DENSITY BASED CLUSTERING APPROACH FOR EFFICIENT WEBSERVICES DISCOVERY

P.Rajeswari

Assistant Professor, Department of Computer Science and Engineering,
Rajalakshmi Engineering College, Chennai, Tamilnadu, India
rajkavin2008@gmail.com

Dr.K.Jayashree

Professor, Department of Computer Science and Engineering,
Rajalakshmi Engineering College, Chennai, Tamilnadu, India
k.jayashri@gmail.com

Abstract

A huge number of Web Services accessible on the Internet has made service discovery a challenging task. Efficient way to minimise the search space is by organising Web Services into similar clusters. This paper presents an efficient Web Service Discovery approach based on Density Based (DB) clustering algorithm. Word frequency-based similarity calculation method in word score calculation is proposed. Clusters are formed using k-means and redefined using DB algorithm. The results obtained from DB is high accurate clusters of WSDL. The performance of the proposed DB approach is evaluated by comparing the accuracy of the service discovery with the existing results and found that DB outperforms than the others.

Keywords**:** Web based Services; XML; Clustering; Word frequency matrix; Service discovery.

1. Introduction

Web services are self-describing services designed by different vendors. They are software components that cater certain business functionalities can be used in various suitable applications dynamically. [9]. The components such as XML (eXtended Markup Language), SOAP (Simple Object Access Protocol), WSDL (Web Services Description Language), and UDDI(Universal Description and Discovery Integration) are highly preferred because they have more interoperability among various Web Services.XML describes the data in a meticulous way using custom defined tags for transmitting data and the information among various systems [10]. Web Service Management System is a comprehensive framework for managing web service lifecycle which covers the development, deployment, publishing, discovery, composition, monitoring and optimizing access to web services. As web services are being extensively deployed for business applications, the reliability of the services provided becomes an important criterion to enable the usage of such services [2]. Internet users are increasing day by day, network requirement also increases to obtain good performance. Therefore, many online services demand a very large bandwidth and network performance [3, 4].

WSDL specifies the services and the major components used to describe their services in terms of service types, messages, port-type, and binding. Service in WSDL is used to aggregate multiple port and functions [18].Service types are used in message exchanges to depict the data containers, messages are an abstruse representation of transmitted information, and port-type is used to define the functions in the webservice. Finally, binding is used to determine the communication protocol. SOAP is used to exchange the data among Web Services and it explains the systematic method of passing XML-encoded data [10]. UDDI lists the set of all feasible services, permits to define and register services in businesses [25].

Technology of Web Services is widely used in large number of recent applications like Stock trading, Supplying on Demand, Online reservation, On-line banking, Healthcare and Weather Forecasting [22]. The author [12] discussed a detailed survey about various methods, approaches and techniques used to cluster webservices and explore the efficient approaches among them in terms of service discovery. Web Services are becoming more and more eminent to perceive the user requirement. Large enterprises entrusted the webservices (WS) as a technique for software development in large scale [23]. A single web service contains various characters are secretive and are challenging to researchers [15]. Multiple Services are combined into a composite service, when a desired requirement is not met by a single service. The increasing amount of Web Services has made the web developer to perform on service discovery, selection and recommendation. [24].

Exploring for efficient and effective Web Service discovery is one of the main challenges for researchers. Researchers are paying more attention on clustering the web services nowadays. Clustering is considered as the most important non-supervised learning technique that can regulate the similar Web Services in a cluster provide high matches with user's request. Similar services are grouped using clustering techniques based on their functions in order to competently enhance the service discovery process by changing the search space. This approach affords better performance in terms of service discovery efficiency. A clustering algorithm called K-mean clustering deployed in the framework. The algorithm iterates these steps until all centre points remain unchanged. The capability of a web service search engine to fetch the closely related service would enhance by area defined clustering of web services [14].

From the above discussion it is identified that, the number of webservices are increasing rapidly. Most of the webservices are performing similar task. Due to the scalability, more similarity, and enormous requests, it is essential to discover and choose the best, available and efficient webservices speedily the most relevant webservice can improve the efficacy of the web applications. In order to improve the efficacy, this paper used DB algorithm for clustering and adopting a highly suitable webservice according to the user request.In this paper, the entire work is based on DB clustering. Initially it is aimed to design and implement a novel framework for providing effective response to the user in terms of webservice discovery. This framework is particularly designed for supporting WS-consumers in their discovery of web services. The entire contribution of the paper is:

- Framework for enhancing the discovery of webservices.
- Word frequency-based similarity calculation method is used.
- Clusters are formed using k-means and are redefined using the DB algorithm.

2. Related Works

To understand the various issues and challenges faced by the earlier approaches and methods. Also, to determine the research problem, this section carry out a detailed literature survey on various methods, techniques and approaches have been focused on web service discovery. Successful execution of applications depends on the selection of most suitable Web Services in the Internet because of the fast development of Service providers. Various methods for service discovery have been suggested by researchers to group the web services. Clustering is a competent approach to reforming discovery performance that can highly curtail the search space for service discovery. [1] proposed the methods for combining the functional and non-functional methods. Selecting and matching processes are used as key parts in Adaptive Resonance Theory (ART) algorithm. It has the proficiency to handle extremely large and big dimensional datasets. Swarm based algorithm is used for clustering data and it is one of the biologically inspired algorithms. They used flocking of Birds approach in non-functional characteristics. There is no clear information about feature extraction and score calculation. [5] proposed Cat Swarm Optimization (CSO) algorithm. The numbers of cats that are required for each iteration are initialized in Cat Swarm Optimization. Dimensions, velocities, fitness, flag are the parameters required for algorithm calculation. Algorithm proceeded till best clustering obtained and it was applied to Term Frequency-Inverse Document Frequency score.

[8] used Description similarity Calculation method for clustering the services. Collaborative filtering algorithm is applied within a cluster that has a target service. Stem words are perceived from database by description of services. Description words are systematic before usage by assuming that they are semantically similar. Description similarity calculation of larger values gives more similar services. They proposed Agglomerative hierarchical clustering algorithm and nearest neighbour algorithm that cluster the services according to their description similarity. Neighbours selected for targeted services from Enhanced Rating Similarity. Neighbour for target service is considered by those services with greater enhanced rating similarity than threshold. Predicted Rating Computation benefits the users to find the relevant services in shorter time.

