Received February 12, 2020, accepted February 24, 2020, date of publication March 9, 2020, date of current version March 26, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2979253

Benchmark Dataset Selection of Web Services Technologies: A Factor Analysis

MUHAMMAD HASNAIN[1], MUHAMMAD FERMI PASHA[1], (Member, IEEE), IMRAN GHANI[2], BILAL MEHBOOB[1], MUHAMMAD IMRAN[3], AND AITIZAZ ALI[1]

[1]School of IT, Malaysia, Monash University, Bandar Sunway 47500, Malaysia
[2]Department of Mathematics and Computer Sciences, Indiana University of Pennsylvania, Indiana, PA 15705, USA
[3]Next Bridge (Pvt.) Ltd., Lahore 54000, Pakistan

Corresponding author: Muhammad Hasnain (muhammad.malik1@monash.edu)

ABSTRACT Web services have emerged as an accessible technology with the standard 'Extensible Mark Up' (XML) language, which is known as 'Web Services Description Language' WSDL. Web services have become a promising technology to promote the interrelationship between service providers and users. Web services users' trust is measured by quality metrics. Web service quality metrics vary in many benchmark datasets used in the existing studies. The selection of a benchmark dataset is problematic to classify and retest web services. This paper proposes a method to rank web services quality metrics for the selection of benchmark web services datasets. To measure the diversity in quality metrics, factor analysis with Varimax rotation and scree plot is a well-established method. We use factor analysis to determine percentage variance among principal factors of four benchmark datasets. Our results showed that the two-factor solution explained 94.501, 76.524, and 45.009% variances in datasets A, B, and D, respectively. A three-factor solution explained 85.085% variance in dataset C. Reliability, and response time quality metrics were predicted as the most dominating quality metrics that contributed to explain the percentage variance in four datasets. Our proposed web metric ranking (WMR) method resulted in reliability as the top-most web metric with (57.62%) score and latency web metric at the bottom-most with (3.60%) score. The proposed WMR method showed a high (96.17%) ranking precision. Obtained results verified that factor solutions after reducing the dimensions could be generalized and used in the quality improvement of web services. In future works, the authors plan to focus on a dataset with dominating quality metrics to perform regression testing of web services.

INDEX TERMS Factor analysis, quality metrics, rotated loading, reliability, response time, regression testing, web services.

I. INTRODUCTION

Web services selection and ranking are key research areas to improve the performance of web services. Quality of web services is a critical criterion used for the selection or ranking of web services. Studies [1]–[3] used nonfunctional aspects of web services to classify web services in categories. Variance in the number of nonfunctional aspects has been observed in several studies. Kuang *et al.* [4] used response time and throughput quality metrics (nonfunctional aspects) for the selection of users' reputed web services. Serrai *et al.* [5] used response time, throughput, reliability, and best practices as nonfunctional aspects (quality metrics) for the selection of accurate web services. The purpose of using quality metrics in [4], [5] was to select the most trustworthy web services. An essential step in web service selection is the utility of the aggregate function of the 'quality of services' QoS metrics. Web services users have to weigh the quality features of web services before they accept the selection or ranking resulting from traditional approaches [6]. We have used features, nonfunctional aspects, and quality metrics interchangeably in this paper.

Web services profiling is a crucial research topic, and various challenges such as selection and ranking have become dependent on it [7]. The dependency of web services' ranking on QoS profiling is useful in proposing the approaches by using various quality metrics. Self-rating by users, or weighing nonfunctional aspects may impact the accuracy of web

The associate editor coordinating the review of this manuscript and approving it for publication was Xiao-Sheng Si.

services' ranking. Web services users do not have sufficient knowledge to consider the metrics. Alternatively, the web-server is also a source of QoS values [8], but providers' consent is mandatory before obtaining the metrics values. Similar to recent works [9], [10], we also use the QoS values collected at the client-side. Many of the primary studies used quality metrics to forecast the quality of web services. Evaluation of the proposed approaches in these studies was carried out by using primarily four types of datasets, which are explained below in section 4. The datasets used in this study have been employed as a benchmark datasets in a large number of primary studies [7]–[10]. These prime studies have a common limitation since none of the studies investigated the comparison of datasets in the context of dimension reduction of quality metrics. Web services quality metrics occur in multiple dimensions [11]. Wang *et al.* [12] mentioned that dimension reduction of web services metrics needs to be undertaken carefully.

The main contributions of this paper can be summarized as follows:

1) The first contribution of this paper is to investigate the quality metrics of web services in four benchmark datasets and analyze them to see how the percentage variance is impacted by the increased or decreased number of quality metrics.
2) The second contribution of this paper is to predict web services quality metrics that dominate in four benchmark datasets. Correct prediction of web services quality metrics helps researchers to execute the classification of web services occurrences. Accurate prediction of web services occurrences will lead authors to use that dataset for regression testing in their future works.
3) The third contribution of this paper is to propose an efficient quality metrics ranking method of web services to predict the most essential web services quality metrics.

The rest of the paper is structured as follows: Section 2 presents the motivation for our paper. Section 3 presents a literature review to place our work in the most recent research works on web services datasets. Section 4 gives an overview of the background of four datasets. Section 5 describes our proposed method in this study. Section 6 presents evaluation and results regarding, fuzzy rules implementation, web metrics individual scoring, and ranking. Section 7 presents the discussion in the context of results to predict the most dominating quality metrics after reducing the dimension of four datasets. Section 8 presents the sensitivity analysis of the proposed approach. Section 9 concludes this study and mentions the future research implications.

II. MOTIVATION

Web services selection that meets the users' requirements is a challenging task. Most of the existing web service selection approaches use QoS metrics. The selection of reliable web services depends on a detailed assessment of the quality metric datasets [30]. Selection based on the analysis of quality metrics is a quick way to evaluate the performance of a web service. However, this kind of selection does not appeal to service providers to bring improvement in their web services. For instance, users access the web service to view web pages. For those users, response time should be low to keep them satisfied. On the other hand, a user downloading the files requires a high throughput value. Both users have their specific requirements. Each quality metric has its criteria for measuring users' needs. Web services users' requirements and available facilities determine the composition of a benchmark dataset. Every benchmark dataset used for the evaluation of proposed web services approaches is composed of different quality metrics.

Since a web service provider agrees with web service users following a service level agreement (SLA) document, most of services providers claim that they meet 100% SLA for response time metric. However, web services do not show the quality metrics statistics given in the SLA document. To solve this mismatching issue is not an easy task for researchers, as web services technology uses a different approach for accessing the programming code. To guarantee that a web service is meeting users' requirements with only a few quality metrics is not sufficient. Subsequently, web services' quality is not always known by web services providers, and they need a comprehensive analysis of a set of quality metrics. Therefore, an appropriate web services dataset can be used to accomplish the task.

The selection of an appropriate benchmark dataset can lead to the improvement of the quality of web services. Quality metrics in a benchmark dataset can be used for the classification of web services. A web service with the quality metrics and their values provide potential opportunities for researchers in solving the classification problem of trusted or untrusted web services. Our work is motivated by the strength of classification approaches, which can be applied to a selection of web services. By choosing classified web services, we can decide which web service is retested on a prioritized basis than the other web services. To address the benchmark dataset selection, we consider the proposal of a conceptual model where a user invokes a web service by sending the HTTP request to a server (Figure 1). The web services server responds to a user for a sent HTTP request. A user learns that how a web service's server behaves against the demand in various measuring metrics. A quality measuring metric can be a single metric or a combination of different quality metrics. A copy of the HTTP response is sent to a data center component, which stores the invocation record of each user's request to a web service server. Multiple users can access one or more than one web service. In a data center, different datasets are generated in perspective of the number of quality metrics. Precisely investigated datasets in the context of quality metrics lead to the selection of a dataset that is used for regression testing of web services. Regression testing at any level (unit testing, component testing, and system testing) brings improvement in web services quality.

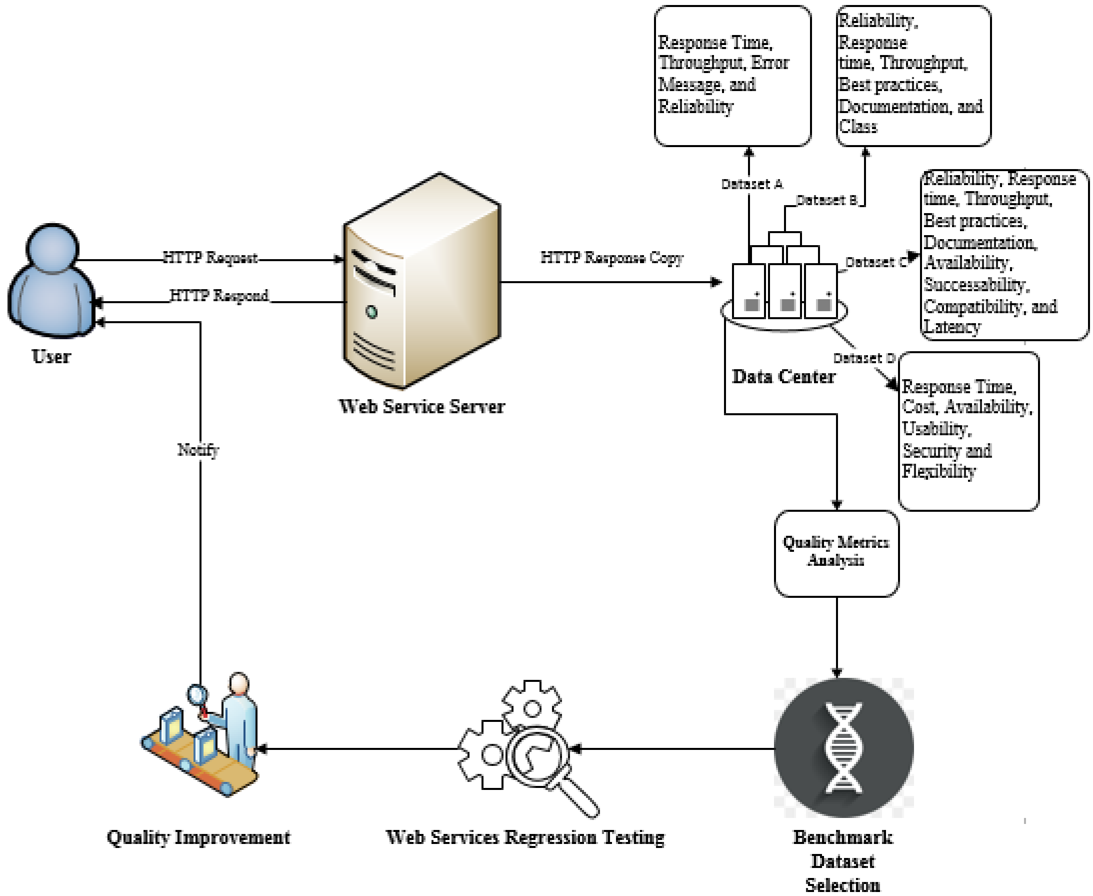


FIGURE 1. A conceptual model of benchmark dataset selection.

