Semantic Annotation Based Effective and Quality Oriented Web Service Discovery

J. Brindha Merin[1], Dr.W. Aisha Banu[2*] and K. Fahima Sanobar Shalin[3]

[1]Assistant Professor (Senior Grade), Department of Computer Science and Engineering,
B.S. Abdur Rahman Crescent Institute of Science and Technology, Vandalur, Chennai, India.
brindhamerin@gmail.com, Orcid: https://orcid.org/0000-0001-9736-1061

[2*]Professor, Department of Computer Science and Engineering,
B.S. Abdur Rahman Crescent Institute of Science and Technology, Vandalur, Chennai, India.
aisha@crescent.education (Corresponding Author), Orcid: https://orcid.org/0000-0001-7799-6393

[3]Student, Department of Computer Science and Engineering,
B.S. Abdur Rahman Crescent Institute of Science and Technology, Vandalur, Chennai, India.
fahima1401@gmail.com, Orcid: https://orcid.org/0009-0000-9938-0418

Received: February 22, 2023; Accepted: April 03, 2023; Published: May 30, 2023

Abstract

The main objective of this work is to design an effective web service discovery engine that retrieves the best matching results based on the measure of syntactic cum semantic similarity between the user query and the web service to be fetched. The idea is to draw bridges across the broad spectrum of lexical entities based on their relativeness. The essence of this work could be extended to support a wide range of applications from 'being inculcated in search engines to fetch user-relevant data' to 'being used for training robots and AI based devices to respond/adhere appropriately to the different phrases of human commands'.With the accelerated revolution of internet, enterprises and organizations highly rely on Service oriented computing. Web services support inter-operation of distributed applications. Humongous amount of web services present on the internet the user is searching for. The paper focuses ondesigninga practical means of fast and relevant retrieval of web services.The phrase used for searching any given web service differs from one person to another. This work deals with the syntactic cum semantic mechanism proposed for retrieving web services based on the measure of similarity between the name of the web service and the search query entered by the user to discover that service. A list of the Web Service Description Language (WSDL) files is taken as the dataset. Protégé is the tool used for semantic annotation of WSDL files for converting them to Semantically Annotated Web Service Description Language (SAWSDL) files. 'Wordnet' is used as the lexical dictionary and Java language has been used to build the rest of the package for implementing the search discovery mechanism.Netbeans is used as IDE.Wamp server with PHPMyAdmin was used for managing the database of SAWSDL files. The degree of similarity is measured by evaluating performance of the engine using metrics such as 'Precision', 'Recall', 'Accuracy' and 'F-measure'.While the syntactic approach is easier to implement, it suffers from keyword polysemy issues. The proposed search discovery mechanism is based on semantically annotating WSDL files and retrieving the files based on a novel syntactic cum semantic discovery algorithm which usesLeocockChordorow function for computing the similarity. The effectiveness of the proposed algorithm is tested experimentally by building a desktop application using Java.The

Journal of Internet Services and Information Security (JISIS), volume: 13, number: 2 (May), pp. 96-116.
DOI: *10.58346/JISIS.2023.I2.006*

*Corresponding author: Professor, Department of Computer Science and Engineering, B.S. Abdur Rahman Crescent Institute of Science and Technology, Vandalur, Chennai, India.

WSDL files from various domains were semantically annotated by tagging related concepts using ontology. Themachine learning algorithm that best classifies the web services on the basis of their performance metrics is identified. The related services are retrieved by the application of the proposed LCH based Syntactic cum Semantic discovery algorithm (LCH based SSDA).A ranking system is proposed to rank the results by evaluating various QoS attributes. The results of the experiment showed that the proposed system yielded high precision and recall value.The solution has been found to be effective in minimizing the execution time and in improving the degree of relevancy. With an impending need for constructing a semantic context based secure structure,the proposed solution will- help in refining the search results and in minimizing users' cognitive load during search formulation and execution.

Keywords: Semantic Annotation, Classification, Web Service Discovery, Ontology, Semantic Similarity, WSDL.

1 Introduction

Web Services pave the way for entities over the World Wide Web to communicate. When a search engine, enterprise system or any application is used, the user interface dwells on our device. The server stores the data and other business rules required. The communication between the interface and the piece of server is facilitated via the web services. Today, web services are used in different fields such as ecommerce and app development.

Web services assist in the interaction and inter-operation among multiple software and applications in fetching the most appropriate results to the plate. Sometimes the user might encounter the need to enforce selection criteria to return specific information. The life cycle of web service composition involves the collection and arrangement of independent Web services; where high level service-based applications aim to deliver additional features. The various stages include: requirement specification phase, discovery phase and the selection of requirement satisfying services.

As depicted in Figure 1, there are three different components of web services. This includes- the web service provider, the service consumer and the Service directory or registry such as the UDDI consisting of the WSDL files.The Service registry stores the web services. Consumers find web services by searching through this service registry.

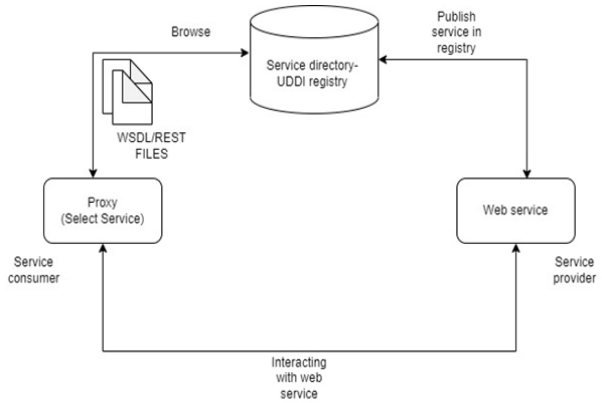

Figure 1: Structure of Web Service

It acts as a bridge between web service-provider and the one who consumes the service. Moreover, it aids in helping multiple service providers in publishing the web services they offer to the users. Service discovery refers to the identification of service which is more or less equivalent to the search criteria.

Semantic annotation describes the practice of tagging the required documents with keywords and concepts that are relevant to it. The tagged concepts basically serve as a metadata to refer to the entity that is being annotated. The annotations could be stored either in the document or can be represented as knowledge graphs wherein each node is used to denote a particular entity.Semantic annotation gathers smart pieces of well-structured data that help as informative reference notes for machines (Belhajjame, K., Embury, S. M., & Paton, N. W.,2013). Semantic annotation has been proved to be widely effective in use cases like risk examination, content recommendation, content discovery, detection ofsupervisory compliance etc. Semantic annotation helps to search beyond keywords and discover relationships that exist beyond human observation and for content aggregation.