[11] proposed Maximum Entropy classification algorithm to classify Web Services into functional similar group. Membrane SOA model is used by them for parsing WSDL documents. Lemmatization and stemming methods are used for comparison. Most uniform models are selected by maximum entropy that satisfy any given constraints. Feature measure is used as a constraint for model distribution. Calculation of parameters for maximum entropy classifier used Hill climbing algorithm. Word count is used as an attribute for Web Service classification. Random variations in data are misclassified as important patterns in maximum entropy. [17] stated that Similarity of Web Services is calculated and data is used to cluster Web Services. K-means clustering algorithm is used by authors. Web Services are categorized depends on the user's request. Parsing, Tag removal, stemming, function word removal, content word recognition is used for vector construction. Cluster similar Web Services depending on the WSDL content, name of Web Service is calculated using K-means algorithm.

[21] used Hybrid Web Service tag recommendation strategy that is employed by means of tag mining, tag co-occurrence. Features are obtained from a Web Service's WSDL document and tags are applied to form a

cluster. Global similarity is the integration of feature-level similarities and tag-level similarities. Service clustering used global similarity. Metrics such as precision and recall are used to assess the performance. Scalability of data tagging in Web Service is an issue in this approach. [19] Categorization of services and service categories are proposed by authors for the implementation, using the clustering technology. Preprocessing, rough clustering and fine clustering are the steps involved in service categorization. Rough clustering is implemented using string matching mechanism. Bisect K-means algorithm is used for clustering and effectiveness of classification is validated using clustering method and similarity measures. Co-occurrence of different functions is not handled in this method. [13] have used WordNet based similarity calculation. Service description document is analyzed and obtain the functionalities of service clustering, and form the documentation for service clustering feature based on service input, output name and service comments. Suffix TRee based INcremental service clustering (STRING) method is used to processthe original clustering feature. Segmentation, Abbreviation expansion, remove stop words are the processing steps in STRING. Homographs are identified using WordNet. Service library contains all clustering features of services enact a suffix tree. Semantics of arrangement of words in service clustering feature scrutinize incremental suffix clustering. Clustering accuracy is provided by Semantic extension for clustering services. STRING was designed to calculate only similarity based on WordNet.

Web Services are ranked in different clusters using Top-k query algorithm. Relationship between services and related types of objects such as requesters and providers are mined using heterogeneous network [28]. Service ranking prior probability model and posterior probability model are used to attain the ranking and clustering for Web Service network. Similarities are calculated between the services and cluster centroids using partitional clustering algorithm. Manual input of cluster number is an issue. Clustering and ranking is not efficient when service number is larger in this method [20]. [26]Web Service from different levels is illustrated by atomic service, abstract service and cluster service, and Web services are maintained in order using clustering-based service organization framework. Weighted hypergraph with web services are produced using Hypergraph-based service clustering method, and then hypergraph partitioning algorithm depend on the hierarchical service organization framework produced cluster of Web Services.

[27] Rendering prior knowledge, a novel Web service clustering method is used to strengthen the process of clustering in a semi-supervised way. Web services Mashuped together are greatly destined to form various groups of clusters and within the same cluster web services that are commented with similar tags are determined with an observation in the dataset of web services. Probabilistic topic model to extort the latent topic vectors from Web service description documents approach is used. Prior knowledge is used to perform clustering in K-means++ algorithm. Earlier work has primarily focused on clustering Web Services using syntactic similarity measures. In the proposed approach, word frequency is calculated for every WSDL document. Frequency matrix is formed with terms and WSDL documents with their frequency value. Density based clustering algorithm is used for clustering WSDL.

Initially WSDL files are collected from various service registries using Internet. Collected WSDL is clustered into functional similar groups based on similarity between the WSDL documents. Each WSDL file represents corresponding Web Service which performs certain task and consists of certain functionality. WSDL consists of enormous amount of information generally in the form of tags and data like XML. Tags are removed, only functional terms are obtained in the parsing and the functional terms obtained from the WSDL parsing has to be refined into qualifying English words by applying NLP techniques such as Tokenization, Stemming, Stop word removal. Preprocessing is a technique that involves revamping raw data into a logical format such as refining WSDL elements. The process of breaking a sentence or line into words, symbols, phrases or other meaningful elements called tokens is known as Tokenization. The system will split GetAccountDetails to three separate words: get, account, and details, using capital letters as breaking point. Stemming is defined as reducing inflected words to their word stem. It is a root form usually a written word form. The word starting, started, starts are converted to start. There is no single common list of stop words used by all NLP tools and the system removes all the stop words from the WSDL terms.

3. Density Based Clustering

Service Provider, Service Consumer and a Service broker are the main constituents of Web Service architecture [8].Service Provider is the platform that hosts access to the services and publishes their service descriptions in a searchable registry known as service registry. Trigger an interaction with a service, queries the service registry for the type of service required or fetch the service description directly are the functions of Service requestor [17]. Determining the suitable Service for users is becoming considerably crucial with the rapid growth of the number of registered Services, [12] and service discovery has a vital role in web-based application. Web service clustering became an effective research topic recently and it is extensively applied in research and industry area [16]. Density based Clustering architecture is proposed to cluster Web Services and it is depicted in Fig.1. Clustering is the task of grouping a set of objects in such a way that objects in the same cluster are more similar to each other than to those in other clusters [14].

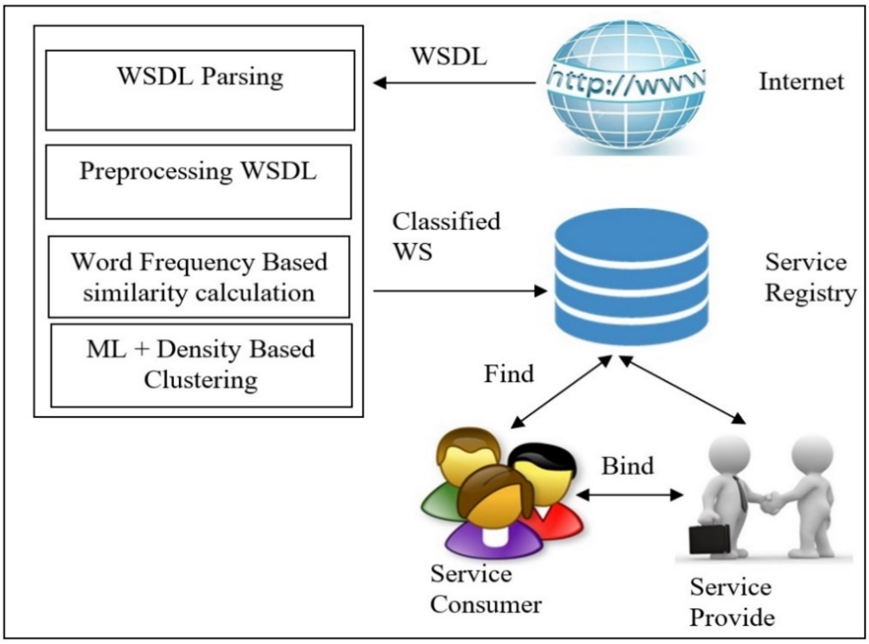


Fig. 1. Density based Clustering architecture

3.1 *Word frequency calculation*

Word frequency is calculated for each and every WSDL document and it is a sum of the number of occurrences of individual terms or qualifying tokens from previous step. Numerical count of every word is calculated for every WSDL document is shown in (1).

