International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2019 IJSRCSEIT | Volume 5 | Issue 4 | ISSN : 2456-3307
DOI : https://doi.org/10.32628/CSEIT195430

Reduction of Search Space in Restful Service Discovery

G. Venugopal[*1], Dr. P. Radhika Raju[2], Prof. A. Ananda Rao[3]

[*1] M.Tech Scholar, Department of Computer Science and Engineering, JNTUA College of Engineering, Ananthapuramu, Andhra Pradesh, India

[2]Ad-hoc Assistant Professor, Department of Computer Science and Engineering, JNTUA College of Engineering, Ananthapuramu, Andhra Pradesh, India

[3]Professor, Department of Computer Science and Engineering, JNTUA College of Engineering, Ananthapuramu, Andhra Pradesh, India

ABSTRACT

Web Services has been enabled IT services and computing technology to perform business services more efficiently and effectively. REpresentational State Transfer (REST) is to be used for creating Web APIs/services. In the existing system, web service search engines for RESTful Web Services/Api's provide Keyword, Tag and Semantic based search functions. One of the RESTful service discovery, referred as Test-oriented RESTful service discovery with Semantic Interface Compatibility (TASSIC) have been developed by the search of RESTful Service's/Api's. TASSIC approach will search the semantic characteristics of search and match interface terms in the service document. An inability to consider the classification and in finding the suitable Api's or services are a key issue of the search space in Tassic. A new approach has proposed for reduction of the search space in restful service discovery to develop a k-Nearest Neighbor classification algorithm. it provide candidate services with ranking based on semantic similarity, and classifying of similar candidate services and service unit testing will be considered. This approach is meant for increasing search precision in the retrieval and quick search for classifying their RESTful services or Api according to user-defined criteria.

Keywords : Web Service, Restful Service Discovery, Semantic search, Service Classification Algorithm

I. INTRODUCTION

Web Services (WS) describes a standardized way the flexible integration of independent systems and applications in open standards over a web. The Web based open standard protocols are using Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL), Universal Description Discovery and Integration (UDDI), Extensible Markup Language (XML), Remote Procedure Call (RPC), and Representational State Transfer (REST). The set of protocols, frameworks, specifications are use for developing, deploying a standard Web Services / Application Programmable Interferences (API's).

In earlier days, RESTful WS and Semantic WS have been major topics for research, practitioners in the area domain of distributed web computation, for developing a Search Engine of the RESTful Web Services/Api's. In The older search engines are RESTful Services/Api's providing only Keyword, Tag, and Semantics based search functions are used. To organize and semantics based search functions are some failure happens by the searching objects to take the services classification and matching to find

suitable web Services/API's fastly and easily. In existing approach is Test-Oriented RESTful Service Discovery with Semantics Interface Compatibility (TASSIC). TASSIC is not finding classifying and clustering of similar Service's /Api's.

Than the overcome, the failures of the existing approach by implementing a new approach service classification to the service discovery of restful service referred to as Service Classification and Clustering of RESTful Service Discovery approach that name as service classification approach. Automatically determining the category of a service, from several pre-defined categories, is an important problem with many large distributed applications such that service discovery, service matching and semantic annotation. RESTful service discovery and compositions with semantics web services and it describes Service Classification and Clustering algorithms.

In this paper presenting a service classification algorithm based on reduction of search space in RESTful service discovery. In this work, the mechanism is to have well searching a Restful services/Api's has to find a good semantics search in making the classifying and clustering of restful service discovery. The introduce Machine Learning Algorithms (MLA) as a better specifying alternative technique for classifying and matching the Restful Services/Api's. Classification algorithm is the K-Nearest Neighbours (KNN) algorithm that can be used to solve classification problems to predict the target Services/Api's, referred to a classify restful Services/Api's into different categories. Another algorithm is_K-means Clustering algorithm to a collection of services aggregating together because of certain similarities kind of services **in the same group should be similar to specified.** These algorithms are used to increasing efficiency, fast searching response of the services.

The remainder of the work is Section II, we discuss to the related literature on Restful WS, Semantics based WS discovery and Service classification; Section III presents the existing system; Section IV gives detailed discussing on how to realize semantic Web service classification and clustering ; Section V Finally, the presentation of experimental results and conclusion.