The proposed system makes use of semantic annotation to annotate the Web Service Description Language files to form SAWSDL files, which will aid in effective discovery of services during the search process. The annotated keywords are then extracted to be used for web service discovery. A tool called Protégé has been used to visualize the structure and knowledge graph of the various concepts by defining the class hierarchy for the same.

Generally service discovery can be accomplished either using syntactic discovery or semantic discovery. With syntactic discovery, the main technique used is keyword matching. Despite being easy, this approach results in low precision and recall rate and often returns ‘no results found’ when an exact match isn’t identified. Moreover, it is highly impractical to assume that all possible keywords are pre-fed into the system for every single service. Semantic discovery on the other hand, uses ontology to boost the clarity of the semantics of web service description (Driss, M., Ben Atitallah, S., Albalawi, A., & Boulila, W, 2022),. This leads to the necessity for a semantic tagged language like OWL-S (Web Ontology Language for Services) or SAWSDL (Semantic Annotations for Web services description language). This method would result in a better precision and recall when compared to the previous method. With the effective functionality, semantic discovery seems to be a more viable solution for the implementation of service discovery. In spite of the clearly visible advantages and benefits of adopting the semantic approach for web service discovery, there are several issues that challenge the implementation. Relationship existing among the concepts in a hierarchical ontology is to be considered. At times, semantic approach consumes relatively more time which might be a bottleneck to performance.

Web services are used by developers as well asnormal users. A developer might be fully aware of the expected values of therequired service’s functional and non-functional parameters. For instance, he would search for a web service that has a specific response time below a certain value (‘x’ ms). On the other hand normal users would generally prefer selecting the service ranked on the top believing that they are the ones that excel in terms of quality. Thereby, it could be inferred that there is a need to develop a separate service recommendation portal for developers and a separate one for the normal consumers to enable efficient and easy search scheme.

With the wide usage of web services, the obvious necessity for service discovery is evident. Developers who are in need of web services can develop their own services. If not, they can search for the required one. In registries such as ‘Programmable Web’, web services are available for public access. Hence, the need to create a web service recommendation system considering quality of service (QoS) and history could be inferred.

With tons of web services being available, there is a need to develop classification models using advanced techniques to help classify any random web service based on quality (Mohanty, R., Ravi, V., & Patra, M. R, 2010). In this project, classification models were built using SVM and Random Forest for predicting the quality of web service based on a set often attributes pertaining to Quality of service. The prediction models are implemented on the basis of historical data considering the explanatory

variables as the various QoS attributes and the Overall quality in turn as the dependent target variable. Each QoS attribute identifies a unique dimension of quality. The collective effect of the QoS attributes on the quality of service is found to be a non-linear relationship; which is normalized using the chosen methods of classification models. This is done with an intention to efficiently guess the quality of a web service that is yet to be classified.

The QoS attributes are based on internal environment dependent attributes, which are not dependent upon the service environment (Kritikos, K., & Plexousakis, D., 2009). QoS is an integral part of evaluating the performance of web services. Thus it has been widely used in applications such as networking and multimedia. When applied to web services, it would help in improving the retrieval through replication, load distribution and service redirection.It is common that a particular service might be offered by more than one service provider. Therefore, the proposed classification model can be used as the basis for ranking the services so that the user would most likely select the one that ranks higher in terms of the QoS metrics.

Apart from these, 'Location' plays an important role in recommendation systems because the quality of web services can vary from one location to the other. Hence,aweb service recommendation system including the consideration of location and QoS metrics is found to be essential.

Considering all the above constraints and inferences, the proposed approach is implemented by semantically annotating the WSDL files and loading them into the dataset. The discovery of services is done via a matching algorithm that tends to weigh the relativity between the search request and service descriptions. Besides making use of keyword matching technique, the proposed approach makes use of Leacock Chordorow for measuring the similarity. This has been done to progresstheefficiency and enhance the accuracy of search results to a great extent. Intelligent machine learning techniques have been employed to build classification models for predicting the rank of a new web service by training the existing dataset based on various QoS attributes. This is done towards facilitating the ranking of web service for normal users. Our approach further involves a separate QoS value based filter system to ease the search experience for developers. In each case, the precision and recall are calculated and are evaluated to generate higher results in comparison to the schemes implemented earlier. This approach has been proposed with the view of providing a seamless search experience to the consumers.

Objectives of this project are

- To design an efficient search discovery mechanism using semantic approach
- To provide higher degree of relevancy.
- To rank the services based on functional and non-functional parameters
- Retrieval based on QoS constrained service requirement specified by the user

In order to achieve the above mentioned objectives, an efficient web service recommendation portal using 'LCH based Syntactic cum Semantic discovery method' (**LCH based SSDA**) has been proposed to generate expected results to the users. With this system, the users can register themselves either as developer or normal user; and each is designed to have their own workflow. Moreover, secure registration and login with admin authentication and random id generation has also been inculcated.

The proposed system is beneficial in many aspects. Semantic annotation of web services helps in improving the accuracy of search results. The proposed search algorithm using a blend of syntactic and semantic discovery method overcomes the limitations of both approaches.

Consideration of QoS metrics into the service retrieval engine helps in delivering services that rank fairly in terms of performance. The system yields higher precision of results and precision of context. It

also provides location specific search results to users. The use of SAWSDL files helps in reducing the time taken by the graphic oriented processing of complex OWLS files.

The remainder of this document has been structured as follows. Section 2 presents a survey on the recent works carried out on discovering web services. Section 3 describes the limitations of the existing system, the proposed semantic discovery approach, its architecture and the algorithms employed. Section 4 deals with the various modules and system requirements. Section 5 elaborates on the process of implementation and finally Section 6 presents the conclusion and proposal for future work.

2 Related Works

Web service discovery is still an open problem to be addressed with multiple areas of research. Different approaches and techniques have been put forth for automating and improvising the discovery and composition framework. MahaDriss has implemented a service composition framework taking OWLS files as dataset. The matchmaking has been done by using the Wu Palmer algorithm while filtering is based on QoS and QoE metrics. The problem here is that the OWLS files are not suitable for legacy systems. Moreover OWLS files will result in higher code complexity and hence will consume comparatively more time with the hierarchical ontology structure. Their approach focused on filtering based on user specific QoS metrics. However, there was no customized system to cater to the ease of use for normal users and developers. Moreover, the existing approach is found to be exhausting for large number of web services. The solution proposed consist of a discovery phase that filters and retrieves web services from a registry using semantic discovery algorithm to compute the semantic similarity called the Wu Palmer algorithm (Driss, M., Ben Atitallah, S., Albalawi, A., & Boulila, W, 2022). However, the computation of similarity using Wu Palmer algorithm failed to consider the distance of the shortest path and focused only on depth.