$$\text{Word Frequency} = \sum \text{Number of the occurrence individual terms.} \qquad (1)$$

Consider GetFinanceinterest is a term which is extracted from previous steps and this term is occurred 8 times in document 1 and 10 times in document 2. Word frequency for this term is 8 and 10 respectively. After calculating word frequency for every document, frequency matrix is formed with terms and WSDL documents with their frequency value is shown in (2).

$$A = \begin{bmatrix} a_{ij} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{m1} & \cdots & a_{mn} \end{bmatrix} = (a_{ij}) \in R^{mn} \qquad (2)$$

The individual items in an m × n matrix A, often depicted by $a_{i,j}$, where max $i = m$ and max $j = n$, are called its Word frequency value . Row size m is resolved by number of files in the dataset and Column size nis determined by total number of words present in the document. Total number of documents is huge however word frequency calculation is achieved by using nested hash map. Nested hash map is hash map of another hash map. Generally, hash map is a collection of key value pair. Word and its count act as key value pair of hash map and another is document name and word hash map pair. By iterating over every document and its words, the key value pair is updated. Finally, the word frequency value from hash maps is transferred to Frequency matrix.

3.2 *Cluster Formation*

Objects related to each other are grouped to form a Cluster. From the calculation of word frequency matrix, it is taken as dataset for clustering. K-means clustering algorithm is used for clustering WSDL, and intents to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean, delivering as a model of the cluster. This results in a segregation of the data space into Voronoi cells. K-means with word frequency method had given the initial clusters. Initial clusters are given as Input for Density based clustering algorithm.

4. Implementation and Results

The proposed system is evaluated in this section. The development platform was eclipse neon for windows and sublime editor. Additionally, WEKA java API and SOA Membrane parser is used. We have taken dataset as collection of WSDL documents, which consists of 1006 web services and in different categories. For evaluation of the clustering method 695 web services have been taken. Parsed WSDL content is analyzed using Apache UIMA package to transform contents to unigrams of terms. Word frequency method is applied to calculate the score and the clustering method is applied on the frequency score matrix to cluster the web services. Experimental setup of UDDI service repository is created to evaluate publish web service and discover web service using clustered WSDL document to prove the efficiency of word frequency method of clustering web services. UDDI setup is created with JUDDI v 3.0.2 and implemented in MYSQL using PHP.

The effectiveness of the proposed web service clustering method is examined first. Then the performance of the proposed method is demonstrated by comparing the results to other relevant research studies. The experiment was performed with 60 initial WSDLs documents. The results are deliberated with standard mining values such as precision and recall and its values have been determined using (3) and (4).

$$precision = \frac{succ(ci)}{succ(ci)+mispl(ci)} \quad (3)$$

$$recall = \frac{succ(ci)}{succ(ci)+missed(ci)} \quad (4)$$

where ***ci*** *connote*s the cluster *i*,

***succ*(*ci*)** is the number of web services successfully located in the cluster *ci*

***mispl*(*ci*)** is the number of web services inappropriately clustered in *ci*

***missed*(*ci*)** represents the number of web services that should be clustered in *ci* but are misplaced in other clusters.

The number of elements that are correctly retrieved out of all elements in a single cluster is called precision. For example, if the credit card cluster retrieved total of 10 WSDLs and out of which only 7 WSDLs are credit card-related services means the precision value would be 70%. Overall accuracy of retrieved elements is recall. For example, if there are total of 7 WSDLs related to the credit card services in the entire data set and those 7 WSDLs are correctly retrieved in the single cluster regardless if any other additional services are also retrieved, then the recall value would be 100% because the framework was able to correctly retrieve all credit card WSDLs. 50 initial WSDLs are collected from the five different categories; Credit card, Address validation, Currency exchange, Email verification and Weather services.

The system holds default weight scale by putting 0.6 for web service name, 0.3 for the word similarity and 0.1 for the structure similarity. Those values were set to support the hypothesis of project stating that the similarity of word 19 representation including Web service name from WSDL is more worthwhile than the structural representation.

4.1 *K-means Algorithm with Word Frequency Method*

K-means with word frequency method had given the initial clusters. Initial clusters are given as Input for Density based clustering algorithm. The clustering result is shown in Table.1. Comparison of feature extraction time under different service numbers is carried out. This experiment is used to count the time of extracting feature towards the different service numbers as shown in Table.2.

Cluster	Precision%	Recall%
Currency exchange	98	98
Weather	99	99
Address validation	97	98
Credit card services	98	100
Bioinformatics	99	100

Table 1. Clustering Results

Service.No.	100	200	300	400	500	600	700	800	900	1000
Clustering time	32	61	98	115	156	170	210	240	274	321

Table 2. Clustering Results with service numbers

The extracting time of feature model is varying for different service numbers. As the service number increases, the execution time also increased. The results obtained are shown in Fig.2. In addition, service clustering accuracy is more than 90%. It can cluster more services and this shows the efficiency of the proposed system. Clustering accuracy has been calculated with different number of services is shown in Fig.3.