III. RELATED WORK

This section is focused on the literature review regarding web services metrics and factor analysis.

In [13], the SERVQUAL model was used to classify twenty-five items of web services information into five dimensions. A SERVQUAL model is aimed at capturing the web services users' expectations and perceptions regarding five aspects. Reliability, assurance, empathy, tangibles, and responsiveness were five dimensions. The authors wanted to verify five SERVQUAL dimensions of web services. Modification in SERVQUAL was the critical finding as the SERVQUAL model served the base framework to assess the quality of web services. In a subsequent research study [14], three dimensions of quality metrics were identified that did not coincide with the five aspects of the SERVQUAL model. In [15], the authors proposed a model to analyze 30 items with the varimax loading and retained 25 items with seven component factors. In a recent work [16], factor loading of a few web services quality metrics was investigated. The authors found that success-ability, best practices, and throughput quality metrics had higher loading than the rest of quality metrics. They suggested that the quality-metrics mentioned above were useful for the selection of web services.

To study the various factors affecting the video quality of web services over the wired network, factor analysis [17] was used to estimate the 'mean opinion score' (MOS), which is a subjective quality. They tested the correlation between variables collected from services and service users by reducing the dimension of factors. The 'linear structural relation 'LISREL framework [18] was used to determine the satisfaction hierarchy of web services with six dimensions. Each dimension was further linked to three variables. For example, reliability as a dimension of web service satisfaction was also comprised of accurate, credible, and trustworthy variables.

In [19], the author reported quality interface metrics such as reusability, modularity, testability, and maintainability. A set of rules and metrics was used to parse the web services interface definitions. Every metric, along with its rules, measures the interface quality. The proposed approach WSAudit, showed limitations to validate the assumption of rules and metrics. In another study, Baski and Misra called that best practice was the main quality metric to implement web services [20]. Best practices in the implementation of web services improved the maintainability of web services. The web services maintainability was investigated about four proposed metrics. Proposed four metrics were 'data weight' (DW), 'distinct message count' (DMC), 'message entropy' (ME) and 'message repetition scale' MRS metrics [20]. All of these four metrics were aimed to predict the better maintainability of web services. These metrics could have a relationship with other features of web services, which are limitations of the proposed approach.

Web services smells and anti-patterns were studied with regards to service component architecture (SCA) [21]. Anti-patterns were found to be critical factors for improving the design quality of web services. To specify anti-patterns, twenty-seven metrics were distributed into static and dynamic features of web services.

Web service selection is a vital aspect of 'service-oriented computing' (SOC). Due to an increase in the number of web services which provide similar functions has resulted in services selection problem for users. To overcome this issue, Oskooei and Daud [55] proposed a model of web services selection by using QoS attributes with low complexity. Further, modifications and changes in web services increase users' complexity to perform relevant operations by the unnecessarily complex interfaces. The latter study is aimed at fixing the design anti-patterns of web services [56].

To predict the maintainability of web services [22], the authors employed traditional software metrics. In a total, five web services interface metrics and eleven source code related metrics were focused in the proposed approach. Four code level metrics were removed, which did not have a relationship with the correlation of more than 0.6. The rationale behind eliminating the four code metrics was to keep highly correlated parameters together. About the proposed approach in [20], Coscia *et al.* [22] found that modification in web services implementation resulted in an increment in several object-oriented class-level metrics. As a result of the increment, DMC, ME, and MRS metrics increased the complexity in the WSDL document. An increment in ME and DMC is undesirable, that could be the side-effect of the 'Distinct Message Ration' DMR metric.

QoS prediction by using a large dataset is another issue in the web services selection and recommendation field. Thinh [57] proposed a 'two-layer model' (TLM) to evaluate the prediction of quality web services. The evaluation of the proposed TLM was performed on the dataset [28], [29]. The main limitation of the study is that it does not mention the applications of the proposed TLM on datasets with different parameters. To bridge this gap, the proposed work in this study is evaluated on four datasets with varying numbers of quality parameters.

Li and Jin [58] studied the problem of the accuracy of QoS prediction. They stated that an appropriate sampling method could improve the prediction accuracy in QoS datasets. This study proposes an effective sampling method for a dataset with two quality parameters. However, the study does not show the evaluation of the proposed method when quality metrics change in size in a respective dataset.

QoS ranking prediction approach, namely CloudRank [59], considers cloud services. Ranking is performed by comparing the pairs of services and their preferences. Experiment results indicated that the proposed CloudRank outperformed the existing QoS ranking approaches. However, high dimensional data with correlated parameters have never been ranked. Since various quality factors correlate with each other and exist in the collected datasets, no approach presents the QoS ranking, which is comprehensive in determining the parameters' classification.

Factor analysis is a statistical method used to group related variables for the same factors which are used for an assumption that a model exists [23]. In other terms, it is known as a method of dimension reduction that reduces a large number of variables into a few factors. The problem of factor analysis is aimed at deducing the number of web metrics and their composition from observations of web services datasets. The factor analysis model can be defined in the following Eq. (1).

$$X - \mu = LF + \varepsilon \tag{1}$$

where X is the representation of the observed variable vector, which has a mean vector of μ, L represents the factor loading matrix, and ε represents error. According to Baumann *et al.* [24], FA is used for reducing the dimension or number of features and then select the principal elements that explain the maximum variation among the original datasets. In other words, FA verifies the linear or non-linear nature of datasets. It is proven that the principal factors should cover at least 80% total variance. But non-linear nature of datasets has less percentage variance, as Deng *et al.* [25] stated that principal factors revealed 25% of total variance that verifies the non-linear nature of a dataset. Liu *et al.* [26], discussed the utility function regarding the evaluation of multi-dimensional composite web services. Quality factors were mapped to a single value to rank the web services. Ma *et al.* [27] conducted an exploratory study to find the dimensions of web services. They found a reduction in 8 to 10 aspects in the same study.

Factor analysis is used to find the correlation coefficient matrix of variables. A small number of variables control and describe the relationship between various variables. Factor analysis is conducted on four datasets independently. Only factors with the eigenvalue 1 or above are extracted. After the identification of several elements or dimensions, the next step is to investigate which item(s) fall within the corresponding dimensions.

IV. BACKGROUND OF WEB SERVICES QoS DATASETS

This descriptive study used four data sets with a varying number of web services quality metrics. We present the background of each dataset in the following.

A. DATASET A

The dataset A is accessible from WS-Dream repository of 'The Chinese University of Hong Kong.' Thus it can be called as WSDream Dataset. However, we call it dataset A in this study. The dataset A [28], [29], contains two potential quality metrics such as response time, and throughput. Each of the

TABLE 1. Datasets features.

Dataset Name	Meta Information	Potential Quality Metrics	Geographical Information	Purpose of using datasets
Dataset A	WSDL	Response time, throughput	Given	Quality prediction of web services
Dataset B	WSDL	Response time, throughput,	Not-Given	Trustworthiness of cloud web services
Dataset C	WSDL	response time, availability, and latency	Not-Given	Scoring, and ranking of web services
Dataset D	Not-Given	Response time, cost, availability	Not-Given	Ranking, and Selection of cloud web services

metrics has a 1974675 invocation records of 5258 web services. Three hundred thirty-nine users distributed in 74 countries have collected throughput and response time records. We generated qualifier values for reliability and the number of error messages metrics in the dataset. The value generation of a qualifier is aimed to make our primary concept clearer to provide the specific meaning. Since we deal with the dataset of web services quality metrics that originally contained only two quality metrics. Therefore, we require to increase the number of web metrics to conduct factor analysis of dataset A. Consequently, we used a total of 4 quality metrics.

B. DATASET B

Lue and Yuan have reported dataset B in [30], with six quality metrics, including reliability, response time, throughput, best practices, documentation, and class. However, the class metric was used as a qualifier value of five web metrics. Class as a web metric determines the popularity of a web service. A web service with more number of stars is popular than the web service with less number of stars. Dataset B contains the average value for each of the six metrics taken from seven weather web services. Objective entropy weight technique was applied to calculate the objective weight of web services.

C. DATASET C

The dataset C [31], [32], contains nine quality metrics such as reliability, response time, throughput, best practices, documentation, availability, success-ability, compatibility, and latency of 8 web services. This dataset is known as the 'quality of web services'(QWS) dataset. Dataset C was obtained after using several filtering techniques. Inaccurate web services were taken out of the final list of web services. Only web services with valid WSDL were kept in dataset C. Each web service contains an average value of all nine quality metrics. Detailed information of dataset C is accessible from (http://www.uoguelph.ca).

D. DATASET D

The dataset D [33] is a simulated dataset of web services that contains a total of six quality metrics. Dataset D is a mixture of qualitative and quantitative quality metrics. These quality metrics are response time, cost, availability, usability, security, and flexibility. Dataset D consists of quality metrics of 63 web services that have been synthetically generated. Due to the unavailability of the cloud web services dataset, they decided to use six attributes from six 'service measurement index' (SMI) categories. SMI is a standardized method to synthesize data for measuring and comparing cloud web services.

From the above-given description of four datasets, we conclude a few potential metrics of each dataset in Table 1. Potential web metrics can be used in the classification of web services. A potential web metric may have more chances of selection as it is collected at a user's end. Therefore, we consider response time, throughput, availability, latency, and cost as potential web metrics, since web services users have directly measured them. Table 1 presents a summary of our used four benchmark datasets. Potential quality metrics have been listed in Table 1 regarding their possibility of selection in our future work on web services classification and regression testing. Dataset A and dataset B have been investigated in terms of the quality and trustworthiness of web services technology, respectively. Dataset C and dataset D have been mainly used for the ranking of web services. The main difference between the use of dataset C and dataset D is that the former dataset has been used for ranking of web services, and the latter dataset is specifically used for ranking of cloud services. Web services QoS prediction, web services selection, and ranking have been addressed in several studies. We need to use them for classification and regression testing of web services. Therefore, the choice of an appropriate dataset for regression testing of web service is a challenging issue. Reducing or increasing the number of quality metrics can influence the efficiency of the proposed approaches. Therefore, it is desirable to focus on the selection of a very appropriate dataset that has exact and consistent information on quality metrics.