A technique of upgrading the quality of semantic annotations using a tri-phase optimization framework was proposed (Huang, K., Zhang, J., Tan, W., Feng, Z., & Chen, S. ,2016). The framework was designed to include the Local and Global-feedback strategy to support successful invocations. Further a Global-propagation strategy was proposed to revise the annotations in course of false operations. It was found to have gained 78.68% improvement in Quality of Semantic annotations and was hence proved worthy to be adopted in service repository aiming to accelerate the performance of its encompassing services. Though it was a good initiative to include user feedback, it suffered from the possibility of fake users accounting for fake ratings.

Works have also been carried out in areas of secured semantic search. A confidential method of semantic search to bridge the gap between ontology and encryption was proposed (Yang, W., & Zhu, Y. 2020). His scheme was used to match the query with the documents stored in public cloud using optimal matching out on the cipher text by verifiable semantic search. Minimum Word transportation cost was taken as a metric of correspondence. This proposal gives an interesting suggestion to embrace encryption algorithms with semantic discovery.

The work suggested using ML algorithms to predict the service quality based on a set of parameters. He (Mohanty, R., Ravi, V., & Patra, M. R. 2010) proposed a model for feature selection, classification and rule-based generation. The algorithms were applied on the dataset for feature selection and in each case the accuracy of algorithm was computed. It was inferred that for every algorithm different attributes proved to be significant. He suggested classifying the web services into four groupsas: Bronze, Silver, Gold and Platinum based on the condition of WSRF values. His work demonstrated that BPNN

algorithm showed a maximum accuracy of 86.11% in classifying the services. However, his approach failed to deal with the accuracy of discovering services and focused only on classification

On the other hand, (Fang, M., Wang, D., Mi, Z., & Obaidat, M. S. 2018) proposed a service discovery method called OFPM (Ontology Filtering and Parameter Matching). The WSDL files along with various "model Reference" annotations form Semantically Annotated WSDL files. The URI of SAWSDL file is parsed to pull out the distinct ontology terms. Hence, the services that reference all ontology terms present in the request are selected by the pre-processor. The selected services are passed through a Reasoning-based service filter that selects all services based on logical matching which are further matched with the request query based on concept relationship using the PBSM_R Relationship-Aware and Parameter-Based Service Matcher; inferring a good precision and recall value.

A work (Merin, B., & Banu, W. A. 2020) suggests the use of Hungarian Algorithm, as the best choice for semantic operation with minimal execution time. The services that are similar in nature were identified and the Semantic web services were modelled with a bipartite graph. The obtained ontology has provided good throughput but consumes a lot of time to respond.

(Moradyan, K., & Bushehrian, O. 2015) has proposed an ontology model to originate the structure of service consumer's request. A semantic based matchmaking is carried out. A set of three approaches including the syntactic go-discovery approach has applied and their proposed approach has been verified to have yielded a good precision value .

A structural design for the discovery of RESTful semantic service has been proposed by using SERIN (Dantas, J. R. V., & Farias, P. P. M. 2020) as the interface for interpreting its description. The design has been entitled to enable automatic retrieval and access call to a specific service.

In order to improve the extensibility of OWL-S, (Sandhya, S., Pabitha, P., & Rajaram, M. 2014) introduced an algorithm that maps the WSDL files to WSMO files and a discovery engine which uses Bi-Clustering algorithm. The efficiency of mapping has been evaluated using Performance gain Factor which has been computed based on the amount of time and resources taken for the discovery.

A novel method for the cloud based classification of web services was proposed by Mohamed S. Alshafaey, where the input services are scanned and filtered using Concepts Preparation Module. The significant topics were selected in a sequential form to reduce the time consumed, the Tree Creation Module was introduced to assign weights and generate a tree structure based on the semantic relationship; followed by the 'Change, Edit, Add Module' to get the best match response to the consumer's query (Alshafaey, M. S., Saleh, A. I., & Alrahamawy, M. F. 2021).

Works on graph-oriented service composition based on its reliance with discovery has been carried out (Rodriguez-Mier, P., Pedrinaci, C., Lama, M., & Mucientes, M. 2015). The proposed algorithm for optimal search composition derives its structure from the graph and in turn would help in reducing the length and the number of services. The framework's progression is initiated by a composition request describing the user requirements (specified as inputs and the most likely intended outputs). The composition graph generation phase takes up the requirement to build a graphical representation containing the pertinent services anddisplays the semantic relations among the inputs and outputs. The approach interleaves this phase with the discovery phase. Following this, the matchmaking segment is used to evaluate the identical degree between inputs and outputs using a Semantic Reasoner. The composition graph is produced and optimised. Finally, a search optimization phase is used to analyse and reduce the optimal composition workflow. Index based and graph based optimizations are introduced for the same. However, the usage of matrices for graphical computations is a hard and expensive process.

(Cheng, B., Li, C., & Chen, J. 2018) has proposed a discovery structure using the approach of semantic mining for the interface parameters. Web service formalizing model deals with the elementary information of the service and this information is extracted from the basic description. Secondly, a service matching engine has been proposed with no dependence on semantics ontology. Finally a discovery framework has been proposed based on the index library aimed to reduce the processing time.

(Kritikos, K., & Plexousakis, D. 2009) analysed the impending need for a semantically rich QoS centred WSDM and an operative QoS oriented Web Service Discovery. He has proposed the procedure to produce a QoS broker that could be used to aid the serviceable Web Service registries that are currently available. The QoS broker is envisioned to consist of the QoS Information Collector for fetching QoS details from Service provider, third party monitoring systems, user feedback etc; QoS Concept Mapper has been intended to match the specification with the ontological description and the QoS Selector returns the result using the selection algorithm. The duty of the entities in prospect of the QoS broker wasanalysed; but not implemented.

