparisons

tuning should be performed to increase accuracy of the results in the testing environment. The default iteration for the testing model is 100, but in our cases the maximum number of iterations taken. Because the accuracy of results automatically increases as the number of iterations or epochs increases, this is also depicted in Figure 5.2. After 450 training epochs, the proposed algorithm and the model's accuracy utilizing a variety of epochs start to converge. Depending on the end user requirements, the proposed model will provide an optimal, trustworthy service to the end user. If the model predicts the same output for more than one service, the priority is given to the first service, or based on the past history of services, the output service will be predicted. The entire approach has been implemented using LogisticRegression and TRWv approach. Since it is an automated ML model, features and end users can be increased at any time. The proposed trustworthy framework model has been implemented using Python 3.7 environment and the whole dataset has been submitted to the Github repository.

This section also compares the performance of the proposed methodology with other models described in the literature review section. the proposed framework model offers an optimal and reliable service to the end user, as the model has trained almost 250 features. The model not only provides the optimal services to the end users , it also provides trustworthy services, because 52 features are identified from end users and 24 features from service providers to enable the recommendation framework to provide trusted services. In addition, the proposed trustworthy recommendation framework model outperforms the other framework model Figure 5.2. Since the proposed framework is intended for real time-environment and provides the best and most reliable services to its end users, also the model would reduce estimation time for an expanding number of features and provide accurate cloud services to the end users, so financial constraints are irrelevant in this case. When compared to other frameworks discussed in the related work, the proposed trustworthy recommendation framework has a good performance or efficiency.

6. Conclusion and Future enhancement. It is a tedious task to determine a trustworthiness of cloud services depending on their CSPs capabilities along with end user requirements. The researchers also worked hard to develop a framework model for selection of cloud services as a significant solution for end users/cloud consumer. Cloud service provider can also have many cloud services with similar features. For end users, it is necessary to ensure their optimal services are trustworthy. Our proposed Trustworthy recommendation framework (TRF) model provides the optimal solution for the end user to identify the suitable trust services in a cloud environment. To check the trustworthiness of the cloud services, Trustworthy recommended weighted value (TRWv) approach is implemented to measure the TRWv. In order to assess the performance of TRF, accuracy and F1 score are calculated using confusion matrix. The best framework cannot be inferred, due to dynamic, distributed, and non-transparent nature of the service provider's availability in a federated cloud environment. Furthermore, the framework model can incorporate maximum no of CSPs features to enhance the dynamic behaviour of the cloud environment.