We have observed that response time and throughput quality metrics have been used in web services selection, ranking, and trustworthiness. Both response time and throughput have been called potential metrics in dataset A, and dataset B. Response Time along with other quality metrics, is mentioned as a quality metric in the remaining two datasets. Dataset meta information in the form of WSDL is available for datasets A, B, and C, while it is not mentioned for the dataset D. It is because the authors [33] generated dataset D for the evaluation of cloud web services. Another difference between the

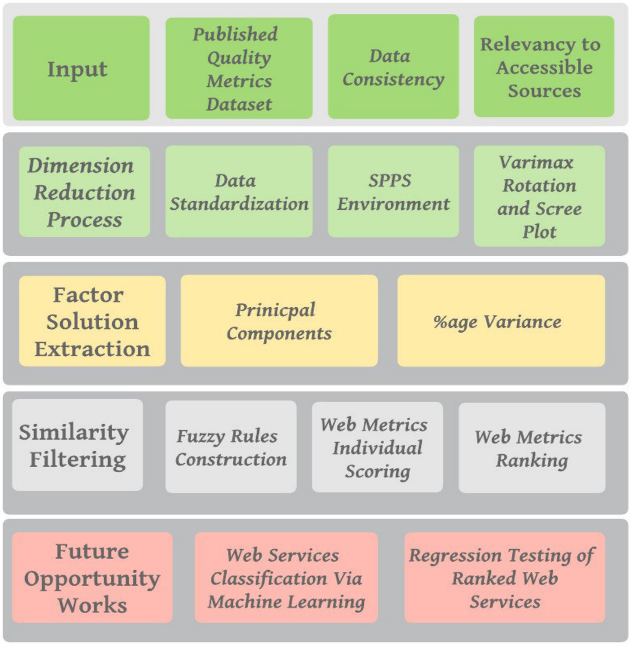


FIGURE 2. Proposed schema of dataset selection.

four datasets is that only dataset A mentioned the geographic location of web services and users.

V. PROPOSED METHODOLOGY FOR BENCHMARK DATASET SELECTION: FACTOR ANALYSIS

The proposed methodology for the selection of the benchmark web services dataset consists of five main modules. The first module is focused on web services datasets; the second one module is focused on the dimension reduction process (DRP). The third one module extracts factor solutions from datasets of quality web metrics. The fourth module is focused on performing the similarity filtering of factors to identify common and dominating web metrics from datasets. The fifth module presents the opportunities for future works regarding web services classification and regression testing. We show the proposed schema with the five necessary modules of this study in Figure 2.

We present a discussion on all five modules of the proposed approach for the benchmark dataset selection.

A. INPUT

The first module of the proposed model reveals the web services datasets as input to our proposed method. Web services data come from relevant sources, which include websites and published works. Quality metrics values of each web service dataset are tested for consistency and accuracy. Data consistency means that a web metric value is correct concerning its unit. For example, response time data is consistent with the unit of seconds. Data consistency and accuracy are the essence for better results. We check the relevancy of the accessible sources and published works to our proposed study. For this, we have revealed the research objectives for using the four datasets in Table 1. Our proposed work is aimed at selecting the most appropriate dataset by using some statistical process. Therefore, factor analysis is one of these processes which can be used to compare our chosen four datasets.

B. DIMENSION REDUCTION PROCESS (DRP)

We use factor analysis as a DRP in our proposed work. Before our proposed work, factor analysis was used in [34] to remove the redundant variables for improving the efficiency of related variables. For instance, Ayyildiz and Koçyigit [35] and Bianchini *et al.* [36] proposed to use non-functional aspects for the selection of web services. They identified metrics that were firmly related to the selection of web services. Tang *et al.* [37], in their proposed approach, determined the trustworthiness of cloud web services. Qi *et al.* [38] proposed a weighted 'principal component analysis' (PCA) method to reduce QoS metrics in the evaluation phase of their study. Using the factor analysis results, we interpret principal components, percentage variance with eigenvalues, and rotated component matrix values. In line with the studies mentioned above, we propose to use factor analysis for removing the redundant web metrics in four datasets. In our proposed work, DRP is used with three steps; the first step ensures that a dataset for factor analysis is standardized. The second step sets the SPSS environment to perform factor analysis, and the third step sets the varimax rotation as a critical loading method as detailed in the following.

1) DATA STANDARDIZATION

Data standardization is the critical step of the second module that allows us to bring data in a standard format. Quality metrics data of web services is measured in various types of units. In other words, standardized data is suitable for regular assessment. For example, dataset A contains a record of a large number of web services, users, and geographic location information of users and web services. Moreover, sometimes, data is missing, which is a big challenge for bringing consistent results. Data standardization techniques such as z-value and mini-max are widely used in research works. Both of these techniques result in standardized scores or data values with a common standard. Mathematical representation of latter technique min-max normalization is given in the following Eq. (2).

$$Zi = \frac{xi{-}min(x)}{max(x) - min(x)} \tag{2}$$

where Zi is representing the normalized value of a record; max(x) and min(x) are maximum and minimum values of quality metric records, respectively.

2) SPSS ENVIRONMENT

A big challenge for a researcher is to select an appropriate tool for statistical analysis. For example, factor analysis is supported by different instruments, including LISREL, SPSS, R, MATLAB, and AMOS, etc. Both R and MATLAB offer a separate menu of factor analysis. IBM SPSS 23.0 is chosen for the factor analysis of web services datasets. SPSS is user-friendly and appropriate for conducting factor analysis of datasets.

3) VARIMAX ROTATION AND SCREE PLOT

The varimax rotation [39] is the change of coordinates which maximize the sum of variances for the squared coefficients after transformation. The sparse representation of data that varimax seeks is met by a small set of features of individual samples. A particular sample contains zero values and non-zero features in the transformed representation that explain the data. Varimax rotation method has been widely used for the rotation of items to have a better interpretability of factors [40]. The use of varimax rotation is based on the assumption of uncorrelated factors. A varimax rotation method results in the false interpretation of highly correlated factors. The direct oblimin rotation has more excellent effects rather than using the varimax rotation to overcome the false interpretation issue [41].

Kaiser criterion is the most popular method used for correct estimation of a number of factors that represent the feature correlation and data variance. This criterion recommends retaining the factors which have eigenvalues greater than 1, as explained in [42]. The varimax rotation perturbs the principal components. As a result, the variance is maximized within each vector. Also, the number of variables with the intermediate loading is decreased for each vector. Therefore, the number of very small or substantial loading is increased [43]. Subsequently, varimax rotation streamlines the principal components as they are significantly dependent on a small number of original variables.

Each eigenvalue on (Y-axis) is plotted against the associated (X-axis) value. To represent the values of the two-axis, a graph known as the scree plot is used to describe the values of the two-axis in factor analysis. The word scree expresses a distinct binding. For identification of the number of extracted factors, a scree plot is considered for only those factors which are present before different bindings.

C. FACTOR SOLUTION EXTRACTION

This section presents the factor solution extraction with two main steps, i.e. principal components and percentage variance.

1) PRINCIPAL COMPONENTS

We propose to extract factor solutions by using the factor analysis method. Before our proposed work, a three-factor solution was extracted by using the PCA method [43]. Our proposed work is flexible in using either a number of factor solutions. Therefore, we suggest principal label components as PC1, PC2, PC3, and so on. Each extracted principal component has a relationship with multiple web metrics, and hence, it explains a percentage variance among related web metrics. This step aims at deciding how many web metrics contribute to explain the percentage variance of principal components. We aim to obtain the factor solution for all of our chosen datasets that are acceptable and justifiable. It might be acceptable if it gives us an effective reduction in the dimension of web metrics, and it can be made justifiable by using the existing literature on the subject of factor analysis. There are several factor extraction methods, which include PCA, 'principal axis factors' (PAF), and 'maximum likelihood' (ML).

2) PERCENTAGE VARIANCE

Percentage variance is the illustration of the total variance in a matrix. The division of eigenvalue calculates it by the number of variables in a matrix. PCA method is selected for the extraction of factor solution that is available in every statistical program. PCA method is used to measure variance in variables. For factor extraction, one is used in the diagonal of the correlation matrix. The first variable at the highest level gives the highest value of variance; the second variable at the highest level gives the second-highest value, and so on continues until 100% variance among variables is not explained.

D. SIMILARITY FILTERING

The fourth module is proposed to calculate the similarity relationship between web metrics by extracted solutions of four datasets. Before our proposed work, Wang *et al.* [44] used collaborative filtering for quality of services (QoS) recommendation of web services. Web services users' mobility was taken as a critical feature in the proposed approach. We perform similarity filtering to identify the common and dominating web metrics in the four datasets.

1) FUZZY RULES CONSTRUCTION

We apply fuzzy logic to construct a few rules to perform similarity filtering. Zadeh [45] stated that a fuzzy rule used predicates and conditional statements. In this regard, fuzzy logic is compelling in the description of linguistic terms and vague information [46]. In our fourth proposed module, Fuzzy rules' value is taken from Tables (6, 7, 8 and 9) in the evaluation and results section. In the following, we show our constructed fuzzy rules.

Fuzzy Rule 1: If the value of a web metric is $>$ than 0.6 that web metric is significant;

Fuzzy Rule 2: If the value of a web metric is ≤ 0.6 that web metric is not significant;

The construction of a fuzzy rule involves a web metric as an antecedent and then consequent of that web metric. Proposed fuzzy rules are given in a simple form. Membership function determines the degree of truth of a crisp value with the range between 0 and 1. Membership function maps the input to a crisp value. In our proposed work, we use the membership function, namely 'Lagrange Interpolation Membership Function' (LIMF) [47].

Researchers in [60] stated that membership degree, which is higher than 0.6 as indicated as significant. Also, this value has been specified as a fuzzy threshold in the literature.

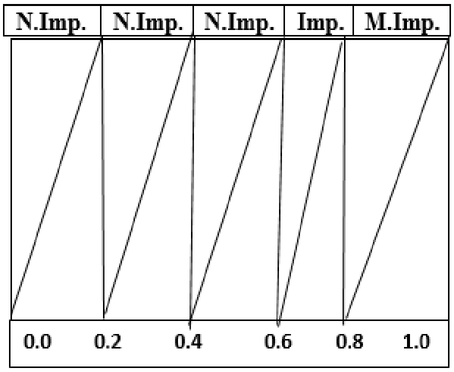


FIGURE 3. LIMF for a quality metric.

So membership degree ≤ 0.6 is not optimal because any value more than 0.6 is transformed into a crisp value [61]. A metric value greater than 0.6 satisfies the more users of web services. Thus we follow the assumption made in [62], [63] because a web metric with threshold value > 0.6 is based on the satisfaction of more users and thus the fuzzy threshold is set to > 0.6.

Figure 3 illustrates the usage of the LIMF membership function, which has been used in our proposed work. In Figure 3, we mean 'N. Imp.', 'Imp.,' and 'M.Imp.' as not important, important and most important web metrics, respectively. Therefore, the value difference between two neighboring labeled web services is 0.20, as shown in Figure 3. From bottom to top, value increases for five categories of web metrics. For instance, the first category of N.Imp. web metrics have value between 0.0 and 0.2, and the second category of N.Imp. web metrics value starts at 0.21 and finishes at 0.4 and the third category of N.Imp.web metrics value lies between 4.1 and 6.0.

Moreover, the value of Imp. class of web metrics lies between 0.61 and 0.80, and the value of M.Imp web metrics starts from 0.81 and continues until value is not reached to 1.0. We manually update fuzzy rules and ranges of linguistic variables. However, the fuzzy rules and their implementation can be extended by following the application of Takagi Sugeno Inference System [54].