Mining algorithms were proposed to overcome combinatorial explosion. George Zheng proposed mining of web services for the activeidentificationof service compositions. The pre-screening planning involved specifying the Scope and determining the search space. The screening stage included operation level filtering algorithm. When publishing the corresponding agent checked for subscribers to the interface and tried to initiate a composition lead between the subscriber and the publisher. The author has envisioned research labs to present their finding of discrete biological processes via Web services (Zheng, G., & Bouguettaya, A. 2009). The WSDL services were bounded with WSML and deployed into WSMX runtime environment. The proposed framework was used for the detection of biological linkages

Bipin Upadhyaya insists on the importance of choosing quality services for composition. Quality of Experience attributes are automatically extracted from user reviews using web crawlers and natural language processing (NLP techniques with POS tags) and stored in the database. His approach was successful in identifying all QoS related data from the reviews (Upadhyaya, B., Zou, Y., Keivanloo, I., & Ng, J. 2015). The proposal also highlighted that the QoS and QoE attributes are highly inter-related and that one can be considered in the absence of the other. However, it failed to take into account, the semantic relativity.

Semantic annotations have been identified as a possible scope for service retrieval in the recent years. Khalid Belhajjame has proposed a method to verify if the semantic annotations serve as a correct description of the service's behaviour. Annotation testing package was created using 'All-Disjoint Concepts' strategy (Belhajjame, K., Embury, S. M., & Paton, N. W. 2013). In addition,anarbitrary similar sized test suite was also generated. The success of the partitioning strategies was measured in terms of 'Increase in Recall and Precision'. It was concluded that partitioning using ontology had a distinct edge over the randomly selected suites and was also found to improve the defect-detection power.

The approach of semantic service discovery comprises of grouping and augmentation of semantic service request (Paliwal, A. V., Shafiq, B., Vaidya, J., Xiong, H., & Adam, N. 2011). Functional service categorization was achieved using ontology. Clustering was employed for organizing the web services on the basis of service functionality. The enhancement of service request involved expansion of additional terms relevant to the requested functionality. The mapping of request to the web service description was achieved utilizing Latent Semantic Indexing (LSI), where if a match was not found suggestions would be given to add additional input to return a matched service.

Re-shaping of traditional web services with semantic annotations using a mapping algorithm has been put forth. (Farrag, T. A., Saleh, A. I., & Ali, H. A. 2012). Native ontology repository and Onto-search engine were considered as the integral part of his system. The proposed algorithm was found to minimize the time interval and effort required for the mapping process.

To enable the users to select services meeting the QoS requirements, the author has proposed a semantic service selection model that focuses on clustering (Natarajan, B., Obaidat, M. S., Sadoun, B., Manoharan, R., Ramachandran, S., & Velusamy, N. 2020). He has also proposed a user preferential model. These were built to map the requested service with the discovered ones and the required non-functional specifications to the QoS parameters of the servicesretrieved. The attributes were determined using a two-level user preference model. The proposed Service Selection Model and the User Preferential Model were found to improve the competence of service selection; but lacked in efficiency of service discovery.

Efficient recommendation of services can be achieved by incorporating domain and Web usage knowledge of a website. In the model proposed two new models were proposed for domain knowledge(Nguyen, T. T. S., Lu, H. Y., & Lu, J. 2013). While the first one makes use of ontology, the second model uses automatically generated semantic network to denote the domain terms and relations.

Furthermore, a concept oriented prediction model has been proposed for auto-generating a network of semantic Web usage knowledge that integrates both domain knowledge and Web usage knowledge. The results of the various queries applied on the knowledgebase demonstrated that the proposed system produced relatively much higher performance than the WUM method. However, the model suffered from long execution time.

Taking cue from the various works that have been carried out, it could be inferred that the existing approaches do suffer from certain limitations; which are intended to be overcome by the proposed methodology.

3 Methodology

In this section, the proposed approach to develop a service discovery framework by considering the user's requirements and semantic relatedness is discussed.

The focus area of this implementation is to discover the web services that are most appropriate to be returned, for any given phrase that could be used by the user when they perform their search for web services.The existing methods either follow a complete keyword based approach or a semantic approach. The novelty of the proposed methodology lies in the combined syntactic and semantic nature of the underlying architecture discussed below in Figure 3 and in the proposed LCH based syntactic cum semantic discovery algorithm (**LCH-based SSDA)** (discussed in section 3.3) that is used for fetching the services (Kolomeets, M., 2019). The motivation of this work is to avoid the frustration and the unintended delay that thedeveloper or a general user might face in scrolling through and picking up the right service when an irrelevant bunch of web services are returned.

Limitations of the Existing Systems(Keyword-Mapping Approach & OWL- based Approach) for Web Service Discovery

In the existing system, the WSDL files are more human interpretable. In the keyword matching approach currently being used, when there are a greater number of WSDL files, it may result in the failure of result discovery. In the base paper (Driss, M., Ben Atitallah, S., Albalawi, A., & Boulila, W. 2022), has

implemented a service composition framework taking OWLS files as dataset. The matchmaking has been done by using the WuPalmer algorithm while filtering based on QoS and QoE metrics. The problem here is that the OWLS files are not suitable for legacy systems. Moreover, OWLS files will result in higher code complexity and hence will consume comparatively more time with the hierarchical ontology structure. Their approach focused on filtering based on user specific QoS metrics. However, there was no customized system to cater to the ease of use for normal users and developers. Moreover, the existing approach is found to be exhausting for large number of web services. The absence of location specific service retrieval is yet another limitation with the existing system. The main problem statement identified is that most of the existing approaches go by a syntactic matching where the search query is matched against the service name and is retrieved only when an exact match is found. This results in low efficiency of service discovery. Moreover, the keyword-based approach suffers from keyword polysemy issues. The number of services that could be retrieved would also be low.

Secondly, it would be difficult for users to select the best service among the discovered list of services without taking Quality of Service attributes into consideration. The absence of an efficient ranking scheme often disappoints the service consumer who blindly believes that the services on the top are the best ones.

Furthermore, different versions of web services might exist for different locations. However, these versions won't differ in their basic WSDL file structure. Hence, this creates hindrance to location specific retrieval of services.

Finally, the criteria of service consumers must be taken into account. Not all users would prefer a similar service recommendation system. Developers might prefer the facility to extract web services based on specific values of QoS metrics while normal users would prefer an auto-ranking system.

The proposed approach will help overcome all the above pain points by means of meaningful solutions.

Proposed System - (LCH-based SSDA)

The proposed system uses semantic annotation, a branch of ontology - as the pillar to relate the different linguistic entities by classifying them as identical groups of wordsets and establishing connections based on their contextual relation (be it synonym, antonym, meronym or hyponym). The proposed system not just calculates the similarity between the tokenized search phrase and the web service name; but instead computes the semantic similarity between the search phrase and each candidate meaning (obtained by the lexical pool of wordnet) corresponding to every single web service. This computation logic is supported by the algorithm proposed in section 3.3 using the LeocockChrodorow function.

