CLOUD SERVICE SCRUTINIZATION, ESTIMATION AND SELECTION FRAMEWORK (CSESF) USING KNN – COSINE METRICS

Sudheer Shetty

Associate Professor, Department of ISE, Alva's Institute of Engineering & Technology, Mangaluru,
Affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India
sudheershetty06@gmail.com

A P Manu

Professor, Department of CSE, PES Institute of Technology & Management, Shivamogga,
Affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India
apmanu@gmail.com

Pavan Kumar V

Research Scholar, Department of CSE, PES Institute of Technology & Management, Shivamogga,
Affiliated to Visvesvaraya Technological University, Belagavi, Karnataka, India
sadgurupavan@gmail.com

Abstract

Cloud Service Selection is still one of the critical decisions that have the highest impact on any association. Making a decision scrutinizing them with estimation is always a complicated task. This paper implemented a three-step process in service selection: Cloud Service Scrutinization, Assessment, and Selection Framework (CSESF) using K Nearest Neighbor – cosine metrics. In this method, all the decision-maker's requirements are initially identified and listed. All the parameters and service providers are listed. An algorithm for scrutinizing is developed. Weights are calculated for preferences given by the decision-makers. KNN with cosine metrics works on choosing the service with the smallest for the positive best solution and harmful for the far solution. The results show that CSESF is robust, practical, and suitable for choosing the service.

***Keywords*: Cloud Service Selection, KNN Cosine Metrics**

1. Introduction:

The solution to store the data, in coming generation is Cloud. The core concept of the Cloud is Virtualization. This enables to share interconnect and manage the computers in a distributed network. For decision-makers, one of the biggest challenges is to provide dynamic services to the user based on their requirements. Over the internet, identifying the accurate and effective service has become a most significant challenge. This leads to attract both the industry and academia. Many studies were conducted to educate users to make wise decisions.

Most of the studies try to solve problems of two types. 1. Developing the standard interface to develop a solution to this problem, a platform that helps users choose the displayed services. 2. One the basis of decision-making methods is selecting suitable cloud services. This paper focuses on cloud service selection.

We found exciting details in the literature about the MCDM methods used for service selection. The AHP is the most commonly used MCDM method than any other method. The use of assessment methods and targeted cloud service layers, where most of the selection techniques are applied to all platforms and different methodologies are focused on individual cloud service layers. Based on their use in standard service selection, the QoS parameters considered for service evaluation are exhibited. We thought the top five QoS parameters for service evaluation, except for security parameters.

If the number of alternative services and their QOS criteria are finite, it is better to evaluate these services concerning QoS by the MCDM methods. The Analytical Hierarchy Process (AHP) is a common MCDM method for solving the problem of cloud service selection (Saaty, 1990). The summary of Mcdm methods used to solve the problem of cloud service selection. The AHP subdivides a complex service selection problem into a hierarchical structure comprising criteria and alternatives. The criteria weights are generated from the hierarchical system by developing a pairwise comparison matrix.

In order to differentiate whether the measurement of the cloud is used, one must first reflect on the service that uses it and deliberate on what compensation the cloud consumes. There are three categories of stakeholders: providers of cloud services, cloud users and end users Cloud providers provide cloud customers with cloud facilities. These cloud facilities are part of the utility computing phase, i.e. the cloud providers use the pay-as-you-go model of these facilities. Using these services, cloud users advance their innovation and bring the invention to the final users.

This paper organized in the following way, Section 2. Contains the related work by other researchers, Section 3 provide the detail information about the proposed work and algorithms implemented. The results and discussion of the cloud service selection are listed in Section 4 and Concluded in Section 5.

2. Related Work:

An AHP-based cloud service ranking model was suggested by Garg et al. (2011), in which cloud services are assessed and ranked according to key performance indicators (KPIs). In the first step, the hierarchy structure is built on the basis of candidate services and their parameters for QoS. Local weights are extracted in the second step by performing the pairwise comparison. The software as a service framework has been proposed to select a specific service type (i.e. IaaS, PaaS, and SaaS) (Godse and Mulik, 2009) to select application services by assessing the alternatives and requirements according to the AHP model. Chen et al. (2011) mapped the AHP model to assess the product services of sales force automation by creating the three level hierarchies that constitute alternative software services and their sub-criteria criteria. Weights of local and global parameters are built on the basis of user expectations and the opinion of the subject matter expert. In order to evaluate and rank both IaaS and SaaS services.