2) WEB METRICS INDIVIDUAL SCORING (WMIS)

In the second part of this module, we propose the ranking criteria to show the essential quality metrics which have been used in our work. Our proposed ranking method aims to calculate the ranking score of each web service quality metric. Before we use the proposed WMR method, we calculate the loading score of web metrics, which contribute to a two-factor solution and three-factor solution. For web metrics that contribute to a two-factor solution, we propose the Web Metrics Individual Scoring ($WMIS_a$) method in the following Eq. (3).

$$WMIS_a = \frac{1}{N}\sum_{k=1}^{n}(x+y) \tag{3}$$

Whare 'x' represents load value on PC1, 'y' represents load value on PC2, and N represents the number of load values. Similar to Eq. (3) we propose the ($WMIS_b$) method to compute the individual score of web metrics which contribute to three-factor solution as follows.

$$WMIS_b = \frac{1}{N}\sum_{k=1}^{n}(x+y+z) \tag{4}$$

Latter proposed Eq. (4) includes 'z' as load value on PC3. Above proposed Eq. (3), and Eq. (4) aim at addressing the redundant occurrence of web metrics in contributing to the formulation of two factors and three factors solution, respectively. We can determine the ($WMIS_a$) and ($WMIS_b$) for those web metrics which contribute to both two factors solution and three factors solution.

3) WEB METRICS RANKING (WMR)

For web metrics which have representation in both two factors solution and three factors solution, we obtain a cumulative ranking score by combining the ($WMIS_a$) score of web metrics in datasets which have two factors solution with the ($WMIS_b$) score of web metrics that have three factors solution as given in the following Eq. (5).

$$WMR = \frac{1}{N}\sum_{k=1}^{n}(d1+d2+d3+d4) \tag{5}$$

where d1, d2, and d4 represent ($WMIS_a$) score for datasets A, B, and D, respectively. Moreover, d3 expresses ($WMIS_b$) score of a web metric for the dataset C. In case a web metric stands out only in the first dataset, then d2, d3, and d4 scores are taken as null values for specific web metrics. To ensure that our proposed WMR method efficiently works on variables (quality metrics, items), we recommend to use null values for such type of entities which have a varying number of WMIS scoring for relevant variables.

VI. EVALUATION AND RESULTS

This section presents the experimental assessment of factor analysis about total variance explained. Moreover, we report findings on the ranking of quality metrics by using the WMR method.

A. TOTAL VARIANCE EXPLAINED IN FOUR DATASETS

In the following, the results of the four datasets regarding total variance and eigenvalues have been explained.

Table 2 shows that dataset A has a total of eigenvalue 2.035, and 1.745 for PC1, and PC2 respectively. Here PC1, PC2, and PC3 represent principal component 1, principal component 2, and principal component 3, respectively. These eigenvalues remained the same in the extraction sum of the square loading. In the rotation sums of squared loading, the total value of two principal components, as shown in Table 2, was found with a negligible difference to total values in the

TABLE 2. Total variance explained of dataset A.

Component	Total Eigenvalues	% of Variance in Initial Eigenvalue	Cumulative % Eigenvalue	Total Extraction Value	% of Variance in Extraction	Cumulative % Extraction	Total Rotation	% of Variance in Rotation	Cumulative % Rotation
1	2.035	50.878	50.878	2.035	50.878	50.878	2.030	50.758	50.758
2	1.745	43.623	94.501	1.745	43.623	94.501	1.750	43.743	94.501
3	0.220	5.499	100.000	–	–	–	–	–	–
4	3.271E-11	8.178E-10	100.000	–	–	–	–	–	–

TABLE 3. Total variance explained of dataset B.

Component	Total Eigenvalues	% of Variance in Initial Eigenvalue	Cumulative % Eigenvalue	Total Rotation	% of Variance in Rotation	Cumulative % Rotation
1	2.886	48.099	48.099	2.884	48.068	48.068
2	1.706	28.425	76.524	1.707	28.456	76.524
3	0.877	14.618	91.142	–	–	–
4	0.477	7.948	99.090	–	–	–
5	0.054	0.908	99.998	–	–	–
6	0.000	0.002	100.000	–	–	–

TABLE 4. Total variance explained of dataset C.

Component	Total Eigenvalues	% of Variance in Initial Eigenvalue	Cumulative % Eigenvalue	Total Rotation	% of Variance in Rotation	Cumulative % Rotation
1	4.403	48.921	48.921	3.386	37.620	37.620
2	2.127	23.633	72.553	2.280	25.329	62.948
3	1.128	12.532	85.085	1.992	22.137	85.085
4	0.933	10.364	95.449	–	–	–
5	0.324	3.597	99.046	–	–	–
6	0.070	0.781	99.827	–	–	–
7	0.016	0.173	100.000	–	–	–
8	3.882E-17	4.313E-16	100.000	–	–	–
9	-8.045E-17	-8.939E-16	100.000	–	–	–

initial eigenvalues and extraction sum of the squared loading. Table 2 shows that (PC1) accounts for 50.878% variance, and (PC2) accounts for 43.623% variance.

Table 3 shows that dataset B has a total eigenvalue of 2.886 and 1.706 for PC1, and PC2, respectively. These eigenvalues are found with a negligible difference for the extraction sum of the square loading and rotation sums of squared loading.

Table 4 shows that dataset C has a total of eigenvalue as 4.403, 2.127, and 1.128 for PC1, PC2, and PC3, respectively. These eigenvalues remained the same in the extraction sum of the square loading. In the rotation sums of squared loading, the total values of three principal components, as shown in Table 4, were found to be contrary to those total values of each component in the initial eigenvalues as well as the extraction sum of squared loading values. Table 4 also shows that PC1 accounted for 48.921% variance, PC2 explained 23.633% variance, and PC3 explained 12.532% variance.

Table 4 is the description of three important factors and the loading of 9 quality metrics of web services. The first component has explained the 48.921% variance in dataset C, and it has a high loading (>0.6) from compatibility, reliability, response time, latency, and best practices web metrics. Among these metrics, reliability, and best practices are negatively related to component 2. The second component explains 23.633% variance and has a high loading (>0.6) from success-ability, and availability metrics. The third component explains the least variance as 12.532% with the higher loading (>0.6) from documentation and best practices metrics.

Dataset D has been recently used in a published work [33]. Various quality metrics have been used to rank web services. Dataset D aims at showing us a total variance obtained from six various quality metrics. Table 5 illustrates that total eigenvalues as 1.483, and 1.217 were obtained for PC1 and PC2, respectively.

B. COMPONENT MATRIX OF DATASETS

In the following, we present the component matrix results of four datasets.

Table 6 shows the loading of variables and the estimation of the correlation between each variable and components. Table 6 (rotated components matrix) indicates the loading of four items of the scale on two factors from dataset A. None of the four items has shown a cross-loading both on component 1 and component 2. The reliability metric has a reliable loading on factor 1, while it has a very low and inverse loading on factor 2. Response time metric has a very low loading on PC1, and a high but inverse loading on PC2. The throughput metric of web services has a very low loading on PC1 but has a very high loading on PC2 as given in Table 6. The final metric as an error message has

TABLE 5. Total variance explained of dataset D.

Component	Total Eigen-values	% of Variance in Initial Eigenvalue	Cumulative % Eigenvalue	Total Extraction Value	% of Variance in Extraction	Cumulative % Extraction	Total Rotation	% of Variance in Rotation	Cumulative % Rotation
1	1.483	24.720	24.720	1.483	24.720	24.720	1.480	24.662	24.662
2	1.217	20.289	45.009	1.217	20.289	45.009	1.221	20.347	45.009
3	0.969	16.149	61.158	–	–	–	–	–	–
4	0.934	15.562	76.720	–	–	–	–	–	–
5	0.745	12.409	89.129	–	–	–	–	–	–
6	0.652	10.871	100.000	–	–	–	–	–	–

TABLE 6. Rotated component matrix of dataset A.

Quality Metrics	Component 1	Component 2
Reliability	0.999	-0.009
Response Time	0.140	-0.934
Throughput	0.120	0.937
Error Messages	-0.999	0.009

TABLE 7. Rotated component matrix of dataset B.

Quality Metrics	Component 1	Component 2
Reliability	0.999	-0.009
Class	0.956	0.062
Throughput	0.736	-0.086
Response Time	-0.589	0.509
Best Practices	0.344	0.907
Documentation	0.180	-0.780

a very high loading but inverse on PC1. Table 6 shows that two items (reliability value and error messages) have a strong relation with PC1. It indicates that for dataset A, both items determine the reliability of web services. The second item (error message) has a strong but inverse relationship with PC1. It can be interpreted that two items that associate with the PC1 have opposite directions. For example, an increase in reliability means a decrease in the error messages.

Moreover, it indicates that an increase in the occurrence of error messages results in decreasing web service reliability. The second component, PC2 is much related to the response time and throughput. Response time has an inverse relationship with the PC2.

Table 7 (rotated components matrix) indicates the loading of six items of the scale on two factors from dataset B. Tabachnick and Fidell [48] mentioned 0.32 value as a rule of thumb for a minimum loading of an item. Item 5 showed a cross-loading both on factor 1 with (0.344) and (0.907), and also showed a gap of 0.2 between primary loading and cross-loading. However, a cross-loading >0.3 is not acceptable in general. Similarly, item 3 also shows loading on two components. Item 4 has (-0.589) loading on factor 1, and (0.509) loading on factor 2, respectively. In the first case, negative value shows an inverse relationship between the item and a factor. It indicates that item 4 is in the opposite direction for factor 1 and the right direction for factor 2.

With the help of our proposed research schema, we were able to reduce the dimensions of nine web services quality metrics of dataset C into three common factors, which

TABLE 8. Rotated component matrix of dataset C.

Quality Metrics	Component 1	Component 2	Component 3
Response Time	0.652	0.283	0.010
Availability	0.127	0.917	-0.125
Throughput	-0.585	-0.271	0.560
Successability	0.293	0.918	-0.101
Reliability	-0.863	-0.110	0.458
Compatibility	0.964	0.176	-0.065
Best Practices	-0.657	0.213	0.710
Latency	0.641	-0.578	-0.170
Documentation	-0.002	0.138	-0.952

TABLE 9. Rotated component matrix of dataset D.

Quality Metrics	Component 1	Component 2
Cost	0.713	0.000
Security	0.675	-0.226
Usability	-0.655	-0.189
Respone Time	-0.113	0.784
Flexibility	0.052	-0.575
Availability	0.268	0.434

provided 85.086% as a total explained variance by following the eigenvalue >1 for the standardized factors. Table 8 provides us information regarding nine quality metrics, along with their component values. Item 1 (response time) has a loading value (0.652) on PC1 that is acceptable. Item 2 (reliability) has a loading value (0.917) on PC2. Throughput as an item 3 has loading values (-0.585), (-0.271) and (0.560) on PC1, PC2, and PC3, respectively. Success-ability with loading value (0.981) on PC2 was the fourth item in Table 8, which had the highest loading value on PC2. The reliability quality metric also showed the acceptable loading values (-0.863) and (0.458) on PC1, and PC3, respectively. Compatibility showed (0.964) loading value on PC1.