Ontology

The concept of Ontology is the backbone of the proposed system . Ontology deals with the process of how the different entities are grouped together and the hierarchy that exists among them. While ontology mostly deals with classification, Semantics is one branch of Ontology that helps in better classifying the world of diverse words, literals and meanings. It helps in understanding the closeness of two given word sets.

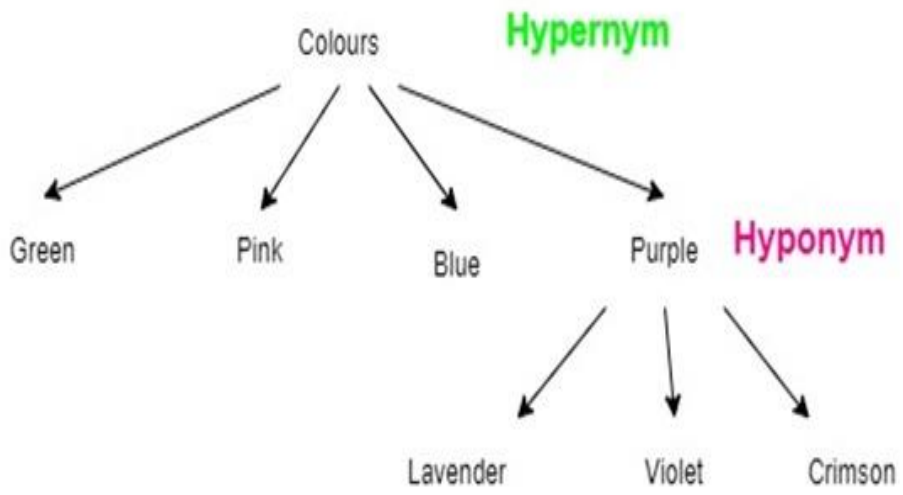

Figure 2: Lexical Ontology

Natural language is a unique model which resolves the linguistic context in relativeness with other cognitive systems. Moreover, the handling of natural language requires special care as it is very much associated with the the conceptual world. The ontology of natural language refers to the semantics of words, phrases, synonyms and other lexical entities.

In the proposed system design, the user has the privilege to register themselves, either as a normal user or a developer.Upon the user request, the admin would be notified, who in turn will generate a unique user id. After registration is done, the user must login using his/her credentials along with the user id generated during the registration process. Once the login request has been placed, the user has to wait for Admin acceptance. The admin would be notified about the list of users requesting login access. On admin acceptance, the user can successfully login to the secure search portal. The admin has the privilege to upload the WSDL files into the database.

As shown in Figure 3, the design flow depends upon the chosen role of the user. In case of a normal user, the user can enter the keyword to search for the relevant web service. The search query is pre-processed to eliminate the stop words. The semantic discovery algorithm uses Leocock Chordorow Algorithm that computes the similarity measure. The discovered services can then be optionally filtered based on location preferences. The filtered services are ranked based a number of functional and non-functional parameter, such that the best service is retrieved on top of the list.

In case of users registered as developer, the design flow post login is different. As developers would be well aware of the exact type of web service they want to fetch, they can search for appropriate web service by specifying the values for various ‘Quality of service’ metrics. The filtered results would be displayed by using the semantic discovery algorithm.

The system takes the Web service description language files as input, semantically annotates the different files with relevant keywords, classifies the services based on QoS metrics, performs keyword mapping for semantic discovery and ranks the classified services. The output obtained is a list of relevant web services.

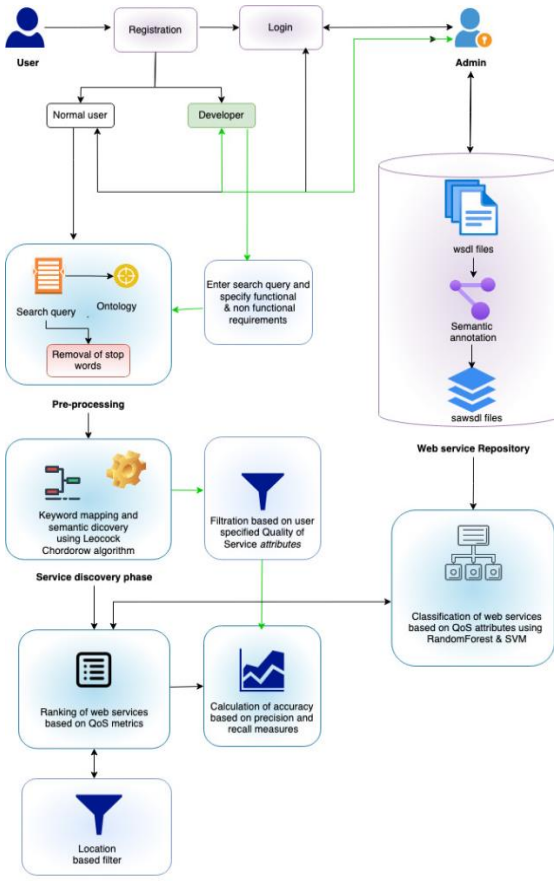

Figure 3: Architecture Diagram

The proposed architecture consists of four major components-

(1) Semantic annotation and pre-processing,
(2) Classification of web services,
(3) Web Service discovery,
(4) Ranking and Requirement specific retrieval.

Each of the component depicted in Figure 3 is discussed in detail in the following sections.

The first phase Section 3.1 discusses the process of annotating the WSDL files and the methods used for pre-processing the dataset. Section 3.2 deals with the analytic study of machine learning algorithms to best classify the services on their QoS metrics. The service discovery module (discussed in Section 3.3) is the area of prime focus. This phase describes the novel syntactic cum semantic discovery algorithm that has been proposed to improve the efficiency of web service discovery process. Finally, Section 3.4 deals with the proposal of QoS based ranking of web services for normal users and Section 3.5 describes the requirement specific retrieval of web services for developers.

Semantic Annotation and Pre-Processing

The proposal is based on semantically annotating the WSDL files and converting the functional and non-functional requirements into ontological descriptions using Protégé. Semantic annotation represents additional information about a particular entity to facilitate identity discovery. Pre-processing involves POS tagging and extraction

Classification

Classification of web services into different classes (such as Platinum, Silver, Gold and Bronze) has been proposed by performing an evaluative analysis of different Machine Learning Algorithms.SVM and Random Forest were chosen mainly for the evaluative study to find the factor which contributes the most to the overall Quality of service.