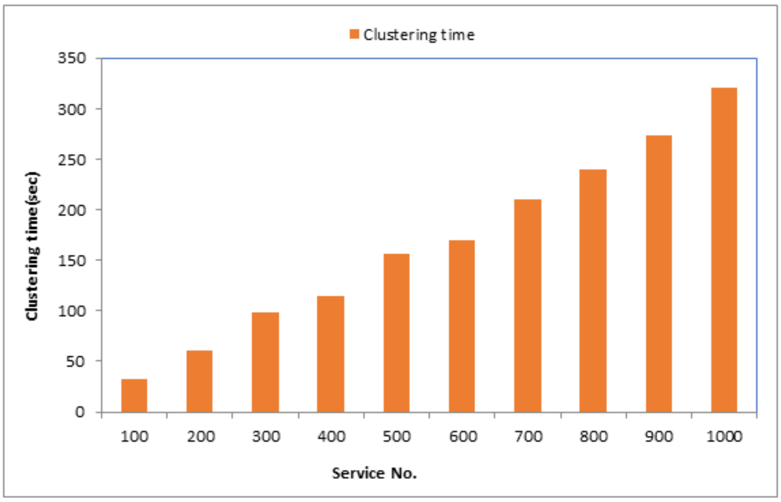


Fig. 2. Clustering Time

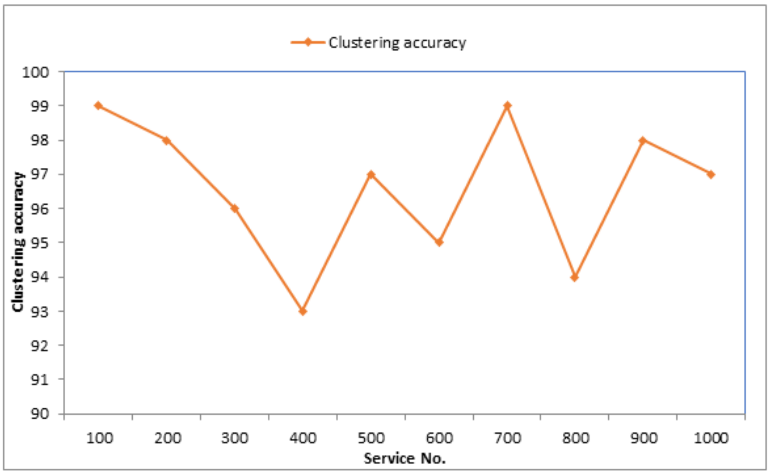


Fig. 3. Clustering Accuracy

Proposed Word frequency-based clustering method has been compared with CSO method proposed in [9]. Evaluation results have been shown in Fig.4.

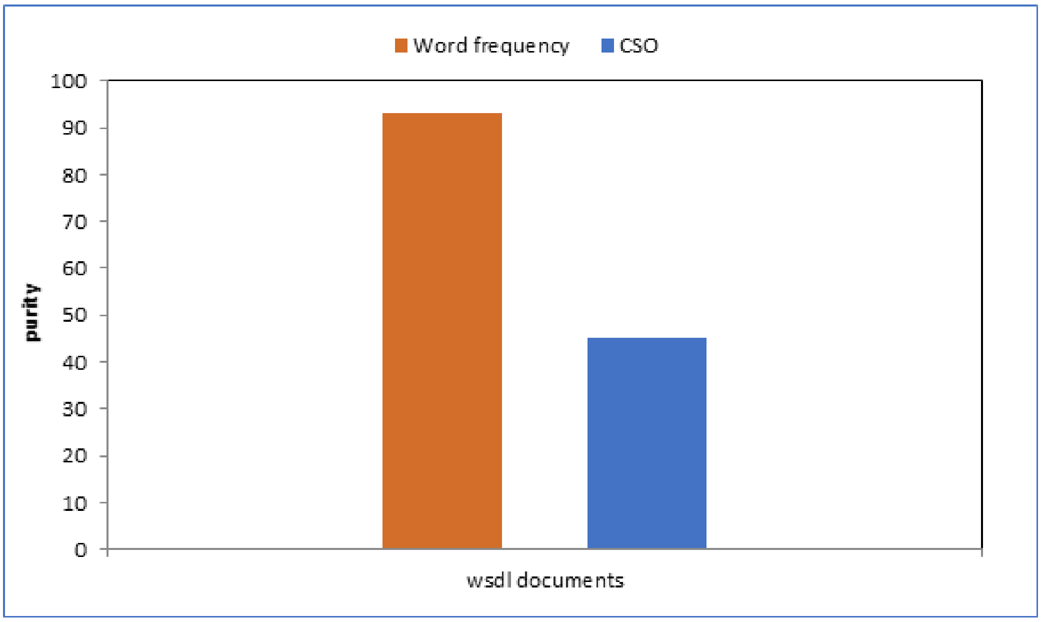


Fig. 4. Comparison of Word frequency based Clustering method vs. CSO Method

From the Fig.4, it has been proved that proposed word frequency method is more efficient than CSO method.

4.2 *Density based Clustering*

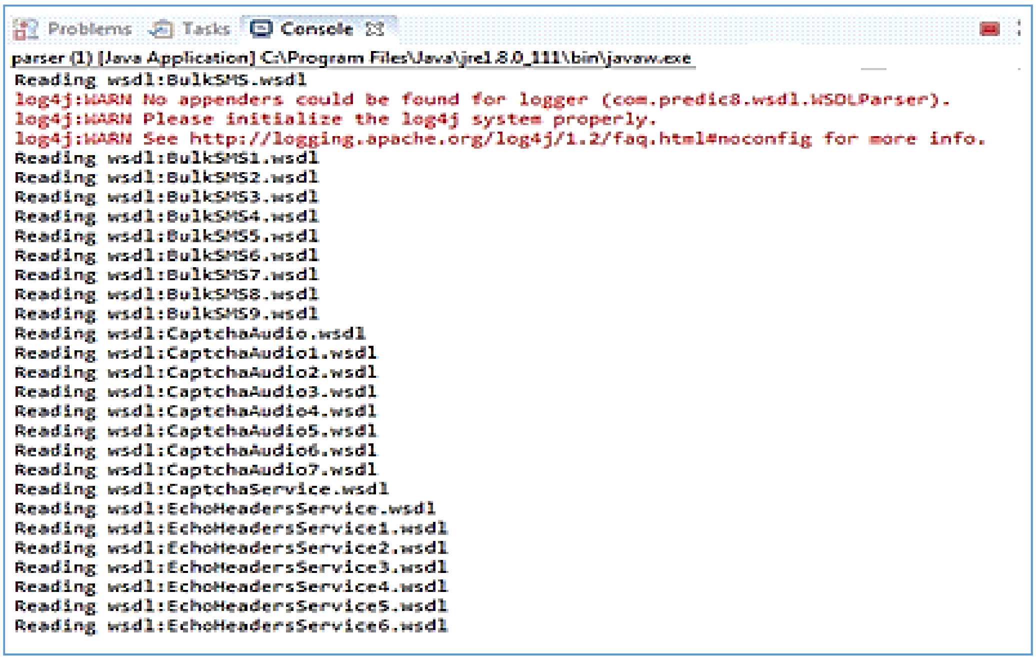
Clusters are redefined using the Density Based algorithm. 80 initial WSDLs are gathered from five different categories such as Finance, Business, Travel, Education and Healthcare as shown in Table.3.