Table 9 (rotated components matrix) indicates the loading of six items of the scale on two factors from dataset D. Most of the items have shown a cross-loading both on factor 1 and factor 2. Only cost as quality metric has a maximum loading as (0.713) on PC1, while it has a (0.00) loading on PC2. The remaining quality metrics of dataset D have shown cross-loading on both PC1 and PC2 factors. For instance, the security metric has a positive loading (0.675) on PC1 and (-0.226) loading on PC2. Usability has an inverse loading (-0.655) and (-0.189) on PC1 and PC2, respectively. However, it does not mean that usability has no relationship with the two-component factors. Response time has (-0.113) loading on PC1 and (0.784) loading on PC2. It is indicat-

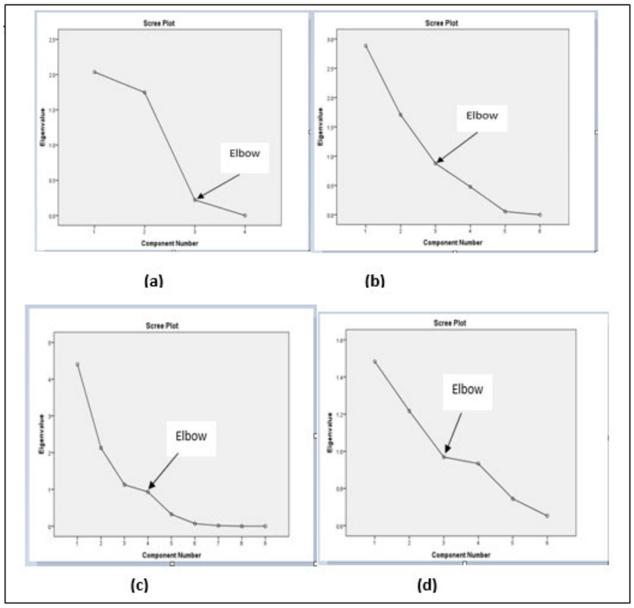


FIGURE 4. Scree plots of four datasets.

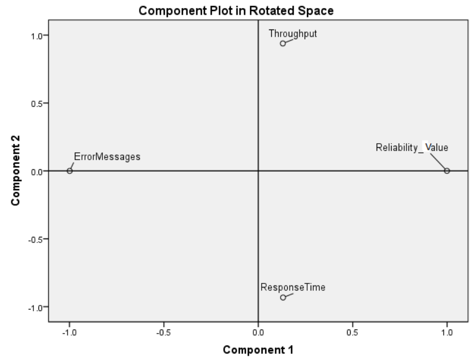


FIGURE 5. Component plot of dataset A.

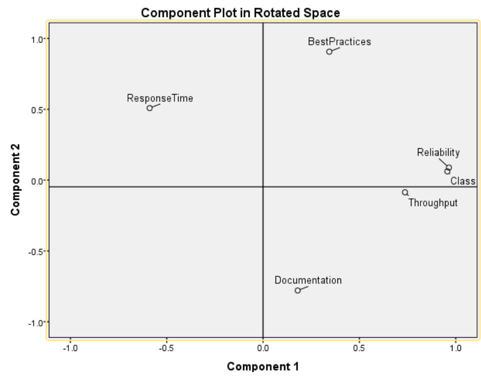


FIGURE 6. Component plot of dataset B.

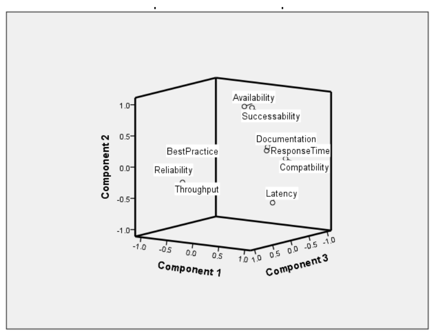


FIGURE 7. Component plot of dataset C.

ing that the response time metric has a strong relationship with PC2. Flexibility quality metric has (0.052) loading on PC1 and (−0.575) loading on PC2. Availability as a quality metric has shown a cross-loading on both PC1 and PC2 factors.

C. SCREE PLOTS AND COMPONENT PLOTS

We present a visual representation of scree and component plots as follows:

Following Figure 4 (a, b, c, and d), is showing us scree plots of four datasets.

Figure 4(a) shows that the distinctive break (elbow) occurs at point 3; thus, we take one factor less than the designated by the relevant break. Therefore, a two-factors solution is deemed fitting to dataset A. Figure 4(b) shows us a break at point 3, which shows that a two-factor solution deemed appropriate for dataset B. Figure 4(c) shows us a break at point 4, and subtracting one by 4 gives us a three-factors solution for the dataset C. Figure 4(d) shows that a break has occurred at point 3. Hence, a two-factors solution is deemed fitting to our used dataset, D. Figure 5 illustrates that PC1 is positively correlated with the reliability value and negatively related to the error messages metric value. These items are located at the right upper and left upper quadrants, respectively. The second component, PC2, is positively correlated with the throughput metric and negatively associated with the response time metric.

Figure 6 shows that the first component PC1 is positively correlated with reliability and class (quality metrics), and negatively correlated with the response time metric. The second component, PC2, is positively correlated with best practices, and negatively correlated with documentation and throughput items (quality metrics).

Figure 7 shows the association between underlying variables and extracted factors based on rotated factor loading (refer to the rotated component matrix). The plot in Figure 7 has three dimensions with positioning of relative factors on it. The PC1 is composed of best practice, reliability, and throughput quality metrics, while the second component PC2 is comprised of availability and success-ability quality metrics. The third component PC3 is comprised of documentation, response time, and compatibility quality metrics of web services. According to Figure 8, it is observed that the first component is correlated with cost and availability items (web metrics) since both of these items are located in the right upper quadrant. The second component is negatively associated with the usability item as it lies in the right lower quadrant.

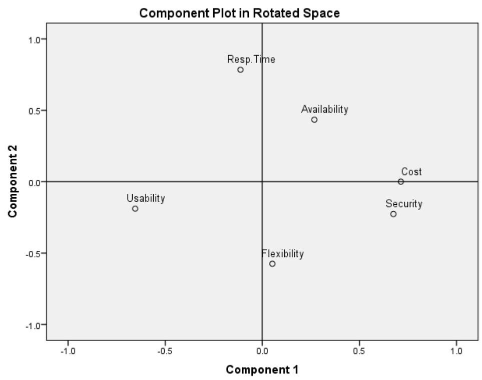


FIGURE 8. Component plot of dataset D.

TABLE 10. WMR score of quality metrics.

Metric Name	$(WMIS_a)$ in dataset A	$(WMIS_a)$ in dataset B	$(WMIS_b)$ in dataset C	$(WMIS.a)$ in dataset D
Reliability	0.495	0.522	0.922	–
Response Time	-0.397	0.040	0.315	0.366
Throughput	0.528	0.325	-0.099	–
Error Messages	-0.495	–	–	–
Class	–	0.509	–	–
Best Practices	–	0.625	0.089	–
Documentation	–	0.300	0.272	–
Availability	–	–	0.306	0.351
Successability	–	–	0.370	–
Compatibility	–	–	0.392	–
Latency	–	–	0.036	–
Cost	–	–	–	0.356
Security	–	–	–	0.225
Usability	–	–	–	-0.422
Flexibility	–	–	–	0.261

TABLE 11. Results of WMR method.

Sr No.	Quality Metric Name	Initial Ranking Score	Ranking Score (%age)	New Ranking
1	Reliability	0.5762	57.62	1
2	Response Time	0.160	16.00	14
3	Throughput	0.2513	25.13	12
4	Error Messages	-0.495	-49.50	3
5	Class	0.509	50.90	2
6	Best Practices	0.357	35.70	7
7	Documentation	0.286	28.60	10
8	Availability	0.3305	33.05	9
9	Successability	0.370	37.00	6
10	Compatibility	0.392	39.20	5
11	Latency	0.036	3.60	15
12	Cost	0.356	35.60	8
13	Security	0.225	22.50	13
14	Usability	-0.422	-42.20	4
15	Flexibility	0.261	26.10	11

Table 10 is presenting the results of our proposed $(WMIS_a)$ and $(WMIS_b)$ methods for calculating the quality metric individual score in four datasets. Obtained individual scores of quality metrics are used to rank them. The proposed WMR method has been applied to the results given in Table 10.

Table 11 is showing us the ranking score as well as the percentage ranking score of web metrics after addressing the occurrence of redundant web metrics in multiple datasets.

TABLE 12. Dominating quality metrics in four datasets.

Categories	Dataset A 2 Factor Solution		Dataset B 2 Factor Solution		Dataset C 3 Factor Solution			Dataset D 2 Factor Solution	
	PC1	PC2	PC1	PC2	PC1	PC2	PC3	PC1	PC2
Performance Factors									
Latency	--	--	--	--	0.641	--	--	--	--
Successability	--	--	--	--	--	0.918	--	--	--
Availability	--	--	--	--	--	0.917	--	--	--
Security Factors									
Best practices	--	--	--	0.92	-0.657	--	0.710	--	--
Reliability	0.999	--	0.960	--	-0.863	--	--	--	--
Security	--	--	--	--	--	--	--	0.675	--
Error Messages	-0.999	--	--	--	--	--	--	--	--
Trust Factors									
Throughput	--	0.937	0.739	--	--	--	--	--	--
Response time	--	-0.934	-0.609	--	0.652	--	--	--	0.784
Class	--	--	0.953	--	--	--	--	--	--
Rapport Factors									
Compatibility	--	--	--	--	0.964	--	--	--	--
Documentation	--	--	--	0.772	--	--	-0.952	--	--
Cost	--	--	--	--	--	--	--	0.713	--
Usability	--	--	--	--	--	--	--	-0.655	--

In the last column of Table 11, we have obtained a new ranking of quality metrics. In the following section, we present a discussion on results.

VII. DISCUSSION

To compare four datasets by using varimax loading, we have extended the categories of web services quality metrics proposed by Arasi1 *et al.* [16]. They categorized the quality metrics into three categories, i.e., performance factors, security factors, and trust factors. It is because datasets, we used contain more quality metrics than those used in the lateral mentioned study. We have added a new category named 'rapport factor' [49], which is either used as positively or negatively to express the relationship of web services users with services providers.

A. DOMINANT QUALITY METRICS AND THEIR CONTRIBUTION

Table 12 shows the contribution and dominance of quality metrics in four datasets. The number of quality metrics in the original datasets were four, six, nine, and six in datasets A, B, C, and D, respectively. Table 12 shows that reliability as a quality metric contributed to the construction of a two-factors solution for each of dataset A, B, and D, and the three-factors solution for dataset C. The error message as a quality metric contributed to the formulation of factor solution with the inverse relationship with PC1. Response time, cost, security and usability metrics participated in the construction of two-factors solution for Dataset D. For dataset A, security factor of web services remained important in explaining the 50.878% variance for PC1, and trust factors including throughput and response time explained 43.623% variance for PC2. For dataset B, reliability quality metric with 0.960 score from security factors contributed along with trust factors in explaining the 48.099% variance for PC1, and best practices and documentation contributed in defining the 28.425% variance for PC1. For dataset C, a three-factors solution showed that quality metrics from all four categories contributed to explaining 48.921% for PC1, and only success-ability and availability quality metrics from performance factors contributed in explaining 23.633% variance for PC2. Best practices and documentation were security and rapport factors respectively, which explained 12.53% variance for PC3 in Dataset C. Besides, reliability, we found that response time is another quality metric that contributed in explaining the percentage variance for either PC1 or PC2 in dataset A, dataset B, dataset C, and dataset D.