II. RELATED WORK

This section reviews of related works on the RESTful service discovery, semantics web service, service classification, clustering of services. Alex Rodriguez is proposed article best practice on RESTful web service is a set architecture of principals, verbs, system resources and representations of the design models representing, [5]. Reihaneh Rabbany Khorasgani, Eleni Stroulia, [6] is proposed WS matching for similar methods of their given restful service discovery on WSDL description. Vikram Singh and Balwinder Saini, [8] is define a Effective Tokenization process based on producing a respective tokens for the vector on Information Retrieval Systems. Ramalingam Sugumar & M.Rama Priya [9] is Enhanced Porter Stemmer Algorithm by the removing stop words; special characters form the retrieval of information. C. Fellbaum WordNet [10] it is a lexical database for the textual information, natural language processing (NLP) and computational linguistics. K.M. Wong, Wojciech Ziarko and Patrick C.N. Wong [11] is an proposed an systematic method for correlation of terms directly from automating indexed schema's that method is vector space model.

Lathem K. Gomadam A. P. Sheth [12] is presenting adding semantics to RESTful services by add's RDFa (annotations) and designing to create a semantically mashup's using semantic annotation- RESTful service. Martha Varguez-Moo, Francisco Moo-Mena [25] is propose, for classification of WS by using the Machine Learning Algorithm in WS discovery than matching their semantic information each WS and

quality of services parameters are satisfied. When using the Naive Bayes, Support Vector Machine, Adaboost algorithms are integrated. S. Sharma, J. S. Lather, and M. Dave [26] is proposed by supervised machine learning algorithms by KNN and support vector machine algorithms are improving the web service classification of semantics and also semantic relatedness measures. It implemented than the utilizing syntactic information as well as semantic with service description. The implements ML algorithm to improve the classification of the web services based on semantics.

Abdelmoniem Helmy, Mervat H. Geith, [27] is present, The introduced service clustering for grouping web services with similar functionalists and clustering is that the consumer question is compare with clusters by using matching techniques similar measuring. Then the clusters are select that have the highest similarity with consumer question and consumer request compared total WS present in the final selected clusters. Vijayan, A. S., & Balasundaram, [28] is introduced approach can be used search engines for efficient WSD based on the relevant consumer requests. The implements the K-Means clustering algorithm to similarity between service description and consumer request. Venkatachalam K, Dr.S.Balakrishnan, [29] is introduced the novel approach integrating the centroid classify algorithm for WSD. The intention of research is automatic data classifications that are using three algorithms there are KNN, SVM, and Centroid Classifier so improving accuracy results for the data classification. Chattopadhyay, and Ansuman Banerjee [30], in this research work is focus on the semantic driven QoS aware service composition for reduce the search space in semantic web service composition by the abstraction and refinement for the semantic-WS.

III. PROPOSED SYSTEM

Restful web services researches have been used the semantics services for some of different purposes, mainly importance for service composition, service discovery, and service execution. The main work focus on restful web service referred as semantic web service based on service classification aware from service discovery. Mostly the previous approaches on restful service discovery referred semantic related information to combine the services of generating by the semantics related information by the discovery solution for search engines adequately. However, it retrieves the information for services / Api's to expediting the service classification algorithm. Here, it presents service classification framework that uses restful discovery in semantic terms expansion of a restful web service to generate classify a services / Api's , form a set of relevant services in a cluster.

The proposed work describes the four levels of searching approach that is service document preparation, service semantic term expansion (or extraction), service classification, and service unit testing. Figure 1.