Karim et al. (2013) developed a layered architecture. The AHP model weighs the parameters for determining alternative cloud services on the basis of consumer requirements. By matching user QoS and composite service QoS, the model is successful at evaluating the composition of IaaS and SaaS services.To choose and rank cloud infrastructure services, Menzel et al. (2013) developed a multi comparison approach for cloud computing (MC2)2. In order to analyze a service's QoS in terms of advantages, expense, risk and opportunities, the analytical networking method (ANP) is adopted.

A cloud genius has been developed by Menzal and Ranjan (2012) as a multi-criteria decision support framework that helps migrate current web services to the cloud platform. For the benefit of different advantages from both the service provider and the customer side, most web services are migrated to the cloud.

A mathematical model for multi-sourcing schemes was devised by Martens and Teuteberg (2012) to choose cloud services on the basis of cost and risk factors. The AHP helps to understand the risk factors in the approaches to Selection. In AHP, it is not certain to evaluate criteria weights by conducting pairwise comparison. Within the interval [0, 1], the fuzzy methodology is used to represent user preferences and the opinion of the subject expert. The membership functions and rules are defined according to the criteria in order to choose the preference that suits real world situations. The (Ashtiani et al., 2009) proposed a fuzzy AHP in order to determine criteria weights to meet user preferences in the defined fuzzy interval.

Chen and Lin (2009) proposed a flexible decision support system to express the subjective user preferences for each criterion. The subjective preferences are converted into the form of the numerical preference matrix, and then the matrix is assessed to rank the alternative services as per the analytical hierarchy process.

Another popular MCDM method to resolving cloud service selection problem is the TOPSIS (Technique for Order Preference by Similarity to an Ideal Solution) (Lo et al., 2010). By constructing triangular fuzzy numbers, they derived criterion weights for alternative services. A numerical example illustrates the proposed method.

In order to test alternative services to satisfy customer needs, Cheng et al. (2011) integrated a service-oriented architecture with fuzzy TOPSIS. The system addresses dynamic changes and converts uncertain user preferences into appropriate preferences. The hybrid model proposed achieves better user satisfaction results and overall computational efficiency.

The outranking methods in which the service rank is associated with every alternative in a pair-wise comparison. In managing more alternatives and QoS parameters, the outranking methods are good. ELECTRE (Elimination and Choice Expressing Reliability) is adopted by applying concordance and dis-concordance indices to construct a partial ranking structure of alternative services.

By applying an overarching method called ELECTRE, Silas et al. (2012) solves a cloud service selection problem. The level of credibility is created by the customer's requirement and the service provider's information; the degree of credibility is the composition of agreement and disagreement of every criterion.

Service selection is carried out by optimizing or minimizing one or more constraint parameters by applying optimization techniques. Chang et al. (2012) suggested a model for choosing cloud storage from a range of storage services available. The proposed model is based on dynamic programming and the formulation of cost and objective function measurements. The limitations for choosing cloud storage are data durability, and customer preferences should be within the budget. Considering the maximum durability and the minimum number of failures, storage services are chosen.

3. Proposed Framework

The proposed CSESF for Requirement analysis, scrutinization, estimation and Selection using cosine metric KNN.

3.1. *Requirement Analysis:*

The assumptions made in this research are, end user or organization or entity that are in of cloud service, must have intermediate level expertise in opting the parameters of the service given. This is a phase where, conclusions or requirements of services is given on the basis of team meetings, slight research on the applications they want to deploy or use on the cloud. For examining the requirements identified aids as the base.