Support Vector Machine

Support Vector Machine is a machine learning algorithm that identifies a hyperplane in N-dimensional space to predict the continuous output and a decision boundary which acts as a demarcation line. It classifies the new point depending on whether it lies on the positive or negative side of the hyperplane depending on the classes to be predicted. SVM is generally suitable for prediction using datasets that contain multiple features

Random Forest Algorithm

Random forest classifier is a supervised machine learning algorithm. While it can be used for both classification and regression, it works best for classification. In Random forest, a collection of models are used to make the prediction. Random forest has been chosen for study as it is capable of classifying more precisely. The Random forest classification algorithm selects specific number of data points from the training dataset and generates decision trees on these subsets of data. While testing the new data points, the prediction result of each decision tree is evaluated and the majority of the result is used as the basis for the final decision Random forest is a classification algorithm that has a number of decision trees; each operating on a particular subset of the dataset taken. The average is found, to improve the accuracy. Random forest is basically based on multiple decision trees. Its depicated in Figure 4.

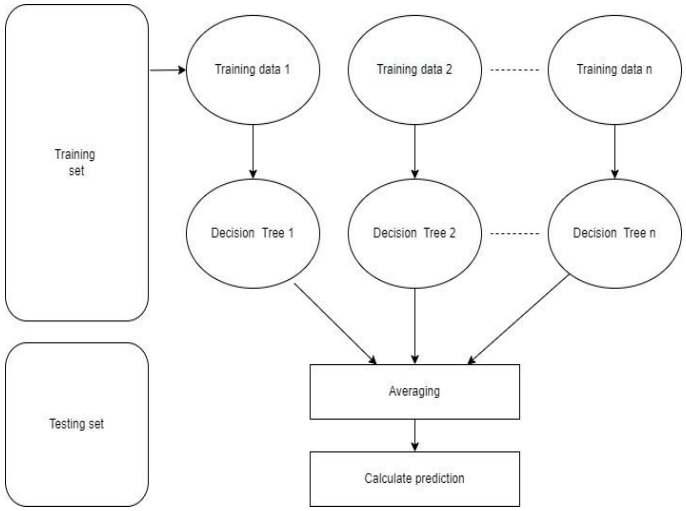

Figure 4: Data Flow Diagram for Classification by Random Forest

Random Forest consumes comparatively lesser time for training, works for larger dataset as well and is capable of managing weakly pre-processed data

The various web services are classified into different ranks as Platinum, Gold, Silver and Bronze numbered from 1 to 4 respectively referring to the different classes of quality. The classification model is trained and tested using machine learning algorithms. The classification model is further used in classifying a service under any of the above four classes.

Web Service Discovery

This phase includes

- Pre-processing and Tokenization of user query and service description.
  The user query is pre-processed by removing the articles, pronouns and other stop-words.
- Calculation of Degree of similarity using
  Keyword mapping

Leacock Chordorow (LCH) algorithm

The service retrieval will be done using the above calculated similarity measures via a proposed LCH based Syntactic cumSemantic discovery algorithm

As depicted in Figure 5, the proposed system is based on a semantic Discovery algorithm that performs matching by computing the semantic similarity using Wordnet's Leocock Chordorow Algorithm in addition to the normal keyword matching.

So, when the similarity value is more than the specified threshold, the service would be retrieved.

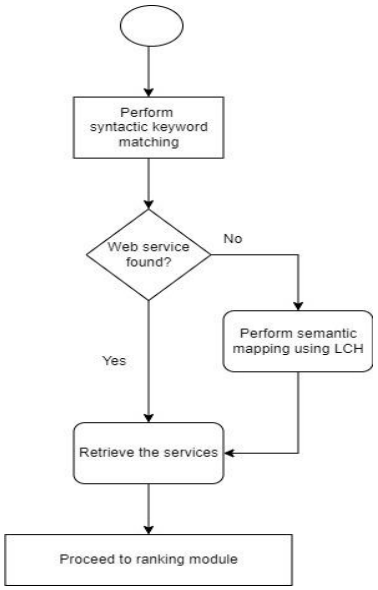

Figure 5: Service Discovery Module

Proposed Algorithm: LCH-based SSDA

The proposed solution uses a semantic discovery algorithm that maps the user query with the service. The algorithm uses a combinational approach that employs both syntactic and semantic matching techniques.

Algorithm: LCH-based Syntactic Cum Semantic

discovery (LCH-based SSDA**)**
user_service-> names of requested service
service_file_names-> the list of available service files
t->threshold
serv_name-> web service name
ont_name-> ontology name

```
1:Count=0;
2: for (all service_name in service_file_names)
3: if (service_name==user_service)
4: OutputList.add(user_service)
5: Count++;
6: end if
7:if(Count==0)
8:sim= lch(user_service, serv_name)
9: if (sim>= t) then
10:OutputList.add(serv_name)
11: end if
12: end if
//second filter
13: if(ont_name==user_service)
14: OutputList.add(serv_name)
15: Count++;
16: end if
17: if(Count==0)
18: sim= lch(user_service, ont_name)
19: if (sim>= t) then
20: OutputList.add(serv_name)
21: end if
22: end if
23: end for
```

Computation Of Semantic Similarity

In order to compute the semantic relatedness, we make use of Wordnet's Leacock Chordorow (LCH). Wordnet is basically a lexical database that contains the semantic relations between words. The relations could be synonyms, hyponyms and meronyms.

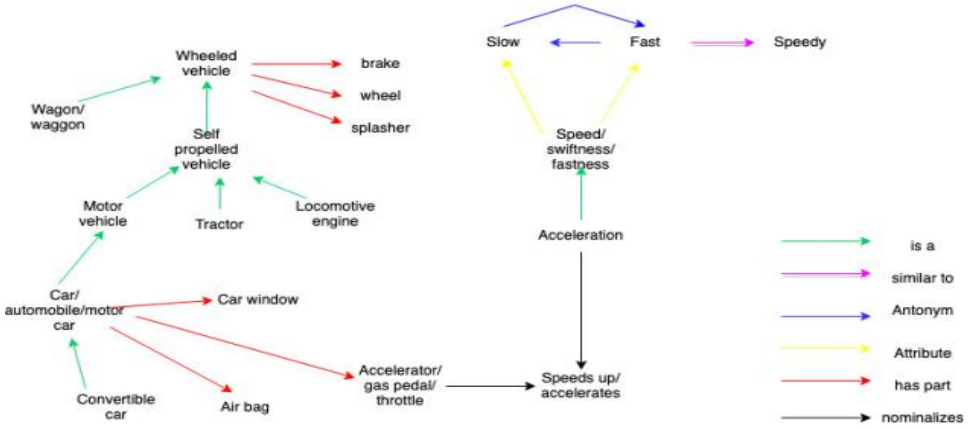

Figure 6:Wordnet Tree

For instance, in the above figure 6, the Wordnet tree structure denotes relation between the different words. LCH is an extended version of Path-based similarity which calculates the shortest distance between the two synonym sets.