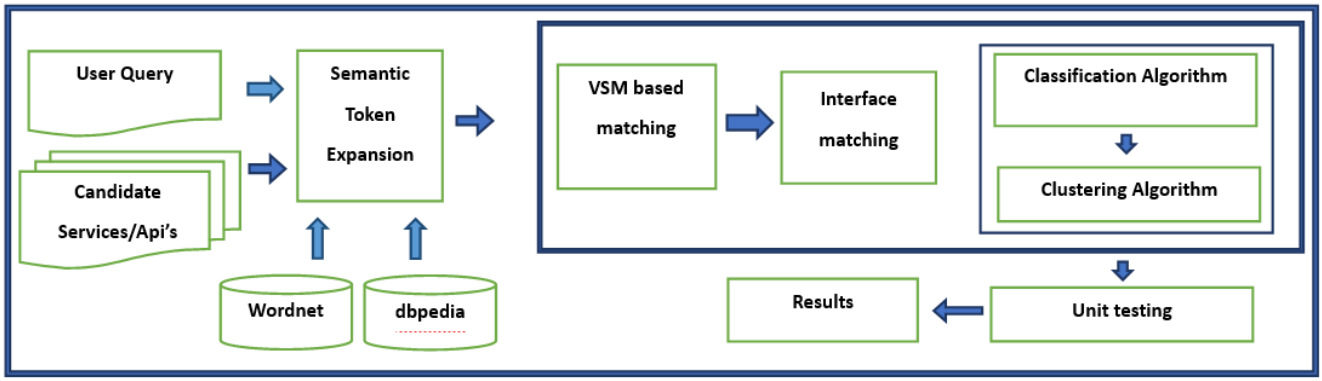

Figure 2. Propose architecture for service classification approach

A. Service Document Preparation

The service document preparation is a pre-processing stage in the task of preparation for service documents or query documents is index blinding. The index blinding following process is tokenization and stemming.

B. Tokenization

Tokenization process is performs service document and query information description into tokens and also eliminate stop words [4]. After tokenization process is collecting a tokens have to submit the stemming algorithm and same time storing results in repository.

C. Stemming Technique

Stemming process is performs more number of words are derivations from the same stem under the concept; it is find the different or similar word with the same root. For example, compute is root word and stemming words are computing, compute, computed, computer this are the stemming words with the root word. The fact stemming is generating through appended for prefixes, suffixes and /or infixes. The porter-stemming algorithm is using in this work. It possible to specifying all of the related and relevant words are matches their stem [5]. In the work obtaining, the stemming will applied on service document and query information.

D. Semantic term expansion

The semantic is changing the meaning of the word in the course environment. The semantic term expansion process of expansion for service document terms to semantic description. Semantic term expansion encountered between the service document terms and user terms included in service description of service matching. The service documents terms and user terms to add to additional set of new terms had better represent a term meaning. Wordnet is mostly use in either web service or restful service research as ontology for the semantic descriptions of services [3], [8]. In this work performs term expansion for service documents and description. In the service document is annotation as an ontology class, annotation terms is expanding based on Wordnet and dbpedia. A parameter of a resource to annotate by associated term is expanding by the Wordnet lexical db. This is semantic term expansion is utilized to restructure of a service documents in restful service discovery.

E. Wordnet Term Expansion

Wordnet is a lexical database sometimes is called a lexical ontology, obtain the synsets of each query document term are included Synonym; Hypernyms and Hyponyms by the lexical expansion, such as transformation and correction have been performed and documented. When the service document is extended the terms and forwarding to the next level. Calculating the term frequency (TF) for each term expanded term.

F. Vector Space Model Semantic Service Matching

The discovery the service by the information indexing, filtering, retrieval, relevant ranking of text or services description information one of best representing models is Vector space model (VSM) for search engine. In the VSM were performed returns all the similar kind of terms in the documents. The implementing works obtaining two levels of processing of term expansion and storing the results in the repository. Previous level of approach a service document its integrating tokenization and term stemming by the extracting terms in service document. The extracting terms is conducting for the service document, set the extended terms into semantic terms for each published Restful web

services. The computation of service document and description, services and query information was separate into three sets of terms that are description, input, and output. It was each term assigned weights; in the level, term is specifies to vector the result of the vector frequency is similarity with the original term. It was calculate similarity between semantic service vector and query vector in the recommended class.

$$\text{vsm}(\text{service}, \text{query}) = \frac{dv_s.dv_q + iv_s.iv_q + ov_s.ov_q}{\sqrt{|dv_s|^2 + |iv_s|^2 + |ov_s|^2} \cdot \sqrt{|dv_q|^2 + |iv_q|^2 + |ov_q|^2}} \quad (1)$$

Whereas
Service vector S = {(dv_s) vector of service description, (iv_s) vector of input, and (ov_s) vector of output} and Query vector Q = {(dv_q) vector of query description, (iv_s) vector of input, and (ov_s) vector of output}.