3.2. *Scrutinization:*

In this phase of CSESF, from a pool of cloud services suitable services are selected for further processing. Previous step of analysis supports this phase. The decision makers in the organization must be very careful while listing the requirements. The conjunctive screening is one of the appropriate ways of selecting the services. This approach does not select the services that meet the minimum criteria. In this phase, a set of apt services will be provided. i.e., $aS = \{sr_1, sr_2, sr_3, \dots sr_4\}$

```
Algorithm: Scrutinization (CSESF)
Inputs: Services: nSR, Requirements: kR
Output: Apt Services: aS
1.  i ← |kR|
2.  for j = 1: n do
        result = checkCompatibility (nSR[j], kR[j])
        if result = True then
            available = checkAvailability(nSR[j])
            if available = false then
                Add nSR[j] to aS
3.  end if
4.  end if
5.  end for
6.  return aS;
```

3.3. *Estimation:*

On the basis of the criteria, weights are calculated for preferences given by the decision makers. Here, by considering multiple criteria the below algorithm is developed. This helps to take a decision in its early stage and identify it best and worst cases.
Step1. Limit the Decision Measures:
Step2. Define the best and worst cases. Best refers to the most valuable and work refers to less prominent criteria.
Step3. Design accord on best and worst cases. When multiple decision finalizers are there, step 3 activates where, detailed analysis, consolidation of information as the requirements, then according the best and worst cases.
Step 4. Gaining best case preferences.
Step 5. Find Aggregate of best cases vector

$$\boldsymbol{TA_B} = (\boldsymbol{ta_{B1}}, \boldsymbol{ta_{B2}}, \boldsymbol{ta_{B3}}, \dots . \boldsymbol{ta_{Bn}}) \tag{1}$$

Step 6. Obtaining the worst-case vector

$$\boldsymbol{A_w} = (\boldsymbol{a_{w1}}, \boldsymbol{a_{w2}}, \boldsymbol{a_{w3}}, \dots . \boldsymbol{a_{wn}}) \tag{2}$$

Step 7. Calculating the Total Worst Vector

$$\boldsymbol{TA_w} = (\boldsymbol{ta_{w1}}, \boldsymbol{ta_{w2}}, \boldsymbol{ta_{w3}}, \dots . \boldsymbol{ta_{wn}}) \tag{3}$$

Step 8. Total optimal updated weights. In this last stage, update weights are calculated that minimizes the differences for all the values, $\{|agw_B - aa_{Bj} * agw_j|, |agw_j, aa_{jw} * ag_w|\}$ where total weight is indicated by agw.

$$\min \max\{|agw_B - aa_{Bj} * agw_j|, |agw_j, aa_{jw} * ag_w|\} \tag{4}$$

Algorithm: Evaluation (CSESF)

Inputs: aS // calculation set of priorities
Output: u_w // calculated weights
1. num = |TA| // no. of Task Assigners
2. tM ← allocate a set of task measures
3. bM ← define the best measure
4. wC ← derive the worst measure
5. *for* ta = 1:n *do* // for each ta
   $A_B[i]$ ← determine the vector
   $A_w[i]$ ← determine the ow vector
   totalB += $A_B[i]$
   totalW += $A_w[j]$
6. *end for*
7. TA_B = total_mean(totalB)
8. TA_w = total_mean(totalW)
9. d ← |dc|
10. *for* j = 1:d *do*
11. U_w ← AGW (TA_B,TA_W)
12. *end* for
13. *return* u_w

3.4. *Selection:*

By implementing KNN with cosine metrics, it works on choosing the service with smallest from the positive best solution and negative for the far solutions. It has few steps involved.

1. Developing a matrix as follows:

$$r_{ij} = \frac{x_{ij}}{\sqrt{\sum_{i=1}^{n} x_{ij}^2}} \; for \; i = 1,..m, j = 1, \dots n \tag{5}$$

2. Weighted Matrix

$$V_{ij} = W_{ij} r_{ij} \tag{6}$$

3. Defining the desirable and undesirable model solutions (S^+ and S^-) as follows:

$$A^+ = \{(\max V_{ij} \mid j \epsilon J), (\min V_{ij} \mid j \epsilon J`) \; i = 1, 2, \dots . m \} \tag{7}$$

$$A^- = \{(\min V_{ij} \mid j \epsilon J), (\max V_{ij} \mid j \epsilon J`) \; i = 1, 2, \dots . m \} \tag{8}$$