REFERENCES

[1] M. Marimuthu, J. Akilandeswari and P. R Chelliah, *Identification of trustworthy cloud services: solution approaches and research directions to build an automated cloud broker*, Computing., 104 (2022), pp. 43–72.
[2] B. Alouffi, M. Hasnain, A. Alharbi, W. Alosaimi, H. Alyami and M. Ayaz , *A Systematic Literature Review on Cloud Computing Security: Threats and Mitigation Strategies*, IEEE Access., 9 (2021), pp. 57792–57807.
[3] Somula. Rama Subbareddy and R. Sasikala,*A survey on mobile cloud computing: mobile computing+ cloud computing (MCC= MC+ CC)*, Scalable Computing: Practice and Experience., 19(4) (2018), pp. 309–337.
[4] R. Regaieg, M. Koubaa, Z. Ales and T. Aguili, *Multi-objective optimization for VM placement in homogeneous and heterogeneous cloud service provider data centers*, Computing., (2021), pp. 1–25.
[5] R. Hentschel and S. Strahringer, *A broker-based framework for the recommendation of cloud services: a research proposal*, In Conference on e-Business, e-Services and e-Society, Springer, Cham., (2020), pp. 409–415.
[6] M. Yang , V. S. Jacob and S. Raghunathan, *Cloud Service Model's Role in Provider and User Security Investment Incentives* , Production and Operations Management., 30(2) (2021), pp. 419–437.
[7] Somula.Ramasubbareddy, Anilkumar. Chunduru , Venkatesh. B ,Karrothu. Aravind , Kumar.Pavan and Sasikala. R,*Cloudlet services for healthcare applications in mobile cloud computing*, proceedings of the 2nd international conference on data engineering and communication technology.,(2019), pp. 535–543.
[8] A. Hussain, J. Chun and M. Khan, *A novel framework towards viable cloud service selection as a service (cssaas) under a fuzzy environment*, Future Generation Computer Systems., 104 (2020), pp. 74–91.
[9] R. Djiroun, M. A. Guessoum, K. Boukhalfa and E. Benkhelifa, *A novel cloud services recommendation system based on automatic learning techniques*,International conference on new trends in computing sciences (ICTCS) IEEE., (2017), pp. 42–49.
[10] H. Mezni and T. Abdeljaoued, *A cloud services recommendation system based on Fuzzy formal concept analysis*, Data & Knowledge Engineering., 116 (2018), pp. 100–123.
[11] C. Mao, R. Lin, C. Xu and Q. He, *Towards a trust prediction framework for cloud services based on PSO-driven neural network*, IEEE Access., 5 (2017), pp. 2187–2199.
[12] P. S Pawar, M. Rajarajan, T. Dimitrakos and A. Zisman, *Trust assessment using cloud broker*, In IFIP International Conference on Trust Management. Springer, Berlin, Heidelberg., (2014), pp. 237–244.
[13] M. Usha, J. Akilandeswari and A. S Fiaz, *An efficient qos framework for cloud brokerage services*, International Symposium on Cloud and Services Computing, IEEE., (2012), pp. 76–79.
[14] M. Tang, X. Dai, J. Liu and J. Chen, *Towards a trust evaluation middleware for cloud service selection*, Future Generation Computer Systems., 74 (2017), pp. 303–312.
[15] N. Somu, K. Kirthivasan and V. S. Sriram, *A rough set-based hypergraph trust measure parameter selection technique for cloud service selection*, The Journal of Supercomputing., 73(10) (2017), pp. 4535–4559.
[16] T. H. Noor and Q. Z. Sheng, *Web service-based trust management in cloud environments*, In Advanced Web Services. Springer., (2014), pp. 101–120.
[17] R. Devi and R. Shanmugalakshmi, *Cloud providers ranking and selection using quantitative and qualitative approach*, Computer Communications., 154 (2020), pp. 370–379.
[18] L. Qu, Y. Wang, M. A. Orgun, L. Liu., H. Liu and A. Bouguettaya, *CCCloud: Context-aware and credible cloud service selection based on subjective assessment and objective assessment*, IEEE Transactions on Services Computing., 8 (3) (2015), pp. 369–383.
[19] K. Jayapriya, N. A. B. Mary and R. S. Rajesh, *Cloud service recommendation based on a correlated QoS ranking prediction*, Journal of Network and Systems Management., 24(4) (2016), pp. 916–943.
[20] A. Kanwal, R. Masood and M. A. Shibli, *Evaluation and establishment of trust in cloud federation*, In Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication., (2014), pp. 1–8.
[21] Somula. Rama Subbareddy and R. Sasikala,*RTTSMCE: a response time aware task scheduling in multi-cloudlet environment*, International Journal of Computers and Applications., 43(7) (2021), pp. 691–696.
[22] J. Liu and Y. Chen, *A personalized clustering-based and reliable trust-aware QoS prediction approach for cloud service recommendation in cloud manufacturing*, Knowledge-Based Systems., 174 (2019), pp. 43–56.
[23] Ramasubbareddy. Somula , Swetha. Evakattu , Luhach. Ashish Kumar and Srinivas T. Aditya Sai,*A Multi-Objective Genetic Algorithm-Based Resource Scheduling in Mobile Cloud Computing*, International Journal of Cognitive Informatics and Natural Intelligence (IJCINI)., 15(3) (2021), pp. 58–73.
[24] S. M. Habib, S. Hauke, S. Ries and M. Muhlhauser, *Trust as a facilitator in cloud computing: a survey. Journal of Cloud Computing: Advances*, Systems and Applications., 1(1) (2012), pp. 1–18.
[25] T. H. Noor, Q. Z. Sheng, S. Zeadally and J. Yu, *Trust management of services in cloud environments: Obstacles and solutions*, ACM Computing Surveys., 46(1) (2013), pp. 1–30.
[26] R. R. Kumar, S. Mishra and C. Kumar, *Prioritizing the solution of cloud service selection using integrated MCDM methods under Fuzzy environment*, The Journal of Supercomputing., 73(11) (2017), pp. 4652–4682.
[27] 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*, ACM International Conference on Information Knowledge Management., (2013), pp. 2509–2512.
[28] Z. Zheng, X. Wu, Y. Zhang, M. R. Lyu and J. Wang, *QoS ranking prediction for cloud services*, IEEE transactions on