Each I/O of the corresponding to ontology based mapping to a unique entity term of the ontology. Parameter mapping is mappings of I/O elements are respectively the total set of I/O parameters and concepts corresponding to ontology. The preparing was adding the semantic term information into the generating semantic term service vector for the VSM to enriched semantically service vector

G. Service Classification

Service classification is the process of automatically classify a RESTful web services / Api's to one or more predefined classify based on its features of vector and similarity between semantics. Service classification helps in identifying, grouping, standard, alternative, related services class during the publication services. The input of service classification algorithm to integrating the semantics for the service vectors and transforming the services that is enriched semantically service vectors. In this work implementing service classification algorithms is KNN classification algorithm and K Means clustering algorithm this are the machine learning algorithm.

H. KNN Classification Algorithm

The K-Nearest Neighbours is the one of the data source in which the set services are pointing their separating into service categorization to predicting the classification the new sampling their services. This algorithm is one of best service categorization algorithm in machine learning. The determining the given user queries based on the service documents to categorize the service's/api's that are nearest to the k =1. The previous levels of the present work the Vector support model method is main instance of this method. This work integrating the vector support model, and Euclidean distance - weighted matching function this are accomplishing the categorizing the restful services, and the calculating Service's/Api's similarity like the query vector and service vector of distance-weighting. Then, it uses a matching score strategy to find the user defined query's class each retrieved service's / Api's contributes a match score for its class, weighted by its similarity to the user query. The query's possible classifications will be rank according to the match score. Algorithm: 1) Make semantic service vectors for each document's in the test set. 2) Than make one centroid service vector for each service. 3) Calculate similarity between each query vector and service vector. 4) Service's /API's belong to the service for which the similarity is the maximum of the service document.

$$\text{Euclidean distance} = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \quad (2)$$

I. K Means

The importance of the K-Means clustering algorithm finding group of services is one of the best technique in machine learning. The clustering is performing to group of service categories into different service groups, and the set of service in the same category

and it is more is more alike than those other categories. K means Clustering algorithm is a useful method to identify the outlines of their services in the variable k. They define the outer as an element of the service group that has few services and is located far from other groups. The K-means clustering algorithm is that given a specific parameter is K, it will partition some of the observations into K clusters, and each observation belongs to the cluster with nearest mean. The cluster is all users queries based on similarity services are cluster and other services are candidate services is undefined cluster. And calculate degrees of similarity calculate the term frequency and document frequency of the service documents by the query vector and service vector.

$$\text{Term frequency tf}(\text{term}, \text{document}) = \frac{n_t}{\sum_k n_k} \quad (3)$$

Document frequency =

$$\text{idf}(\text{term}, \text{Document}) = \log \frac{|D|}{|\{d_i \in D | t \in d_i\}|} \quad (4)$$

$$\text{tf} - \text{idf}(t, d, D) = \text{tf}(t, d) \,.\, \text{idf}(t, D) \quad (5)$$

This algorithm put all similar items into one cluster and keeps on iterating until all similar records arrange in appropriate cluster. The main aim of adding clustering is to enable users to know what the rest services are providing similar type of work without giving query. In existing work user has to give query to know list of services but this extension work will automatically put all related services into single cluster. By seeing clusters user can understand which are similar services and which work it is going to perform.

J. Unit testing

Unit testing is testing technique importance for software engineering. This technique is comparing the actual response and expected response the published RESTful Service's / Api's than confirm the availability and functionality of the candidate service document; the service description document includes the unit test case than the service response meets the user needs: declaration written by their service provider.

After implementing the service classification algorithms the classification set of published services vector into classify (or category) and clustering of their services having semantically similar services/Api's will belong to the same category and cluster. To semantically specifying in the classifying and clustering of their semantic service set vectors was the accurately among the available published RESTful services/Api's.

This approach integrating service classification algorithm reduces the search space of the Service's / Api's than automatically human intervention is reduced and choosing the servicers / Api's appropriate class for a services through RESTful service discovery process will improve. The RESTful service discovery of any relevant(or related) RESTful Service's / Api's, appropriate for the query of user client is required, services found and all the services present in the matched with the queries and discover the most relevant services/Api's specified.