Cluster	Precision%	Recall%
Finance	**95**	**96**
Business	**97**	**100**
Travel	**98**	**97**
Education	**94**	**99**
Healthcare	**93**	**100**

Table 3. Clustering Results

4.2.1 Experimental Results

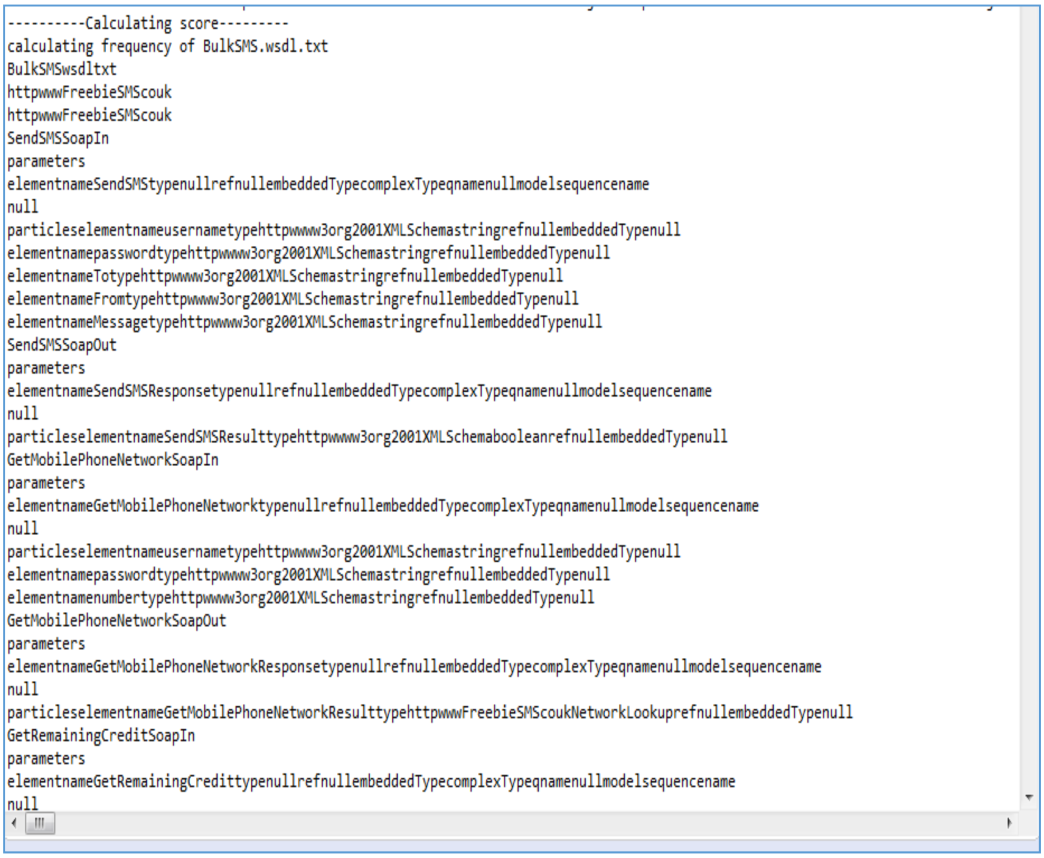
Raw WSDL file has been taken as dataset. When the program is started to execute the WSDL file and it is read one by one and the information is parsed from the WSDL. Parsed WSDL file data is reserved as same file name in the raw data folder. The parsing results of WSDL file are depicted in Fig.5.



```
parser (1) [Java Application] C:\Program Files\Java\jre1.8.0_111\bin\javaw.exe
Reading wsdl:BulkSMS.wsdl
log4j:WARN No appenders could be found for logger (com.predic8.wsdl.WSDLParser).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
Reading wsdl:BulkSMS1.wsdl
Reading wsdl:BulkSMS2.wsdl
Reading wsdl:BulkSMS3.wsdl
Reading wsdl:BulkSMS4.wsdl
Reading wsdl:BulkSMS5.wsdl
Reading wsdl:BulkSMS6.wsdl
Reading wsdl:BulkSMS7.wsdl
Reading wsdl:BulkSMS8.wsdl
Reading wsdl:BulkSMS9.wsdl
Reading wsdl:CaptchaAudio.wsdl
Reading wsdl:CaptchaAudio1.wsdl
Reading wsdl:CaptchaAudio2.wsdl
Reading wsdl:CaptchaAudio3.wsdl
Reading wsdl:CaptchaAudio4.wsdl
Reading wsdl:CaptchaAudio5.wsdl
Reading wsdl:CaptchaAudio6.wsdl
Reading wsdl:CaptchaAudio7.wsdl
Reading wsdl:CaptchaService.wsdl
Reading wsdl:EchoHeadersService.wsdl
Reading wsdl:EchoHeadersService1.wsdl
Reading wsdl:EchoHeadersService2.wsdl
Reading wsdl:EchoHeadersService3.wsdl
Reading wsdl:EchoHeadersService4.wsdl
Reading wsdl:EchoHeadersService5.wsdl
Reading wsdl:EchoHeadersService6.wsdl
```