Due to substantive interpretation, a three-factor solution or other less dimensional factor solutions are preferred to get better or accurate results. Sometimes three factors have causal relation, and hence they are loaded together. For a higher number of variables, the smaller the factor solution would have better selection and recommendation results [64]. Before our research work, Konosu and Kasahara [65] stated that less dimensional factor solution (two-factors solution and three-factors solution) had better results than the higher dimensional factor solution. Thus our findings indicate that more accurate and useful results can be achieved from the latter mentioned less dimensional factor solutions.

Usability as a quality metric refers to the extent a specific user is satisfied with the use of a software product [50]. A web service with high usability enables users to understand the information displayed on a web page. As usability is showing an inverse relationship with the PC1 in dataset D, we can conclude that this dataset does not fit with the information about the design and constructs of web services. Security as a metric for web services has been kept with the reliability and trust metrics [51]. Therefore, trust in web services is primarily based on whether web services meet users' non-functional quality requirements. Security metric with 0.675 loading indicates that principal component 1 on dataset D is determining the users' trust in web services. Since performance metrics were used in dataset C, and hence show their dominance in the relevant dataset. We cannot say that performance factors did not contribute to explaining other datasets. If they have been taken in other datasets, we would have measured them to show their presence in respective datasets.

Figure 9 is indicating that reliability, throughput, and response time quality metrics have been found as the most critical quality metrics. PC1A, PC1B and PC1C, and PC1D represent principal factor1 for four datasets, respectively. PC2A, PC2B, PC2C, and PC2D represent primary factor 2 for our used four datasets. PC3C is indicating principal factor 3 only for dataset C. This is because that dataset C has given us a three-factor solution as compared to the two-factors solution for the remaining three datasets.

Finally, it has been concluded that factor analysis yielded a two-factors solution for three datasets [29], [30], [33], and the three-factors solution for dataset mentioned in [31]. As shown in Table 2 that PC1 represents the explained maximum percentage variance (50.878) in dataset A, it means that four quality metrics have more significant effects on dataset A. Both, response time and throughput were considered as

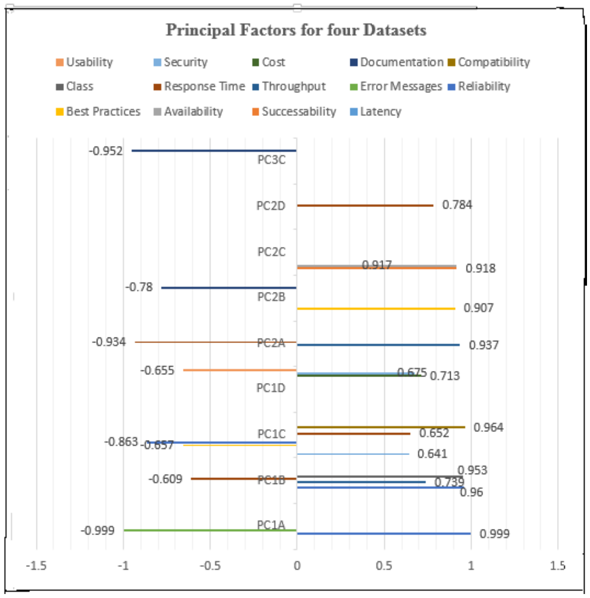


FIGURE 9. Principal factors for four datasets.

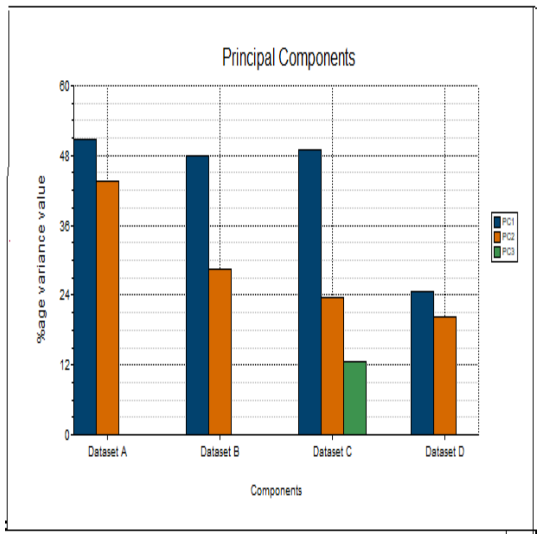


FIGURE 10. Principal components of four datasets with scores.

potential metrics as shown in Table 1. According to results in Table 12, reliability and error messages, along with the response time and throughput, were found outstanding quality metrics. Following this high percentage variance, PC1 in dataset B, and dataset C also showed 48.009 and 48.921% variance values in datasets B and C, respectively.

Figure 10 is showing us the PC1, PC2, and PC3 for four datasets. Percentage variance in PC1 for datasets A, B, and C remained higher than percentage variance in PC1 of dataset D. This could be due to a low degree of correlation between quality metrics. We found that reliability and response time were among quality metrics, which showed an increased percentage variance in explaining the principal components.

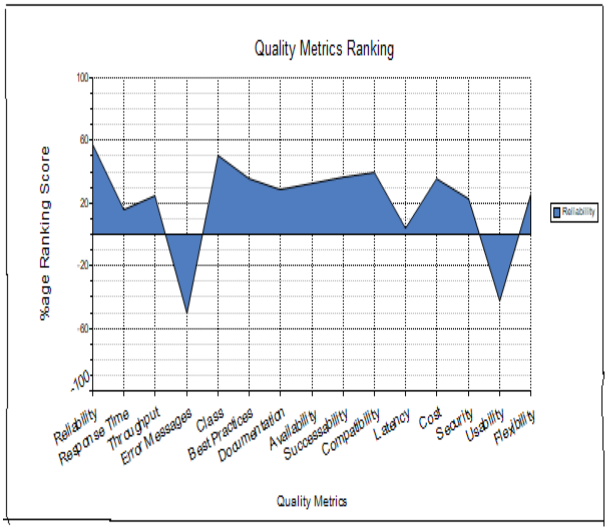


FIGURE 11. Quality metrics ranking.

B. WEB METRICS' RANKING

Figure 11 shows the percentage score of 15 web metrics from our used four datasets. Reliability is predicted as a top-ranked with (57.62%) score followed by class web metric with (50.9%) score, as shown in Figure 11. Only error messages and usability web metrics were found with the –ive percentage scoring from our results. Such a high ranking score of reliability quality metric indicates that it has a high probability of achieving the space among the web metrics of datasets. Based on the computed ranking score of web metrics, we recommend that dataset A with a high score of quality metrics is suitable for future works on web services classification and regression testing.

C. OPTIMAL FACTORS SOLUTION

Findings from this study indicate that datasets with the combination of web metrics (reliability, and response time) have a more significant effect in explaining the percentage variance. Also, the component plots, as shown in Figure 5 and Figure 6, indicate that the two-factor solution remained optimal for dataset A and dataset B, respectively. A significant percentage of total variances was explained by two-factors solution: 94.501%, and 76.524% in dataset A, and dataset B, respectively. However, variance explained by the two-factors solution in dataset D is 45.009% that is lower than 50.00% value. It shows that items contributing to a two-factors solution for dataset D are not sufficient. Another reason behind the low total variance is that multiple items failed to show a prominent association with PC1 and PC2, respectively. According to Sarstedt [52], the total variance must meet minimum criteria of at least 50% variance among all items. They recommended 75% total variance value.

D. EVALUATION OF THE PROPOSED RANKING METHOD

To evaluate the proposed ranking method, we extend the Ranking Precision [53] as given in the following Eq. (6).

$$Ranking\ Precision = \frac{\sum R/r}{Number\ of\ Load.Values} \tag{6}$$

where R is the current ranking/position of a web service metric after ranking, and r is the original position of a web metric given in Table 11, where N is expressing the total number of used web services quality metrics. Therefore, the ranking precision of our proposed WMR method is computed from the list of web services attributes, which are given in Table 11. We achieved 96.17% RP of our proposed WMR method, which is indicating that the proposed method is efficient in showing the ranking precision of web services quality metrics.

E. FUTURE OPPORTUNITY WORKS

From the findings of this study we can undertake a few future research works. Two of them are given in the following.

1) WEB SERVICES CLASSIFICATION VIA MACHINE LEARNING

The first future research work can be focused on the classification of web services. We plan to use the dataset with web metrics which have been computed important with dominating ranking score from our proposed WMR method. If we have a benchmark dataset, which shows maximum desirable features for classification, we can use classifiers on that dataset for binary or multiple classification. Web services classification can be calculated by using various subjective methods, which are trustworthiness, ranking, and scoring. Web metrics classification can help us to calculate users' trust scores of web services. The calculated trust score defines the priority testing of web services.

2) REGRESSION TESTING OF RANKED WEB SERVICES

Our second future work can be focused on regression testing of the ranked web services. Ranking of web services is derived from calculated users' trust in web services. For instance, a pool of users shows their trust in web services through their feedback values; it means a web service performance is better than that web service, which has a low trust score from users. Therefore, regression testing of a web service with a low trust score from a user is conducted before a web service with a high trust score of users.

F. THREATS TO VALIDITY

In this part of the paper, we discuss the validity threats of our web service ranking method for benchmark dataset selection.

A threat to internal validity is the choice of the subject on web services quality metrics. Also, the proposal of benchmark dataset selection from factor analysis may have a few risks regarding the knowledge and understanding of the criteria which support the chosen method. Due to less understanding on the process may affect the results and their interpretations. To mitigate these impacts, we have used enough time to familiarize ourselves with the background information using the literature on factor analysis and its applications in other domains.

The external validity is concerning to the generalization of results from experiments performed in this study. The main external threat to our proposed approach is to its evaluation of the different datasets than the historical information based datasets. Therefore, the applications of machine learning models have been widely used, and performance prediction of web services may be well presented with the time series information of web services. Our work includes only four datasets, and among them, dataset 'A' has a more substantial information from thousands of users located in diverse regions. The rest of the three datasets consist of a smaller number of records of quality metrics, which may impact the ranking accuracy of factor analysis results. Although the execution of our proposed approach shows positive results, we plan to expand our evaluation by recruiting more datasets from accessible dataset repositories.