IV. EXPERIMENTAL RESULTS

The implementing work have implemented a prototype system to verify or validate new approach, for comparison with the (TASSIC) actually last level approach implement improved the precision for searching services. Its prepared inputs to new service classification approach to Service document and query document before the experiment. In this approach submitted the service description document(SDD) without semantic mapping information and also 10 representative query documents(QD) include the expected services/api's titles, input parameters, and output parameters. An

important point is when the QD are specified for RESTful service with the only get method. Experiments evaluation for performance with individually entered 10 user query documents into new service classification approach and TASSIC and than searched for set of different scenarios services/api. The returned services/api's were ranked (or match score) according to their score values than calculated the accuracy of the values. Output of the approach retrieved the relevant service/apis for each QD were identified the automated test. Therefore verify and validate the performance of approach by precision and recall.

Performance measures:

$$\mathbf{Precision^{K}(Q_i)} = \frac{|\mathbf{Relevent(Q_i)} \cap \mathbf{Rank^{K}(Q_i)}|}{|\mathbf{Rank^{i}(Q_i)}|} \quad (6)$$

$$\mathbf{Recall^{K}(Q_i)} = \frac{|\mathbf{Relevent(Q_i)} \cap \mathbf{Rank^{K}(Q_i)}|}{|\mathbf{Relevent(Q_i)}|} \quad (7)$$

$$\mathbf{Precision^{R}} = \frac{|\mathbf{Relevent(Q_i)} \cap \mathbf{Rank^{R}(Q_i)}|}{|\mathbf{Rank^{R}(Q_i)}|} \quad (8)$$

Where Q_i = ith user query, K = services/api's returned, value of (1-12) and observed the effects R = no of relevant services/api's for a user query, Relevant (Q_i) = Set of Relevant services/api's with numbers of ith user query, Rank (Q_i) = Top-K services/.api's returned.

V. RESULTS

The performance compares the TASSIC and new service classification approach in entered the separated 10 user input query document, and the search for set of similar suitable service's/api's. The retrieving services/api's to their match score. Calculate the accuracy of Tassic and new service classification approach.

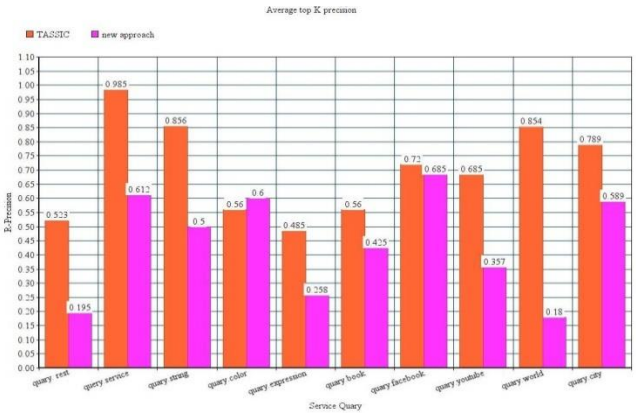

Figure 2. R-Precision

The first experiment R-precision results figure 2 is bar chart belong to compare of the service query data in the searching the 10 user defined service query is x axis and another y axis is R precision. when the search processing of the degree of similarity based on user query and service document the Euclidean distance calculating the match score (or ranking) in this scheme high performance, due to 10 experiments are average match values are normally, their compare the TASSIC and new approach that is service classification of RESTful service discovery approach will be best one.

The second experiment comparing of efficiency and accuracy of restful service discovery by service classification approach results figure 3 is bar chart. The input service documents and accuracy in the searching document services by the (10~180) service documents. Their documents are classified the restful services discovery in the service classification algorithm and calculate the accuracy of the approach in the graph x-axis is service documents and another y-axis is accuracy. The service classification approach of RESTful service discovery will be highly accurate, improve the search precision, and recall their services.