parallel and distributed systems., 24(6) (2012), pp. 1213–1222.
[29] X. Li, H. Ma, F. Zhou and W. Yao, *T-broker: A trust-aware service brokering scheme for multiple cloud collaborative services*, IEEE Transactions on Information Forensics and Security., 10(7) (2015), pp. 1402–1415.
[30] M. B. Smithamol and S. Rajeswari, *TMM: Trust Management Middleware for Cloud Service Selection by Prioritization*, Journal of Network Systems Management., 27(1) (2019), pp. 66–92.
[31] Somula. Rama Subbareddy and R. Sasikala,*A load and distance aware cloudlet selection strategy in multi-cloudlet environment*, International Journal of Grid and High Performance Computing (IJGHPC)., 11(2) (2019), pp. 85–102.
[32] A. Al-Faifi, B. Song, M. Hassan, A. Alamri and A. Gumaei, *A hybrid multi criteria decision method for cloud service selection from Smart data*, Future Generation Computer Systems., 93 (2019), pp. 43–57.
[33] Y. Wang, J. Wen, W. Zhou, B. Tao, Q. Wu and Z. Tao, *A cloud service selection method based on trust and user preference clustering*, IEEE Access., 7 (2019), pp. 110279–110292.
[34] F. Nawaz, M. R. Asadabadi, N. K. Janjua, O. K. Hussain, E. Chang and M. Saberi , *An MCDM method for cloud service selection using a Markov chain and the best-worst method*, Knowledge-Based Systems., 159 (2018), pp. 120–131.
[35] S. Chauhan, E. S. Pilli and R. C. Joshi, *BSS: a brokering model for service selection using integrated weighting approach in cloud environment*, Journal of Cloud Computing., 10(1) (2021), pp. 1–14.
[36] Somula. Rama Subbareddy and R. Sasikala,*Round robin with load degree: an algorithm for optimal cloudlet discovery in mobile cloud computing*, Scalable Computing: Practice and Experience., 19(1) (2018), pp. 39–52.
[37] Somula. Rama Subbareddy and R. Sasikala,*A honey bee inspired cloudlet selection for resource allocation*, smart intelligent computing and applications., (2019), pp. 335–343.
[38] Ramasubbareddy. Somula, Ramasamy. Sasikala , Sahoo. Kshira Sagar , Kumar R. Lakshmana , Pham. Quoc-Viet and Dao. Nhu-Ngoc,*Cavms: application-aware cloudlet adaption and vm selection framework for multicloudlet environment*, IEEE Systems Journal., 15(4) (2020), pp. 5098–5106.
[39] A. B. Saxena, D. Sharma and D. Aggarwal, *Trust Prediction Tree: Use of Decision Tree in Cloud Scenarios*, International Journal of Mechanical Engineering., 7(5) (2022), pp. 271–278.
[40] Dataset uploaded, https://github.com, [Accessed: 01-July-2022]
[41] Cloud broker, *https://en.wikipedia.org/wiki/Cloud-broker*, [Accessed: 01-July-2022]

Edited by: Vinoth Kumar
Received: Jul 19, 2022
Accepted: Nov 1, 2022