VIII. SENSITIVITY ANALYSIS

Sensitivity analysis is used to gauge parameters that have a more significant impact on availability. In other words, it means to identify those components which are highly concerning to achieve an increased availability [66]. The focus of sensitivity analysis is to evaluate the QoS metric, which has a more significant impact on web service ranking from metrics' dimension reduction. As shown in Table 11, we presented the sensitivity analysis of ranking of quality metrics computed by the proposed WMR method. Table 11 is presenting all QoS metrics of four datasets. The parameters are shown with the ranking score (%), initial ranking, and new ranking. Individual WMR score of metrics have been given in Table 10. Based on the determined WMR score, we set the priority of quality metrics. An increase in the value of the WMR score is dependent on the increased quality metrics in datasets.

Before our work, Li *et al.* [67] and Ibrahim [68] performed a sensitivity analysis of QoS metrics to analyze their effects on the decision results. However, we follow the ways adopted in [69] for sensitivity analysis. Generally, the new ranking of quality metrics is influenced by the WMR score. If the WMR score of an individual quality metric is higher in percentage, it means it has a higher effect. However, some of the quality metrics have obtained a lower WMR score than the rest of the quality metrics. For instance, latency as a quality metric with a 3.6% WMR score gets the least position in the ranked QoS metrics.

On the other hand, reliability with 57.60% WMR score attained the top position in the ranked QoS metrics. However, a change in the WMR score may affect the ranking of QoS metrics. The WMR score is derived from component matrix of four datasets. For instance, throughput may find place earlier than response time in datasets A and B if its value is changed. Moreover, shift in success-ability and best practice metric values in dataset C may have higher effects if their values are increased. Success-ability may find a place with the top-ranked metrics if its value increases. For dataset D, the usability metric may have higher effects against the reduction of the factors in Dataset D if it receives higher value from users than the earlier mentioned values. Overall, both the WMR score and the number of datasets remained crucial factors for the determination of sensitivity analysis in our study.

IX. CONCLUSION AND FUTURE RESEARCH IMPLICATIONS

Overall, the factor analysis results based on four benchmark datasets provided a two-factors solution for each of dataset A, dataset B, and dataset D, and a three-factors solution for dataset C. A total of 15 web services quality metrics have been categorized in performance, security, trust, and rapport categories of factors. Our factor analysis of web services quality metrics revealed that reliability and response time were among top metrics which explained a high percentage variances. According to results, success-ability and availability were found among the top factors of performance metrics. Reliability and best practices were found top factors among security factors. Response Time and throughput were found the top two factors among trust factors. Documentation and compatibility were found among the prime factors of the rapport category of web metrics. Proposed WMR method results showed that reliability was top-ranked web quality metric, followed by class web quality metric. The proposed WMR method showed us the precision accuracy of 96.17%, which reflects that our proposed ranking method is efficient than the existing ranking methods.

The research implications of this study include the selection and ranking of RESTful web services by our proposed study schema. RESTful web services evaluation requires regression testing. Hence, the ranking of potential quality metrics of RESTful web services by our proposed WMR method can be a leading work in the paradigm of web services. Web services profiling regarding our findings about benchmark dataset is another research implication that gives optimized results only using the least number of quality metrics. The introduction of fuzzy rules in a benchmark dataset selection is the third implication that leads to solve the complex labeling issues and calculate users' trust in web services.

REFERENCES

[1] A. Liu, X. Shen, Z. Li, G. Liu, J. Xu, L. Zhao, K. Zheng, and S. Shang, "Differential private collaborative Web services QoS prediction," *World Wide Web*, vol. 22, no. 6, pp. 2697–2720, Nov. 2019.

[2] M. Garriga, A. D. Renzis, I. Lizarralde, A. Flores, C. Mateos, A. Cechich, and A. Zunino, "A structural-semantic Web service selection approach to improve retrievability of Web services," *Inf. Syst. Frontiers*, vol. 20, no. 6, pp. 1319–1344, Dec. 2018.

[3] D. Ameller, X. Burgués, D. Costal, C. Farré, and X. Franch, "Non-functional requirements in model-driven development of service-oriented architectures," *Sci. Comput. Program.*, vol. 168, pp. 18–37, Dec. 2018.

[4] L. Kuang, L. Yu, L. Huang, Y. Wang, P. Ma, C. Li, and Y. Zhu, "A personalized QoS prediction approach for CPS service recommendation based on reputation and location-aware collaborative filtering," *Sensors*, vol. 18, no. 5, p. 1556, 2018.

[5] W. Serrai, A. Abdelli, L. Mokdad, and Y. Hammal, "Towards an efficient and a more accurate Web service selection using MCDM methods," *J. Comput. Sci.*, vol. 22, pp. 253–267, Sep. 2017.

[6] S. Wen, C. Tang, Q. Li, D. K. W. Chiu, A. Liu, and X. Han, "Probabilistic top-K dominating services composition with uncertain QoS," *Service Oriented Comput. Appl.*, vol. 8, no. 1, pp. 91–103, Mar. 2014.

[7] W. Rong, B. Peng, Y. Ouyang, K. Liu, and Z. Xiong, "Collaborative personal profiling for Web service ranking and recommendation," *Inf. Syst. Frontiers*, vol. 17, no. 6, pp. 1265–1282, Dec. 2015.

[8] X. Luo, H. Luo, and X. Chang, "Online optimization of collaborative Web service QoS prediction based on approximate dynamic programming," *Int. J. Distrib. Sensor Netw.*, vol. 11, no. 8, Aug. 2015, Art. no. 452492.

[9] P. Zhang, H. Jin, Z. He, H. Leung, W. Song, and Y. Jiang, "IgS-wBSRM: A time-aware Web service QoS monitoring approach in dynamic environments," *Inf. Softw. Technol.*, vol. 96, pp. 14–26, Apr. 2018.

[10] H. Wu, K. Yue, B. Li, B. Zhang, and C.-H. Hsu, "Collaborative QoS prediction with context-sensitive matrix factorization," *Future Gener. Comput. Syst.*, vol. 82, pp. 669–678, May 2018.

[11] R. Mohanty, V. Ravi, and M. R. Patra, "Web-services classification using intelligent techniques," *Expert Syst. Appl.*, vol. 37, no. 7, pp. 5484–5490, Jul. 2010.

[12] Y. Wang, Q. He, D. Ye, and Y. Yang, "Formulating criticality-based cost-effective fault tolerance strategies for multi-tenant service-based systems," *IEEE Trans. Softw. Eng.*, vol. 44, no. 3, pp. 291–307, Mar. 2018.

[13] Y. N. Li, K. C. Tan, and M. Xie, "Measuring Web-based service quality," *Total Qual. Manage.*, vol. 13, no. 5, pp. 685–700, Aug. 2002.

[14] S. Gounaris and S. Dimitriadis, "Assessing service quality on the Web: Evidence from business-to-consumer portals," *J. Services Marketing*, vol. 17, no. 5, pp. 529–548, Sep. 2003.

[15] A. M. Aladwani and P. C. Palvia, "Developing and validating an instrument for measuring user-perceived Web quality," *Inf. Manage.*, vol. 39, no. 6, pp. 467–476, May 2002.

[16] F. E. M. Arasi, S. Govindarajan, and A. Subbarayan, "Evaluation of quality of service using factor analysis method," *Indian J. Sci. Technol.*, vol. 9, no. 41, 2016.

[17] M. García-Pineda, J. Segura-García, and S. Felici-Castell, "A holistic modeling for QoE estimation in live video streaming applications over LTE advanced technologies with full and non reference approaches," *Comput. Commun.*, vol. 117, pp. 13–23, Feb. 2018.

[18] C. M. Cheung and M. K. Lee, "The structure of Web-based information systems satisfaction: An application of confirmatory factor analysis," in *Proc. Int. Conf. Web Inf. Syst. Technol.* Barcelona, Spain: Springer, 2007, pp. 257–273.

[19] H. M. Sneed, "Measuring Web service interfaces," in *Proc. 12th IEEE Int. Symp. Web Syst. Evol. (WSE)*, Sep. 2010, pp. 111–115.

[20] D. Baski and S. Misra, "Metrics suite for maintainability of eXtensible markup language Web services," *IET Softw.*, vol. 5, no. 3, pp. 320–341, Jun. 2011.

[21] F. Palma and N. Mohay, "A study on the taxonomy of service antipatterns," in *Proc. IEEE 2nd Int. Workshop Patterns Promotion Anti-Patterns Prevention (PPAP)*, Mar. 2015, pp. 5–8.

[22] J. L. O. Coscia, M. Crasso, C. Mateos, A. Zunino, and S. Misra, "Predicting Web service maintainability via object-oriented metrics: A statistics-based approach," in *Proc. Int. Conf. Comput. Sci. Appl.* Salvador de Bahia, Brazil: Springer, 2012, pp. 29–39.

[23] X.-C. Xiao, X.-Q. Wang, K.-Y. Fu, and Y.-J. Zhao, "Grey relational analysis on factors of the quality of Web service," *Phys. Procedia*, vol. 33, pp. 1992–1998, 2012.

[24] P. Baumann, D. S. Hochbaum, and Y. T. Yang, "A comparative study of the leading machine learning techniques and two new optimization algorithms," *Eur. J. Oper. Res.*, vol. 272, no. 3, pp. 1041–1057, Feb. 2019.

[25] H. Deng, D. Fannon, and M. J. Eckelman, "Predictive modeling for US commercial building energy use: A comparison of existing statistical and machine learning algorithms using CBECS microdata," *Energy Buildings*, vol. 163, pp. 34–43, Mar. 2018.

[26] Z.-Z. Liu, D.-H. Chu, Z.-P. Jia, J.-Q. Shen, and L. Wang, "Two-stage approach for reliable dynamic Web service composition," *Knowl.-Based Syst.*, vol. 97, pp. 123–143, Apr. 2016.

[27] Q. Ma, J. M. Pearson, and S. Tadisina, "An exploratory study into factors of service quality for application service providers," *Inf. Manage.*, vol. 42, no. 8, pp. 1067–1080, Dec. 2005.

[28] Z. Chen, L. Shen, and F. Li, "Exploiting Web service geographical neighborhood for collaborative QoS prediction," *Future Gener. Comput. Syst.*, vol. 68, pp. 248–259, Mar. 2017.

[29] K. Su, L. Ma, B. Xiao, and H. Zhang, "Web service QoS prediction by neighbor information combined non-negative matrix factorization," *J. Intell. Fuzzy Syst.*, vol. 30, no. 6, pp. 3593–3604, Apr. 2016.

[30] L. Lu and Y. Yuan, "A novel TOPSIS evaluation scheme for cloud service trustworthiness combining objective and subjective aspects," *J. Syst. Softw.*, vol. 143, pp. 71–86, Sep. 2018.

[31] E. Al-Masri and Q. H. Mahmoud, "Investigating Web services on the world wide Web," in *Proc. 17th Int. Conf. World Wide Web (WWW)*, 2008, pp. 795–804.

[32] T. Bouasker and M. Langar, "Consumer oriented Web services ranking and classification," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2017, pp. 1–7.