4. When we get more than one solution, then calculate the individual measures, with the distance formula for KNN

$$Positive\ (Sr_i) = \left[\sum_{j=1}^{n} (V_{ij} - V_j{}^+)^2\right]^{1/2}, i = 1, 2, 3, .. m \tag{9}$$

$$\boldsymbol{negative}\ (Sr_i) = \left[\sum_{j=1}^{n} \left(V_{ij} - V_j^{-}\right)^2\right]^{1/2},\ i = 1, 2, 3, ..\ m \tag{10}$$

5. Most relative services are calculated for every ideal solution

$$Rl_i^{*} = \frac{Sr_i^{-}}{Sr_i^{+}}, 0 < Rl_i^{*} < 1 \tag{11}$$

3.5. *KNN with Cosine Metrics:*

Machine Learning based algorithms gained its prominence in pattern recognition and data field. In our proposed model we developed KNN in the cloud environment for selecting the cloud service from various providers.
By considering the different parameters that are opted by the user, from the list of cloud service providers, the proposed system selects the suitable service providers. Different functions that provided to the end user.

- Displaying the list of CSP's and their services.
- Displaying priorities for the parameter of services provided by the cloud.
- Allowing the user to select the services and parameters for each service selected.
- Displaying the best CSP to user based on services and parameters selected by the user.

Service provider is selected by using the cosine similarity of the nearest neighbour algorithm. Imagining that the user have prior understanding towards the cloud computing and knowledge about the services, a user interface is developed to select the various parameters and different kinds of services.
The different services mentioned are: storage, servers, network, cloud SQL, RDS, VMWARE hosting, DNS, APP Engine, Files, Data flow. The parameters for these services are, Reliability, cost, QoS, Scalability, Security, Availability, Performance, Sustainability, Interoperability.

Proposed Algorithm: Service Selection using KNN

Algorithm: Service Selection using NN

Input: User selects the services and parameters
Output: Returns the best CSP for each service
1. Begin
2. *for* C each CSP
    select the CSP which contain all the service which are given by user
    if(selected_services == services in CSP)
    Select the CSP
3. *end*
4. *end for*
5. *for each* selected_csp
    Calculate the final array
    Final_array=User_request-CSP_ranking
6. *end for*
7. *for each* selected_csp
    Calculate cosine distance and cosine similarity
    Cosine_distance=1-(|final_array||User_request|)/(√($\boldsymbol{final\_array}$)2+√($\boldsymbol{user\_request}$)2
    Cosine_similarity=1-cosine_distance
8. *end f*or
9. *for* each cosine_similarity
    Select the maximum cosine_similarity
    if(max_cos_similarity <cosine_similarity)
        max_cos_similarity =cosine_similarity
    end for
10. *return*<max_cos_similarity>

Generally, distance metrics helps to identify the pattern in order to take a decision. Distance metrics in KNN assists to improve the performance of classification. Distance functions, calculates the distance between two elements of a set. This function may vary for variant metrics. One of the most used distance formulae is Euclidean.

$$\boldsymbol{d(x,y)} = \sqrt{\sum_{i=1}^{n}(\boldsymbol{x_i} - \boldsymbol{y_i})^2} \tag{12}$$

Cosine Distance metric is assistances to find similarities. The degree of similarity between two vectors are measured. The services and parameters selected by the user and parameters and services provided by service provider are collected as metrics. Cosine metric used when the orientation is measured but not the magnitude of the vectors. This is derived by dot product of vectors.

$$\vec{\boldsymbol{a}}.\vec{\boldsymbol{b}} = ||\vec{\boldsymbol{a}}|| \, ||\vec{\boldsymbol{b}}|| \cos\boldsymbol{\theta} \tag{13}$$

$$\cos\boldsymbol{\theta} = \frac{\vec{\boldsymbol{a}}.\vec{\boldsymbol{b}}}{||\vec{\boldsymbol{a}}|| \, ||\vec{\boldsymbol{b}}||} \tag{14}$$

To find the similarities, the angles that are helpful are $\cos 0 = 1$, $\cos 90 = 0$, and $\cos 180 = -1$. The cosine value 1 for the similar vectors and 0 for the vectors that are slightly similar can be called as orthogonal vectors and -1 for the vectors that are in opposite direction.

Basically, this is a set similarity problem, this problem worsens when the parameter and services required are more. By using Cosine and KNN, the similarity between two sets can be determined better. Equation (2) and (3) helps to calculate the distance of the observations in multi-dimensional space. Generally, a multi-dimensional space is represented using a vector. Where each parameter/ service represents a dimension and each observation is characterized as a point in this space.