Fig. 5. Parsing

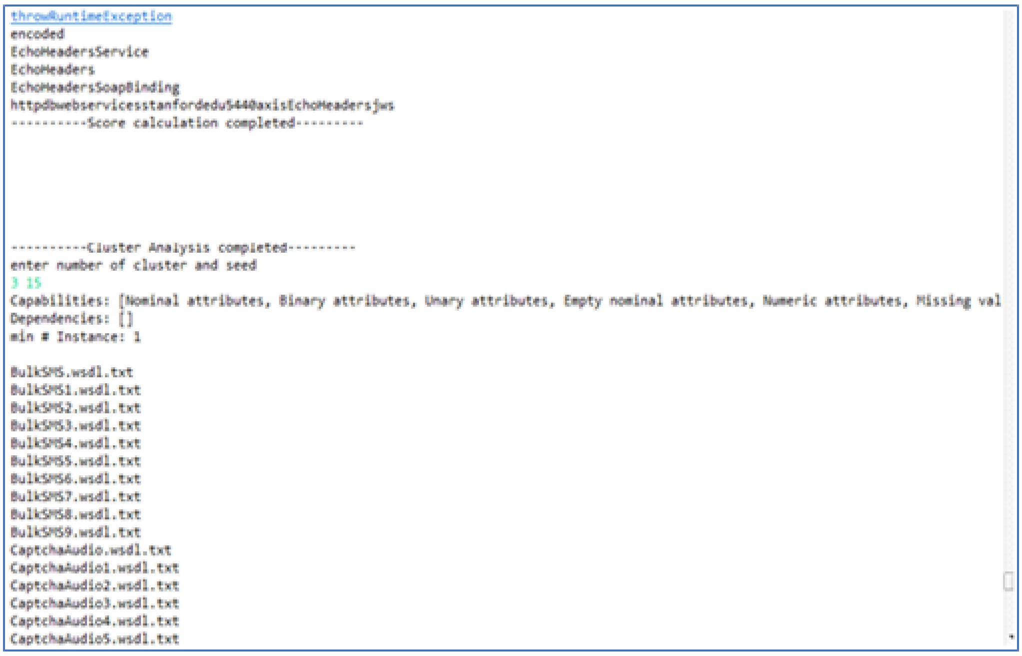
After parsing, the raw data is analyzed using apache UIMA and stored in the data folder then the word frequency had been calculated in the files from data folder in the score calculation part. Score calculation part had been run automatically after completing parsing and analyzing. Score calculation results are shown in Fig.6.



```
----------Calculating score---------
calculating frequency of BulkSMS.wsdl.txt
BulkSMSwsdltxt
httpwwwFreebieSMScouk
httpwwwFreebieSMScouk
SendSMSSoapIn
parameters
elementnameSendSMStypenullrefnullembeddedTypecomplexTypeqnamenullmodelsequencename
null
particleselementnameusernametypehttpwwww3org2001XMLSchemastringrefnullembeddedTypenull
elementnamepasswordtypehttpwwww3org2001XMLSchemastringrefnullembeddedTypenull
elementnameTotypehttpwwww3org2001XMLSchemastringrefnullembeddedTypenull
elementnameFromtypehttpwwww3org2001XMLSchemastringrefnullembeddedTypenull
elementnameMessagetypehttpwwww3org2001XMLSchemastringrefnullembeddedTypenull
SendSMSSoapOut
parameters
elementnameSendSMSResponsetypenullrefnullembeddedTypecomplexTypeqnamenullmodelsequencename
null
particleselementnameSendSMSResulttypehttpwwww3org2001XMLSchemabooleanrefnullembeddedTypenull
GetMobilePhoneNetworkSoapIn
parameters
elementnameGetMobilePhoneNetworktypenullrefnullembeddedTypecomplexTypeqnamenullmodelsequencename
null
particleselementnameusernametypehttpwwww3org2001XMLSchemastringrefnullembeddedTypenull
elementnamepasswordtypehttpwwww3org2001XMLSchemastringrefnullembeddedTypenull
elementnamenumbertypehttpwwww3org2001XMLSchemastringrefnullembeddedTypenull
GetMobilePhoneNetworkSoapOut
parameters
elementnameGetMobilePhoneNetworkResponsetypenullrefnullembeddedTypecomplexTypeqnamenullmodelsequencename
null
particleselementnameGetMobilePhoneNetworkResulttypehttpwwwFreebieSMScoukNetworkLookuprefnullembeddedTypenull
GetRemainingCreditSoapIn
parameters
elementnameGetRemainingCredittypenullrefnullembeddedTypecomplexTypeqnamenullmodelsequencename
null
```

Fig. 6. Score Calculation

After calculating the score, clustering part had been executed by taking number of clusters and seed as input. Cluster results are mapped to folders by matching it with corresponding file name. Fig.7. shows the clustering results.



```
throwRuntimeException
encoded
EchoHeadersService
EchoHeaders
EchoHeadersSoapBinding
httpdbwebservicesstanfordedu5440axisEchoHeadersjws
----------Score calculation completed---------




----------Cluster Analysis completed---------
enter number of cluster and seed
3 15
Capabilities: [Nominal attributes, Binary attributes, Unary attributes, Empty nominal attributes, Numeric attributes, Missing val
Dependencies: []
min # Instance: 1

BulkSMS.wsdl.txt
BulkSMS1.wsdl.txt
BulkSMS2.wsdl.txt
BulkSMS3.wsdl.txt
BulkSMS4.wsdl.txt
BulkSMS5.wsdl.txt
BulkSMS6.wsdl.txt
BulkSMS7.wsdl.txt
BulkSMS8.wsdl.txt
BulkSMS9.wsdl.txt
CaptchaAudio.wsdl.txt
CaptchaAudio1.wsdl.txt
CaptchaAudio2.wsdl.txt
CaptchaAudio3.wsdl.txt
CaptchaAudio4.wsdl.txt
CaptchaAudio5.wsdl.txt
```

Fig.7. Clustering Results

4.2.2 Comparison and Discussion

Purity of cluster formation of Density based clustering with K-means algorithm by using different dataset and also in WSDL dataset is compared. Resultant Table.4 shows the efficiency of Density based Cluster when compared with K-means algorithm. Purity of density-based clustering algorithm is closer to K-means algorithm but in case of Web Service clustering the cluster formation in Density based Algorithm is better than the K-means when usage of Word frequency similarity calculation method and for larger data set. From results, Cluster formation of density-based algorithm is similar to K-means when dataset is smaller. When dataset is large density-based clustering forms better cluster, because density-based clustering algorithm works very well when cluster density become large. Cluster formation purity graph for K-means and density-based algorithm is shown in Fig.8.

Dataset Name	No. of. Documents	Attributes	Clusters	K-Means Purity	DB Clustering
Diabetes	798	9	5	70	72
Glass	214	10	6	54	58
Iris	150	5	3	67	90
Labour	57	17	3	78	80
Soybean	683	36	4	79	83
WSDL	695	6755	10	91	95