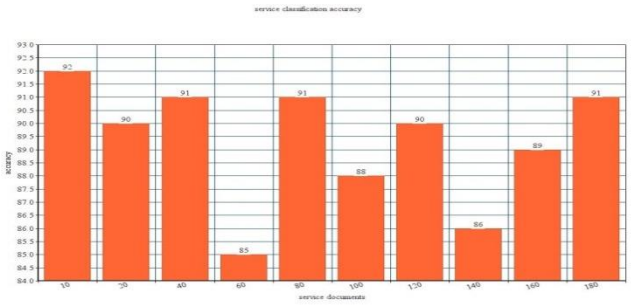

Figure 3. Comparison between services efficiency vs accuracy

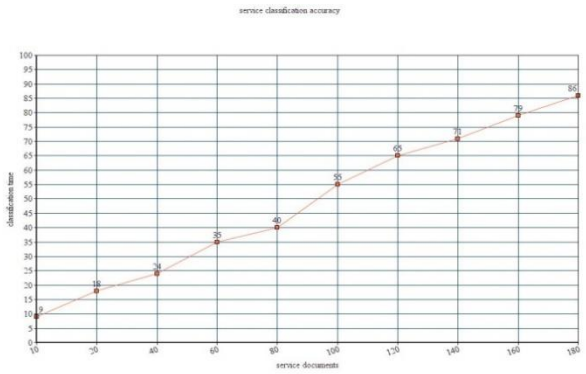

Figure 4. Comparison between services finding efficiency vs classification time

The third experiment comparing services documents and classification time of restful service discovery by service classification approach result figure 4 in the line chart. The searching document services by the (10 ~ 180) service documents. The set of documents are classifying and calculating classification time. This graph representing in the x-axis is service documents and another y-axis is classification time. The service classification of RESTful service discovery approach increasing sampling documents and it improve the execution time and increasing search efficiency.

VI. CONCLUSION & FUTURE WORK

The main aim of project integrating service classification and service clustering algorithms is to enable the reduction of search space on RESTful service discovery by the classifying and grouping relevant services/api's and their match scores each services/api's. The proposed algorithms are KNN classification and K-Means Clustering. These algorithms works on classify the and similar services/api's into one cluster and keeps on iterating till all similar services arrange in appropriate cluster. By seeing clusters user can understand which similar services. To verify the searching the RESTful service discovery is present the published RESTful services / api's or not to check that reason unit testing is performed in this approach. The RESTful web service discovery, user can search a queries in search engine it retrieves the services/api's very quickly response accurately retrieval of published service's / api's to enhance RESTful service discovery improved the efficiency.

In further research, work users request and Quality of service information. its planning to implement the other service classification algorithm are used to improve the restful service discovery based on quality parameter constraints. Use other clustering algorithms also to implement.

VII. REFERENCES

[1] Shang-Pin Ma*, Ying-Jen Chen, Yang Syu, Hsuan-Ju Lin, and Yong-Yi Fanjiang, "Test-Oriented RESTful Service Discovery with Semantic Interface Compatibility" IEEE Transactions on Services Computing, 2018.

[2] R. T. Fielding and R. N. Taylor, "Principled design of the modern web architecture," ACM Transactions on Internet Technology, vol. 2, pp. 115--150, 2002.

[3] M. Paolucci, T. Kawamura, T. R. Payne, and K. Sycara, "Semantic matching of web services capabilities," in The Semantic Web—ISWC 2002, ed: Springer, 2002, pp. 333-347.

[4] M. Garriga, C. Mateos, A. Flores, A. Cechich, and A. Zunino, "RESTful service composition at

a glance: A survey," Journal of Network and Computer Applications, vol. 60, pp. 32-53, 2016/01/01/ 2016.

[5] Alex Rodriguez, RESTful Web services, Updated February 9, 2015 - Published November 6, 2008.

[6] Reihaneh Rabbany Khorasgani ; Eleni Stroulia , Web service matching for RESTful web services, 2011 13th IEEE International Symposium on Web Systems Evolution (WSE), DOI 10.1109/WSE.2011.6081829.

[7] SWSF (2005). Semantic Web Services Framework (SWSF), Overview, W3C Member Submission 9 September2005,http://www.w3.org/Submission/SWSF/.

[8] Vikram Singh and Balwinder Saini, An Effective Tokenization Algorithm For Information Retrieval Systems.

[9] Ramalingam Sugumar & M.Rama Priya, Improved Performance Of Stemming Using Enhanced Porter Stemmer Algorithm For Information Retrieval,] DOI: 10.5281/zenodo.1228745.

[10] C.Fellbaum WordNet: An Electronic Lexical Database (Language Speech and Communication) The MIT Press May 1998.

[11] S. K. M. Wong, W. Ziarko, P. C. N. Wong, "Generalized vector space model in information retrieval," in the 8th Annual International ACM SIGIR Conference on Research and Development n Information Retrieval, New York, 1985, pp. 18-25.