4. Results and Discussions:

In this session, we present the details our outcomes of all the three phases. The parameters considered in the Selection of appropriate cloud service. The cloud selection requirements of any individual or organization must be identified as a sample is given below.

The different services mentioned are: storage, servers, network, cloud SQL, RDS, VMWARE hosting, DNS, APP Engine, Files, Data flow. The parameters for these services are, Reliability, cost, QoS, Scalability, Security, Availability, Performance, Sustainability, Interoperability. Services and Parameters are mentioned in the table 1. The standards recognized are Performance of CPU, DISK, and Memory, Quantity, Cost is another factor mostly considered when it is reliable and availability. On the basis of cloud scrutinization, the most suitable is mentioned as Y and not suitable is mentioned as N in the Table 2. On the basis of weights calculation from algorithm 2 the comparison of five decision and values generated/ assigned to the parameters mentioned in Table 3. Ranking of the service providers

Services	Symbol	Parameters	Symbol
Storage	S1	Reliability	P1
Servers	S2	Cost	P2
Network	S3	QoS	P3
Cloud SQL	S4	Scalability	P4
RDS	S5	Security	P5
VMWARE Hosting	S6	Availability	P6
DNS	S7	Performance	P7
APP Engine	S8	Sustainability	P8
Files Server	S9	Interoperability	P9

Table 1. Different Parameters and Services listed

Services	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
S1	Y	Y	N	Y	Y	Y	Y	Y	Y	Y
S2	Y	N	Y	N	Y	N	Y	Y	Y	Y
S3	Y	Y	N	Y	Y	N	Y	Y	N	Y
S4	Y	N	Y	N	Y	Y	N	Y	Y	Y
S5	Y	Y	N	Y	N	Y	N	Y	N	Y
S6	Y	Y	Y	Y	Y	N	Y	Y	Y	Y
S7	N	N	N	Y	Y	Y	N	N	Y	Y
S8	N	Y	Y	N	Y	Y	Y	N	Y	Y
S9	Y	N	N	N	N	Y	Y	Y	Y	N

Y – Most compatible, N – Not compatible

Table 2. Services scrutinization at phase I

Decision	P1	P2	P3	P4	P5	P6	P7	P8	P9
1	1	9	3	2	2	3	3	3	3
2	1	9	3	3	3	4	2	3	3
3	1	9	2	3	4	2	2	4	3
4	1	9	3	3	4	3	3	2	3
5	1	9	2	3	3	3	2	3	3
TA_B	1	9	2.6	2.8	3.2	3	4	2.4	3

Table 3. Comparison of Criteria with worst and best

Test Case	**Input**	**Test description**	**Output**
1	Services and their parameters are unselected.	No services and their parameters are selected.	Appropriate error message.
2	Services are unselected and parameters are selected.	If the user does not select the service and selects the parameters of that service.	Appropriate error message.
3	Services and their parameters are selected	If the user selects the services and their parameters	Display the best CSP for that service.

Table 4. Testing for services and their parameters

Fig 1. List of CSP's and their services

Here the sample list of cloud service providers and their services that we are using as data to apply in our algorithm. 10 cloud service providers - Joy net, Amazon, Go Grid, Rackspace, Google, Century Link, HP, Nephoscale, Windows Azure, and Nine Fold. The services provided by one cloud might not be available in other clouds.