LCH uses Wordnet to compute the value of similarity by calculating the negative log of the shortest path between the two synsets (i.e.,instances are the groups of synonyms that denote the same concept) divided by two times the total depth of the taxonomy (D).

Wu Palmer considers onlydepth,whereas LCH considers both the length of the shortest distance as well as the depth of the root of taxonomy.

Parameters for LCH

Max score = InfinityError score = -1.0

Uses all senses and includes root node

$$LCH\ SIMILARITY = -\log(\frac{spath(synset1,synset2)}{2*Depth}) \quad (1)$$

Similarity Calculation using LCH

Example

T1=Htrees(college#n#1)=[1]*ROOT*#n#1< entity#n1<abstraction#n6<group#n1<social_group#n1<body#n2<college#n1
T2=Htrees(student#n1)=
[1]*ROOT*#n#1<entity#n1<physical_entity#n1<causal_agent#n1<person#n1< 110nrolee#n1< student#n1

[2]*ROOT*#n#1<entity#n1<physical_entity#n1<object#n1<whole#n2<living_thing#n1<organism#n1<person#n1<110nrolee#n1<student#n1

Length(entity#n1)=11

MaxiDepth(n)=20

$$Score = -log(\ length(LCS)/(2 * MaxiDepth(LCS.pos)) \quad (2)$$
$$= -log(11/(2 * 20)) = 1.29$$

Thus, the novel LCH based syntactic cum semantic discovery algorithm has been used to map the keywords to the relevant web services.

Ranking Phase

This phase is specific to normal users. The list of discovered services would then be ranked by considering the various attributes pertaining to Quality Of service and Quality of Experience. This will help in saving the user's time for searching the best service from the list. The best ones would be reflected on the top order of the retrieved results

In addition, the proposed system will allow the users to filter and search for web services prominent in a particular location by entering their preferred location. However, this would be optional for the users.

Requirement Specific Retrieval Phase

In general, developers will have knowledge of the expected value of QoS parameters. So, the search mechanism for developers is designed to facilitate them to retrieve services based on their input values of attributes like Response time, Throughput, Success-ability etc. Hence, in addition to performing the

semantic discovery of services, the developers would be capable of filtering the services by their input parameters.

Finally, the efficiency and accuracy of retrieval is evaluated by computing the precision and recall measures. Thus, the proposed system will provide an efficient web service discovery mechanism that will help in refining the search results and in minimizing service consumers' cognitive load during search formulation and execution

The base of the proposed mechanism can be extended further to the wider scope of usage, adding value to the modern AI driven world. The proposed mechanism of calculating the semantic similarity that helps is improving the understanding ability of the search engine can also be used in modern day devices such as Alexa that operate based on recognizing human voice commands. Such devices can make use of the proposed algorithm to be able to not just respond to a set of preconfigured commands but to be able to processs and respond to a different range of commands in a more realistic manner. It can basically calculate the similarity between the human command and its pre fed instruction using this mechanism, draw lines to map and decice what would be the most appropriate action in response. This is one of the most interesting future scope of this work.

4 Experimental Evaluation and Results

For the experimental evaluation of the proposed approach WSDL files downloaded from WS Dream Repository were used as the dataset. The dataset contains WSDL files pertaining to different domains such as banking, ecommerce, business, address lookup etc. The web services were semantically annotated with the related keywords describing the service. Protégé tool is usedto represent the ontology's graphical structure using its Graph Onto or Graph Viz plug in. In order to involve the reflection of the non-functional metrics on these services, the SAWSDL dataset was enriched with QoS attributes such as response time, availability, latency, reliability, success-ability, throughput, best practice and compliance. The application was built using NetBeans IDE in Java language.

Experimental Setup

The annotated keywords were extracted. The preprocessed dataset along with the extracted keywords was stored using WampServer's phpMyAdmin database. To enhance the privacy of users, a secure login and registration portal was developed. Theuser registration and login are authenticated with a random unique id generation and admin acceptance. The user id generated during registration is specific for a particular user and is to be entered every time the user attempts to login. The registration portal has been designed to allow users to register themselves either as a normal user or a developer. This has been done to ensure that cyber criminals and non-authenticated users don't get access to the web service discovery portal and to provide a safe experience to users and service providers.

The Web services were classified into 4 different classes (Bronze, Silver, Gold, Platinum) denoting their performance rating. The dataset is divided into training and testing test. Different machine learning algorithms such as SVM and Random Forestwere applied on the dataset, taking the various QoS attributes as independent variables and the class as dependent variable.

After attempting to classify with multiple ML algorithms, it was found that Random Forest and SVM provided results in classification with relatively higher accuracy. Modeling the data with the classification algorithm would help in predicting the class of a newly published web service based on the value of its QoS metrics. Hence a comparative analysis was performed between SVM and Random Forest to identify which algorithm best classifies a given web service.

The accuracy, precision and recall are calculated by measuring the true positive, true negative, false positive and false negative values using confusion matrix. A confusion matrix is a method of evaluating the performance of a classification ML algorithm which denotes the positive and negative values.

A true positive is the result obtained when the model under study correctly predicts the positive class (i.e.) It implies that both the predicted and actual values are positive. On the other hand, true negative is the outcome where the system correctly forecasts the negative class. This means that both predicted and actual values happen to be negative. A false positive is an outcome where the model incorrectly predicts the positive class. This implies that the predicted value is positive while the actual value is negative. And a false negative is an outcome where the model incorrectly predicts the negative class. This happens when the predicted value is negative but the actual value is positive.

On classifying the web services in the dataset into four distinct categories using Support Vector Machine, the confusion matrix depicted in Table 1 was obtained.

Table 1: Confusion Matrix for Classification by SVM

	Predicted values	
Actual values	Positive	Negative
Positive	59	20
Negative	11	10

Accuracy is calculated as the percentage of correct predictions made by the model used for classification.

$$\text{Accuracy}=\frac{(\text{True positive}+\text{True Negative})}{\text{Number of records in the data}} \quad (3)$$

On applying the above formula, an accuracy of 79% has been attained by SVM.