Table 4. Purity of Cluster Formation for Different Datasets

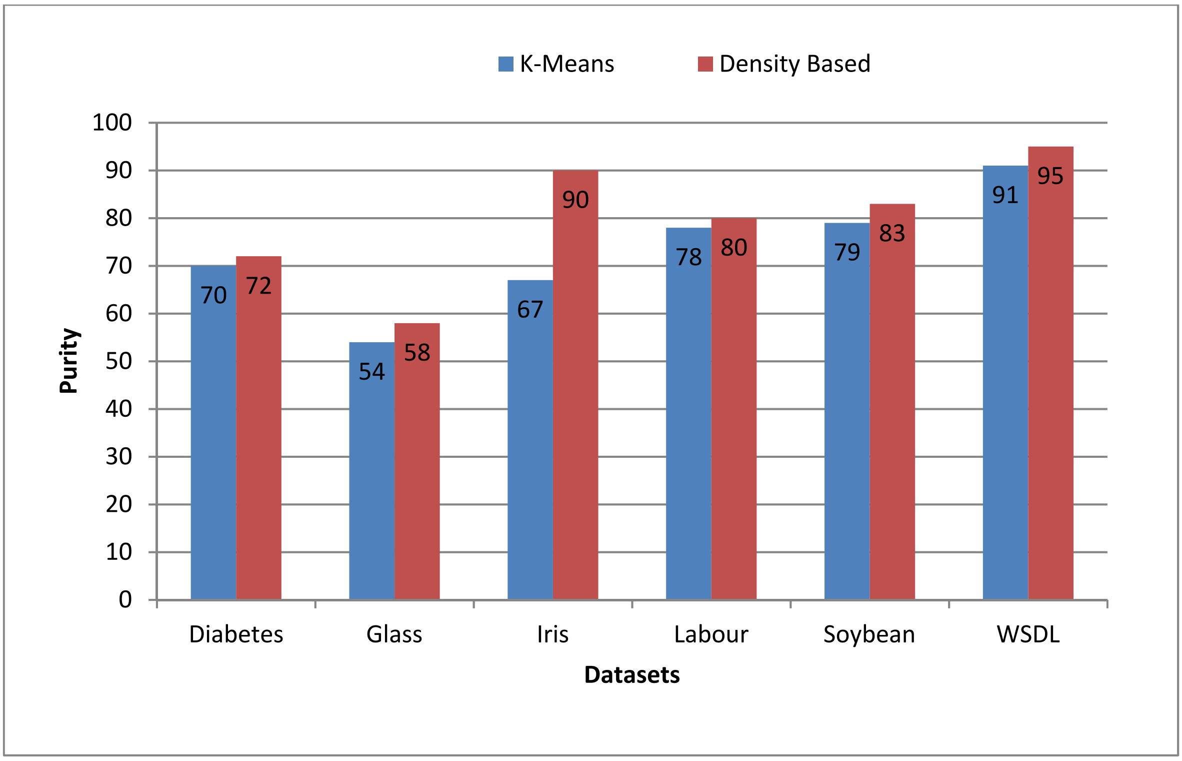


Fig. 8. Performance Comparison in terms of Purity

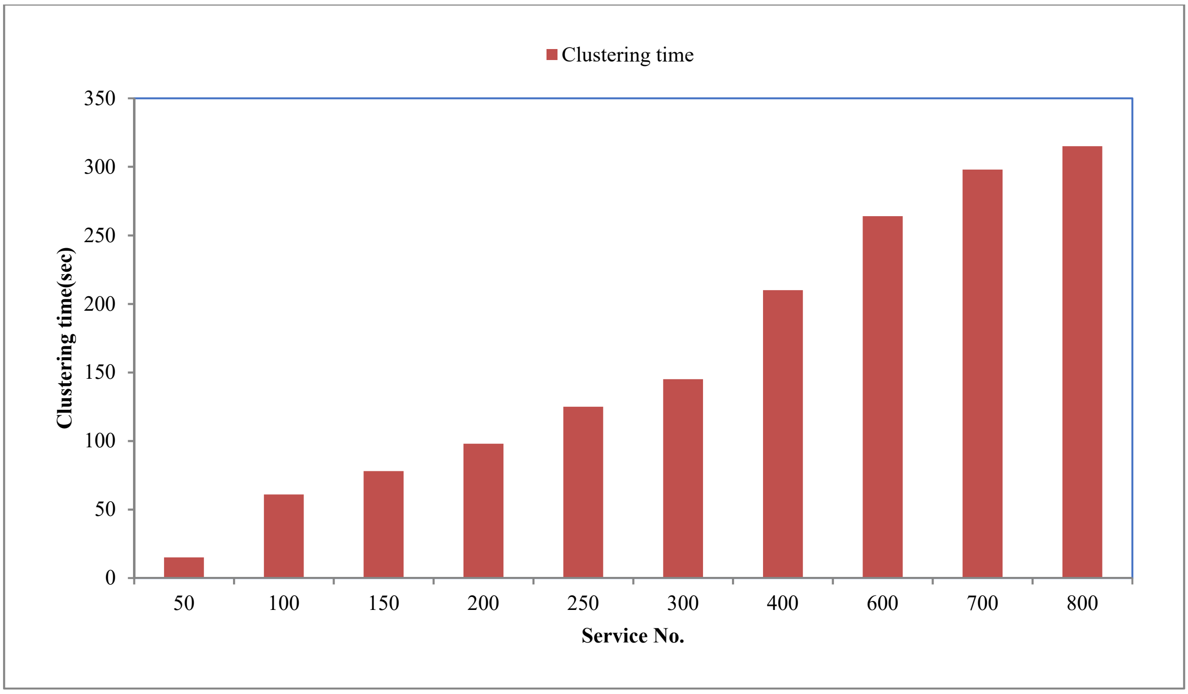


Fig. 9. Clustering Time

Service. No.	**50**	**100**	**150**	**200**	**250**	**300**	**400**	**600**	**700**	**800**
clustering time	15	61	78	98	125	145	210	264	298	315

Table 5. Clustering Time

Service. No.	**50**	**100**	**150**	**200**	**250**	**300**	**400**	**600**	**700**	**800**
clustering time	96	96	95	96	96	97	97	97	97	96

Table 6. Clustering Accuracy

4.2.3 More comparisons

Comparison of feature extraction time under different service numbers is dragged out. Count time of extracting feature towards the different service numbers is experimented shown in Table.5. The extracting time of feature model is varied for different service numbers. As the service number increases, the execution time is also increased. Results are shown in Fig.9.In addition, service clustering accuracy is more than 90%.It can cluster more services and this shows the efficiency of the proposed system. Clustering accuracy had been calculated with different number of services is given in Table.6.

5. Conclusion and Future Work

This paper presents an efficient Web Service Discovery approach Density Based (DB) clustering algorithm. Word frequency-based similarity calculation method in word score calculation is proposed. Clusters are formed using k-means and are redefined using the DB algorithm. The results obtained from DB is high accurate clusters of WSDL. The performance of the proposed DB approach is evaluated by comparing the accuracy of the service discovery with the existing results and found that DB outperforms than the others.

Determining the suitable Service for users is becoming considerably crucial along with rapid increase in the number of registered Services, the search involved in the space of a service discovery function is reduced using Clustering. Initially clusters are formed using k-means algorithm, and then the Density based clustering algorithm takes the clusters as input. Clusters are redefined using the Density Based algorithm, as the result more accurate clusters of WSDL have been formed. Comparing purity of cluster formation of Density based clustering with K-means algorithm by using different dataset and also in WSDL dataset. Cluster formation of density based algorithm is similar to K-means when dataset is smaller. When dataset is large density-based clustering forms better cluster, because density-based clustering algorithm works very well when cluster density become large. Experiments have been carried out and results were compared with existing methods and efficiency of proposed method has been proved.