[12] Lathem K. Gomadam A. P. Sheth "Sa-rest and Adding semantics to RESTful services" in ICSC IEEE Computer Society pp. 469-476 2007.

[13] Shailja Sharma1 · J. S. Lather2 · Mayank Dave3 "Semantic approach for Web service classification using machine learning and measures of semantic relatedness".

[14] Z. Zhen, C. Shizhan, and F. Zhiyong, "Semantic Annotation for Web Services Based on DBpedia," in Service Oriented System Engineering (SOSE), 2013 IEEE 7th International Symposium on, 2013, pp. 280-285.

[15] Gabriela D. A. Guardia, Luís Ferreira Pires, Ricardo Z. N. Vêncio, A Methodology for the Development of RESTful Semantic Web Services for Gene Expression Analysis, Published: July 24, 2015, https://doi.org/10.1371/journal.pone.0134011.

[16] Cardoso J, Sheth A (2005) Introduction to semantic web services and web process composition. Semantic Web Services and Web Process Composition: 1–13.

[17] Klein, M. and A. Bernstein (2001). Searching for Services on the Semantic Web Using Process Ontologies. International Semantic Web Working Symposium (SWWS), Stanford University, California, USA.

[18] Sheth, A., C. Ramakrishnan, et al. (2005). "Semantics for the Semantic Web: The Implicit, the Formal and the Powerful." Intl. Journal on Semantic Web and Information Systems 1(1): 1-18.

[19] Shimaa E. El-Sayyad, Ahmed I. Saleh, Hesham A. Ali A new semantic web service classification (SWSC) strategy, doi.org/10.1007/s10586-018-2367-9

[20] C. Wu, E. Chang, and A. Aitken, "An empirical approach for semantic web services discovery," in 19th Australian Conference Software Engineering, 2008.

[21] Yang Xue, Chunhong Zhang, Yang Ji, "RESTful Web Service Matching Based on WADL", Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC) 2015 International Conference on, pp. 364-371, 2015.

[22] X. Dong A. Halevy J. Madhavan E. Nemes J. Zhang "Similarity search for web services" Proceedings of the Thirtieth international conference on Very large data bases vol. 30 pp. 372-383 2004.

[23] E. Stroulia Y. Wang "Structural and semantic matching for assessing web-service similarity" International Journal of Cooperative Information Systems vol. 14 pp. 407-437 2005.

[24] W. Abramowicz K. Haniewicz M. Kaczmarek D. Zyskowski "Architecture for web services filtering and clustering" Internetand Web Applications and Services (ICIW) pp. 18 May 2007.

[25] Martha Varguez-Moo, Francisco Moo-Mena and Victor Uc-Cetina, Use of Classification Algorithms for Semantic Web Services Discovery, doi:10.4304/jcp.8.7.1810-1814.

[26] S. Sharma, J. S. Lather, and M. Dave, "Semantic approach for Web service classification using machine learning and measures of semantic relatedness," Service Oriented Computing and Applications, vol. 10, pp. 221-231, September 01 2016.

[27] Abdelmoniem Helmy, Mervat H. Geith, An Enhanced Approach for Web Services Clustering using Supervised Machine Learning Techniques. International Journal of Scientific & Engineering Research, Volume 8, Issue 1, January-2017, ISSN 2229-5518.

[28] Vijayan, A. S., & Balasundaram, S. R. (2013, February). Effective web-service discovery using K-means clustering. In International Conference on Distributed Computing and Internet Technology (pp. 455-464). Springer Berlin Heidelberg.

[29] Venkatachalam K, Dr.S.Balakrishnan, Dr.R.Prabha, S.P.Premnath, Effective Feature Set Selection And Centroid Classifier Algorithm For Web Services Discovery, Volume 119 No. 12 2018, 1157-1172

[30] Soumi Chattopadhyay, Member, IEEE, and Ansuman Banerjee, Member, IEEE," A New Methodology for Search Space Reduction in QoS Aware Semantic Web Service Composition".

Cite this article as :

G. Venugopal, Dr. P. Radhika Raju, Prof. A. Ananda Rao, "Reduction of Search Space in Restful Service Discovery", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 5 Issue 4, pp. 143-152, July-August 2019. Available at doi : https://doi.org/10.32628/CSEIT195430
Journal URL : http://ijsrcseit.com/CSEIT195430