[33] A. Ezenwoke, O. Daramola, and M. Adigun, "QoS-based ranking and selection of SaaS applications using heterogeneous similarity metrics," *J. Cloud Comput.*, vol. 7, no. 1, p. 15, Dec. 2018.

[34] P. Kline, *An Easy Guide to Factor Analysis*. New York, NY, USA: Routledge. 2014.

[35] T. E. Ayyildiz and A. Koçyiğit, "Size and effort estimation based on problem domain measures for object-oriented software," *Int. J. Softw. Eng. Knowl. Eng.*, vol. 28, no. 2, pp. 219–238, Feb. 2018.

[36] D. Bianchini, V. De Antonellis, and M. Melchiori, "The role of developers' social relationships in improving service selection," *Int. J. Web Inf. Syst.*, vol. 12, no. 4, pp. 477–503, Nov. 2016.

[37] M. Tang, X. Dai, J. Liu, and J. Chen, "Towards a trust evaluation middleware for cloud service selection," *Future Gener. Comput. Syst.*, vol. 74, pp. 302–312, Sep. 2017.

[38] L. Qi, W. Dou, and J. Chen, "Weighted principal component analysis-based service selection method for multimedia services in cloud," *Computing*, vol. 98, nos. 1–2, pp. 195–214, Jan. 2016.

[39] H. F. Kaiser, "The varimax criterion for analytic rotation in factor analysis," *Psychometrika*, vol. 23, no. 3, pp. 187–200, Sep. 1958.

[40] S. Grover, D. Dua, S. Chakrabarti, and A. Avasthi, "Factor analysis of symptom dimensions (psychotic, affective and obsessive compulsive symptoms) in schizophrenia," *Asian J. Psychiatry*, vol. 38, pp. 72–77, Dec. 2018.

[41] A. Lovik, V. Nassiri, G. Verbeke, G. Molenberghs, and A. K. Sodermans, "Psychometric properties and comparison of different techniques for factor analysis on the big five inventory from a flemish sample," *Pers. Individual Differences*, vol. 117, pp. 122–129, Oct. 2017.

[42] M. Morchid, R. Dufour, P.-M. Bousquet, G. Linarès, and J.-M. Torres-Moreno, "Feature selection using principal component analysis for massive retweet detection," *Pattern Recognit. Lett.*, vol. 49, pp. 33–39, Nov. 2014.

[43] A. Bai, S. Hira, and P. S. Deshpande, "An application of factor analysis in the evaluation of country economic rank," *Procedia Comput. Sci.*, vol. 54, pp. 311–317, 2015.

[44] S. Wang, Y. Zhao, L. Huang, J. Xu, and C.-H. Hsu, "QoS prediction for service recommendations in mobile edge computing," *J. Parallel Distrib. Comput.*, vol. 127, pp. 134–144, May 2019.

[45] L. A. Zadeh, "Fuzzy logic, neural networks, and soft computing," in *Fuzzy Sets, Fuzzy Logic, And Fuzzy Systems: Selected Papers*. London, U.K.: World Scientific, 1996, pp. 775–782.

[46] I. B. Turksen, "Recent advances in fuzzy system modeling," in *Frontiers of Higher Order Fuzzy Sets*. New York, NY, USA: Springer, 2015, pp. 51–66.

[47] S. K. De, A. Goswami, and S. S. Sana, "An interpolating by pass to Pareto optimality in intuitionistic fuzzy technique for a EOQ model with time sensitive backlogging," *Appl. Math. Comput.*, vol. 230, pp. 664–674, Mar. 2014.

[48] B. G. Tabachnick and L. S. Fidell, *Using Multivariate Statistics*, 4th ed. Boston, MA, USA: Allyn and Bacon, 2001.

[49] A. Bilgihan and M. Bujisic, "The effect of Website features in online relationship marketing: A case of online hotel booking," *Electron. Commerce Res. Appl.*, vol. 14, no. 4, pp. 222–232, Jul. 2015.

[50] A. Oztekin, "A decision support system for usability evaluation of Web-based information systems," *Expert Syst. Appl.*, vol. 38, no. 3, pp. 2110–2118, Mar. 2011.

[51] S. Wang, H. Zou, Q. Sun, and F. Yang, "Bayesian approach with maximum entropy principle for trusted quality of Web service metric in e-commerce applications," *Secur. Commun. Netw.*, vol. 5, no. 10, pp. 1112–1120, Oct. 2012.

[52] M. Sarstedt and E. Mooi, *A Concise Guide to Market Research: The Process, Data, and Methods Using IBM SPSS Statistics*. Berlin, Germany: Springer-Verlag, 2018.

[53] O. Koval, S. Nabareseh, P. Klimek, and F. Chromjakova, "Demographic preferences towards careers in shared service centers: A factor analysis," *J. Bus. Res.*, vol. 69, no. 11, pp. 4798–4803, Nov. 2016.

[54] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," in *Readings in Fuzzy Sets for Intelligent Systems*. Amsterdam, The Netherlands: Elsevier, 1993, pp. 387–403.

[55] M. Ahmadi Oskooei and S. Mohd Daud, "Quality of service (QoS) model for Web service selection," in *Proc. Int. Conf. Comput., Commun., Control Technol. (I4CT)*, Sep. 2014, pp. 266–270.

[56] H. Wang, M. Kessentini, and A. Ouni, "Interactive refactoring of Web service interfaces using computational search," *IEEE Trans. Services Comput.*, to be published.

[57] L. V. Thinh, "Qos prediction for Web services based on restricted Boltzmann machines," *J. Service Sci. Res.*, vol. 9, no. 2, pp. 197–217, Dec. 2017.

[58] J. Li and J. Lin, "A probability distribution detection based hybrid ensemble QoS prediction approach," *Inf. Sci.*, vol. 519, pp. 289–305, May 2020.

[59] Z. Zheng, Y. Zhang, and M. R. Lyu, "CloudRank: A QoS-driven component ranking framework for cloud computing," in *Proc. 29th IEEE Symp. Reliable Distrib. Syst.*, Oct. 2010, pp. 184–193.

[60] M. Koyuncu and A. Yazici, "IFOOD: An intelligent fuzzy object-oriented database architecture," *IEEE Trans. Knowl. Data Eng.*, vol. 15, no. 5, pp. 1137–1154, Sep. 2003.

[61] S. Martinis, J. Kersten, and A. Twele, "A fully automated TerraSAR-X based flood service," *ISPRS J. Photogram. Remote Sens.*, vol. 104, pp. 203–212, Jun. 2015.

[62] K. K. Borah, S. Roy, and T. Harinarayana, "Optimization in site selection of wind turbine for energy using fuzzy logic system and GIS—A case study for Gujarat," *Open J. Optim.*, vol. 2, no. 4, pp. 116–122, 2013.

[63] A. Z. Dhunny, J. R. S. Doorga, Z. Allam, M. R. Lollchund, and R. Boojhawon, "Identification of optimal wind, solar and hybrid wind-solar farming sites using fuzzy logic modelling," *Energy*, vol. 188, Dec. 2019, Art. no. 116056.

[64] D. Goretzko, T. T. H. Pham, and M. Bühner, "Exploratory factor analysis: Current use, methodological developments and recommendations for good practice," *Current Psychol.*, pp. 1–12, May 2019, doi: 10.1007/s12144-019-00300-2.

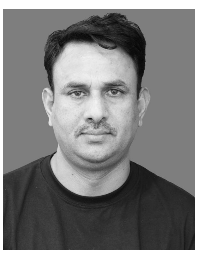
[65] S. Konosu and K. Kasahara, "Multiple fatigue crack growth prediction using stress intensity factor solutions modified by empirical interaction factors," *J. Pressure Vessel Technol.*, vol. 134, no. 1, Feb. 2012.

[66] J. Dantas, R. Matos, J. Araujo, D. Oliveira, A. Oliveira, and P. Maciel, "Hierarchical model and sensitivity analysis for a cloud-based VoD streaming service," in *Proc. 46th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. Workshop (DSN-W)*, Jun. 2016, pp. 10–16.

[67] L. Li, M. Liu, W. Shen, and G. Cheng, "Recommending mobile services with trustworthy QoS and dynamic user preferences via FAHP and ordinal utility function," *IEEE Trans. Mobile Comput.*, vol. 19, no. 2, pp. 419–431, Feb. 2020.

[68] A. A. Z. A. Ibrahim, "PRESEnCE: A framework for monitoring, modelling and evaluating the performance of cloud SaaS Web services," in *Proc. 48th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Netw. Workshops (DSN-W)*, Jun. 2018, pp. 83–86.

[69] A. Ouadah, A. Hadjali, F. Nader, and K. Benouaret, "SEFAP: An efficient approach for ranking skyline Web services," *J. Ambient Intell. Humanized Comput.*, vol. 10, no. 2, pp. 709–725, Feb. 2019.



MUHAMMAD HASNAIN was born in Bhakkar, Punjab, Pakistan, in 1977. He received the M.Sc. degree in computer science from Abasyn University Islamabad, Pakistan, in 2016. He is currently pursuing the master's degree with the School of Information Technology, Monash University Malaysia. From 2016 to 2017, he worked as a Lecturer with the Army Public College of Management Sciences Rawalpindi Pakistan. His research interest is focused on Web services quality enhancement.

MUHAMMAD FERMI PASHA (Member, IEEE) was born in Indonesia. He received the Ph.D. degree in computer science from Universiti Sains Malaysia, in 2010.

After his Ph.D. degree, he worked as a Research Fellow with Universiti Sains Malaysia. He is currently working as a Lecturer with the School of Information Technology, Monash University Malaysia. He is also supervising the PhD. students in the latter mentioned research areas. His research interest is focused on computational neuroimaging, intelligent network security traffic analysis, and healthcare and radiology IT with emphasis on big data.

IMRAN GHANI was born in Pakistan. He received the Ph.D. degree from Kookmin University, South Korea, in 2010, and the M.Sc. degree in computer science from UTM, Malaysia, in 2007. He worked as a Senior Lecturer with Monash University Malaysia. He is currently working as an Associate Professor of computer science with the Mathematical and Computer Science Department, Indiana University of Pennsylvania. He is also supervising the Ph.D. students in the latter mentioned research areas. He has published more than 80 research articles in reputed journals. He also edited two books. His research interests include software engineering, Web services, Web mining, and cloud computing.

BILAL MEHBOOB received the M.S. degree in software engineering from the National University of Science and Technology (NUST), Pakistan, in 2017. He is currently pursuing the Ph.D. degree with Monash University. His current research interests include software engineering, machine learning, and data mining.

MUHAMMAD IMRAN was born in Lahore, Pakistan. He received the master's degree in computer science from COMSATS University Lahore, Pakistan. He is currently working as a Senior Software Engineer with Next Bridge (Pvt.) Ltd., Lahore. His research interests include data mining, machine learning, and software engineering.

AITIZAZ ALI received the M.S. degree in computer science from the Ghulam Ishaq Khan Institute of Engineering Sciences and Technology. He is currently pursuing the master's degree with the School of Information Technology, Monash University Malaysia. His research interests include networking, cyber security, and trust management.