As a future work, methods would be implemented to reduce the execution time of feature extraction and also combining the semantic similarity with the word frequency method for more accurate clustering.

Conflicts of interest

“The authors have no conflicts of interest to declare”

References

[1] Praveen Joe, I.R.; Varalakshmi, P. (2016). A two phase approach for efficient clustering of Web Services. Springer science, 165-169.

[2] Jayashree, K.; Sheila Anand. (2013). Web Service Diagnoser Model for managing faults in web services, Computer Standards and Interfaces,154-164.

[3] Rajeswari, P.; Jayashree, K. (2018). Survey on QoS metrics and ranking in cloud services. International Journal of Engineering and Technology, 7, 146-149.

[4] Jayashree, K.; Sheila Anand. (2015). Run Time Fault Detection system for Web service composition and Execution. Smart computing Review. 5(5), 469- 482.

[5] Kotekar, S.;Sowmya,S. (2016). Enhancing service discovery using cat swarm optimisation based web service clustering. Perspectives in Science (Elsevier), 715-717.

[6] Kumara, B.T.G.S. (2015). Improving Web Service Clustering through Ontology Learning and Context Awareness, (Thesis).

[7] Fensel,D.;Bussler,C. The Web Service Modeling Framework WSMF, http://www.w3.org/ 2001/01/WSWS/, 1-33.

[8] Dhanalakshmi,K,; Anitha,A.;Michael.;G.;Venkatesan,K.G.S.(2015). Recommendation System Based on Clustering and Collaborative Filtering. International Journal of Innovative Research in Computer and Communication Engineering, 3(3),2482-2488.

[9] Haas, H..; Brown, A. (2004). Web Services glossary. W3C Working Group Note.

[10] Mehdi, G. (2012). Investigating SOAP and XML technologies in Web Service. International Journal on Soft Computing, 3(4), 15-19.

[11] Nisa, R.; Qamar, U. (2015). A text mining based approach for web service classification. Information Systems and Business Management, 13(4,) 751-768.

[12] Nayak, R.; Lee, B. (2007). Web service discovery with additional semantics and clustering. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence. Washington, DC, USA: IEEE Computer Society, 555–558.

[13] Li, J.; Ma, D. (2013). Suffix Tree Based Incremental Web Services Clustering Method. Journal of networks, 8(6), 1566-1573.

[14] Zhang, X.; Yin, Y.; Zhang, M.; Zhang, B. (2009). Web service community discovery based on spectrum clustering. Computational Intelligence and Security, 187–191.

[15] Zhu, Z.; Yuan, H.; Song, L.; Bi, L.;Liul, G. (2010). WS-SCAN:An effective approach for Web Services clustering. Proc. International Conference on Computer Application and System Modeling, China, 618-622.

[16] Rupasingha, A. (2016). Domain-aware Web Service Clustering based on Ontology Generation by Text mining. IEEE Electronics and Mobile Communication Conference, 1-7.

[17] SanthanaVijayan, A.; Balasundaram, S.R. (2013).Effective Web service Discovery using K-means clustering. International conference on Distributed Computing and Internet Technology, 455-464.

[18] Elgazzar, K.; Hassan, A.;Martin, P. (2010). Clustering WSDL Documents to Bootstrap the Discovery of Web Services. IEEE International Conference on Web Services, 147-154.

[19] Liang, Q.; Li, P.; Hung, C.K.; Wu, X .(2009).Clustering Web Services for Automatic Categorization. IEEE International Conference on Services Computing, 380-387.

[20] Xiong, Q.; Dong, J.; Wen, J.; Li, P. (2015). A Novel Web Service Top-K Query Algorithm Based on Heterogeneous Service Network Analysis. Journal of Computational Information Systems, 11(1), 387-398.

[21] Wu, J.;Chen, J.; Zheng, J.; Lyu, R.;Wu, Z. (2013).Clustering Web Services to facilitate service discovery. Springer-Verlag London, 38(1),207-229.

[22] Tamilarasi, K.; Ramakrishnan, M. (2016). Semantic cluster based search in UDDI for Health Care Domain. Indian Journal of Science and Technology, 9(12), 1-5.

[23] Gottschalk, D.; Graham, S.; Kreger, H.; Snell, J. (2002). Introduction to Web Services architecture. IBM Systems Journal, 41(2), 170-177.

[24] Saha, S.; Murthy, C.A.; Pal, S.K. (2008). Classification of Web Services using tensor space model and rough ensemble classifier. Foundations of Intelligent Systems, 508-513.

[25] Mustapha, A.; Mohamed, Q.; Nicolas, D.; Zahi, J. (2013). Web Services Discovery and Recommendation Based on Information Extraction and Symbolic Reputation. International Journal on Web Service Computing (IJWSC), 4(1), 1-18.

[26] Lei, W.; Zhen, Y.; Shijun, L.; Minggang, H. (2018). An Approach to Web Service Organization Based on Hypergraph Clustering. IEEE 22nd International Conference on Computer Supported Cooperative Work in Design, 87-91.

[27] Min, S.; Jianxun, L.; Buqing, C.; Yiping, W.; Xiangping, Z. (2018). A Prior Knowledge based Approach to Improving Accuracy of Web Services Clustering. IEEE International Conference on Services Computing, 1-8.

[28] Babu, R.; Jayashree, K. (2021). A Generic Model for Identifying QoS Parameters Interrelations in Cloud Services Selection Ontology during Runtime. Symmtery, 13(4), 1-18.

Authors Profile

Mrs. P. Rajeswari is a Research Scholar in Anna University. She has completed her Masters in Computer Science and Engineering with a merit of 11th rank in Anna University and Bachelors in Information Technology from P.S.R Engineering College affiliated to Madurai Kamaraj University. She is presently Assistant Professor in the Department of Computer Science and Engineering at Rajalakshmi Engineering College, affiliated to Anna University Chennai. Her areas of interest include Web Services, Data Mining and Cloud Computing.

Dr. K.Jayashree is an Engineer by qualification, having done her Doctorate in the area of Web services Fault Management from Anna University, Chennai and Specialization in Embedded System Technologies from College of Engineering, Guindy - Anna University and Bachelors in Computer Science and Engineering from Madras University. She is presently Professor in the Department of Computer Science and Engineering at Rajalakshmi Engineering College, affiliated to Anna University Chennai. Her areas of interest include Web services, Cloud Computing, Data Mining and distributed computing.