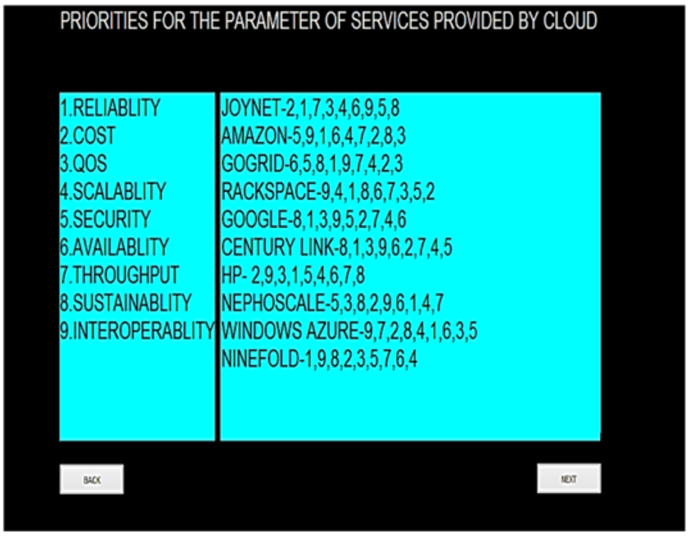


Fig. 2. Priorities for the parameter of services provided by the cloud

Here we are providing arbitrarily priorities for the different parameters of the services provided by the cloud. This is done so that we are able to match whatever parameters selected by the user with that of the priorities for the parameter provided by the cloud. This helps to match the parameters and suggest the CSP that user can opt for each service that he has selected. In this project we have taken a total of nine parameters for each service selected.

Fig 3. Warning message when parameters are selected without selecting service

If the user does not select the service and tries to select the parameters then appropriate warning message will be displayed saying that the user has to select the services in order to prioritize the parameters. Because the software will not run properly if the user intentionally does this. Therefore the user has to select the particular service in order to select their respective parameters in the row.

Fig 4. Services and parameters selected by the user

Here user selects various kinds of services that he needs and parameters for each service. There are two options for each service he can select or unselect the service from the pop-up menu. If the user unselects the service then he cannot select the parameters for that service. If by mistake he does that then the software will put a warning message saying that he should select the service to select their parameters. Each parameter is provided with options which user can select based on their requirements

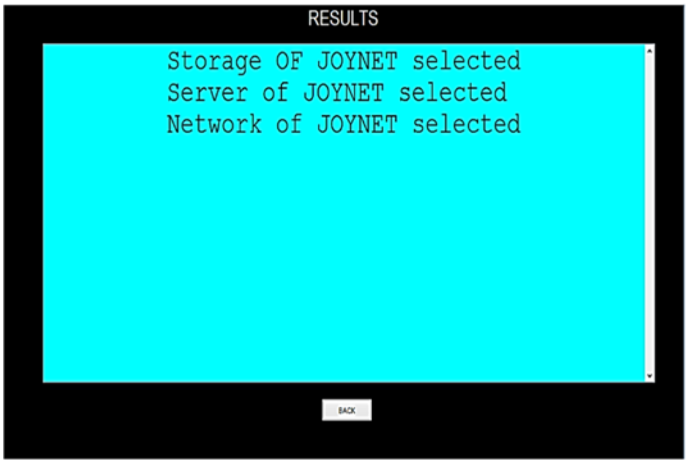


Fig 5. Best CSP selected for each service based on user requirements

Here the best CSP's for each service is displayed for the user. User can opt for the CSP if he wants. In the above example the user has selected three services that is storage, server, network and by applying NNA and based on the parameters selected for these services the algorithm tries to suggest the best CSP for each service.

5. Conclusion:

In this paper, we focus on the Cloud Service Selection by developing a three step process. By considering the parameters as the user requirements a robust system is developed to choose the service provided by different service providers. The proposed framework CSESF has four steps Requirement analysis, scrutinization, estimation and Selection using cosine metric KNN. The parameters for the services are, Reliability, cost, QoS, Scalability, Security, Availability, Performance, Sustainability, Interoperability. In each step, as per the requirements, whole process filters the selection of service provider. Scrutiniations eliminates the services that fulfill minimum criteria. Weights are calculated in the Estimation phase that helped to take a decision in the earlier stage. KNN with cosine metrics selects the cloud service from filtered providers.

6. Conflicts of Interest

The authors declare no conflict of interest.

References

[1] Garg, S. K., Versteeg, S., & Buyya, R. (2011, December). Smicloud: A framework for comparing and ranking cloud services. In 2011 Fourth IEEE International Conference on Utility and Cloud Computing (pp. 210-218). IEEE

[2] Godse, M., & Mulik, S. (2009, September). An approach for selecting software-as-a-service (SaaS) product. In 2009 IEEE International Conference on Cloud Computing (pp. 155-158). IEEE.

[3] He, B., Yang, M., Guo, Z., Chen, R., Lin, W., Su, B., ... & Zhou, L. (2009, May). Wave Computing in the Cloud. In HotOS.

[4] Kaur, H., & Kaur, M. (2015). KAMAN protocol for preventing virtual side channel attacks in cloud environment. TELKOMNIKA Indonesian Journal of Electrical Engineering, 15(1), 184-190.

[5] Delaram, J., Fatahi Valilai, O., Houshamand, M., & Ashtiani, F. (2021). A matching mechanism for public cloud manufacturing platforms using intuitionistic Fuzzy VIKOR and deferred acceptance algorithm. International Journal of Management Science and Engineering Management, 16(2), 107-122

[6] Yeoh, Eng-Juh, et al. "Classification, subtype discovery, and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling." Cancer cell 1.2 (2002): 133-143.

[7] Akerkar, R. A.; Lingras, P. (2008). *An Intelligent Web: Theory and Practice*, 1st edn. Johns and Bartlett, Boston.

[8] Lawrance, H., & Silas, S. (2013). Efficient Qos based resource scheduling using PAPRIKA method for cloud computing. International Journal of Engineering Science and Technology, 5(3), 638.

[9] Chang, V. (2013). Brain Segmentation? A Case study of Biomedical Cloud Computing for Education and Research. In Learning Technologies Workshop, Higher Education Academy (HEA), University of Greenwich.

[10] Chumpitaz, L., Furda, A., & Loke, S. (2019). Evolving variability requirements of IOT systems. In Software Engineering for Variability Intensive Systems (pp. 321-334). Auerbach Publications

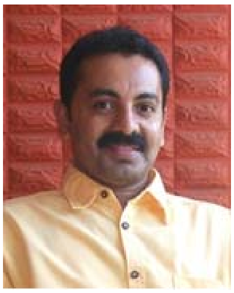
[11] Ahmed, F. D., & Majid, M. A. (2019). Towards agent-based petri net decision making modelling for cloud service composition: A literature survey. Journal of Network and Computer Applications, 130, 14-38.

[12] Jiao, L., Huo, L., Hu, C., & Tang, P. (2021). Refined UNet v3: Efficient end-to-end patch-wise network for cloud and shadow segmentation with multi-channel spectral features. Neural Networks, 143, 767-782.

[13] Khalaf, B. A., Mostafa, S. A., Mustapha, A., Ismaila, A., Mahmoud, M. A., Jubaira, M. A., & Hassan, M. H. (2019). A simulation study of syn flood attack in cloud computing environment. AUS journal, 26(1), 188-197.

[14] Kholidy, H. A. (2020). An intelligent swarm based prediction approach for predicting cloud computing user resource needs. Computer Communications, 151, 133-144.

Authors Profile



Sudheer Shetty is working as Associate Professor & Head in the Department of Information Science and Engineering at Alva’s Institute of Engineering & Technology, Moodbidri, Mangaluru, Karnataka. He has 21+ years of experience in teaching and 5+ years of experience in research domains. He obtained his Masters (MTech) from Visvesvaraya Technological University, Belagavi. He has 14 journal publications and 4 conference publications at national and international levels. He also has a best paper award into his credit. He received Toppe+Elite+Gold Certificate in Design and Analysis of Algorithms and Topper+Elite+Silver Certificate in Theory of Computation courses in the national level exams conducted by NPTEL. He is the life member of Indian Society for Technical Education (MISTE).

Manu A P, Professor in the Department of Computer Science and Engineering, having about 20+ years of academic and industrial experience. Awarded Doctoral degree from IIIT, Allahabad. India. Has authored two books titled “A to Z Linux” and “ABC of Hadoop, Docker and Ansible” for bridging the academic and industrial gap. He is a Fellow of the Institute of Engineers member of CSI and ISTE. His research scholars are working in Cloud computing, IoT, and Security. He is a reviewer of Elsevier, InderScience, Taylor, and Francis journals.

.

Pavan Kumar V, had completed M. Tech from Indian Institute of Information Technology, Allahabad. He is pursing PhD from Visvesvaraya Technological University, Belagavi. He has 11+ years of experience in Industry & Teaching and 6+ years of experience in Research. He has 14 publications in International Journal and Conference. His research interests include Cloud Computing, IoT.