Using Random forest, the confusion matrix depicted in Table 2 was obtained

Table 2: Confusion Matrix for Classification by Random Forest

	Predicted values	
Actual values	Positive	Negative
Positive	61	5
Negative	3	37

With the obtained values, the performance of the classifier is evaluated by calculating the accuracy. The accuracy of classifying the web services using Random forest is obtained as 89%.

Table 3: Confusion Matrix for Classification by Random Forest

	SVM	**Random Forest**
Precision	84.2	95.3
Recall	85.5	92.4
Accuracy	79	89
Specificity	64.5	92.5

When classified with Random-Forest, the results obtained were

Accuracy: 89%

Precision:0.953

Recall:0.924

Specificity:0.925

So as depicted in Figure 7, we interpret that Random Forest was able to best classify an incoming web service into any one of the 4 categories based on its Quality of service and Quality of Experience metrics.

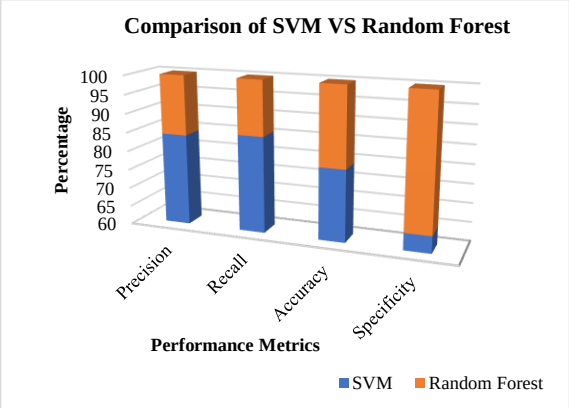

Figure 7: Comparison of SVM VS Random Forest

This was followed by pre-processing the user query where the stop words were removed. The focus keywords were identified and weremapped with the matching services by the proposed novel syntactic cum semantic discovery Algorithm. Thus, the mapped services that have a similarity value greater than a threshold value of (0.4) were retrieved.

Experimental Results

The precision and recall of service retrieval can be used to identify the degree of relevancy of the service discovery process.

In case of Information retrieval, the precision is calculated as the ratio of relevant services retrieved to the overallnumber of services. Recall is the ratio of the relevant services retrieved to the overall number of services returned.

In scenarios dealing with information retrieval, F-measure is used instead of accuracy to compute and denote the degree of relevancy.

The results listed on searching for the various e commerce services available for shopping is shown in Figure 8.

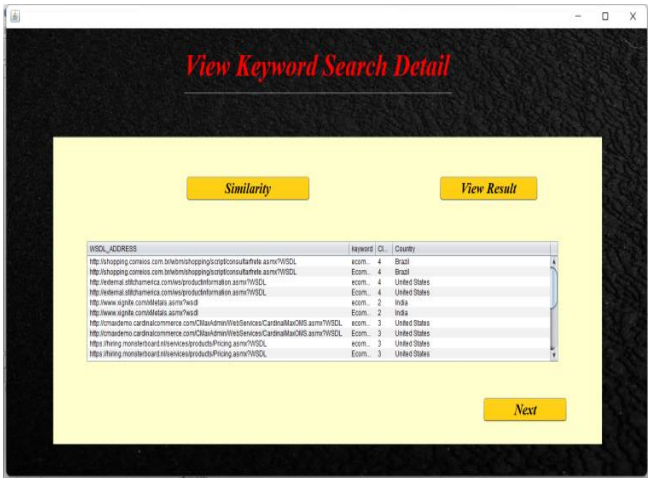

Figure 8: Search Result for E Commerce Scenario

By evaluating the relevancy of search discovery for various other scenarios, the average precision and recall values were obtained as shown in Table.

Table 4: Performance Comparison Chart

Measures	Wu-palmer method	Keyword Matching	LCH-based SSDA
Precision	94.44%	69%	96%
Recall	98.33%	68%	99.4%
F-measure	96.34%	68.49%	97.47%

$$F\ measure = 2 * \{(precision * recall)/(precision + recall)\} = 97.47\% \quad (4)$$

The results of service discovery are evaluated by means of the above-mentioned parameters. When the proposed LCH-based syntactic cum semantic discovery method is compared with the keyword matching technique, the precision is increased by more than 25% and the recall is improved by 30%. On comparing with the Wu Palmer approach, both the precision and recall have shown around 2% increment. Table 4 shows that the precision and recall have shown a significant increase in their values by using theLeocockChordorowfor computing similarity and by using semantically annotated WSDL files.

From the graph depicted in Figure 9, it is clearly understood that the proposed LCH based Syntactic cum Semantic discovery method gives results of higher relevancy than the syntactic approach followed in base paper. The recall value has almost achieved perfection. 96 percent precision and 99.4 percent recall achieved with the proposed system is way higher than the existing counter-approaches. From the performance chart, we could see a huge improvement of the proposed methodology over the syntactic approach. When compared with WuPalmer approach, the improvement is only around 2% to 2.5%.

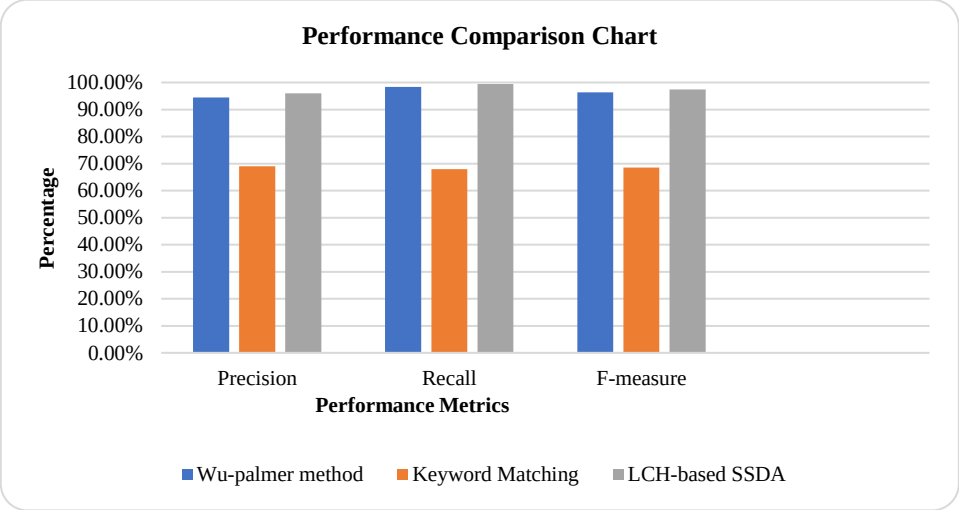

Figure 9: Performance Comparison Chart

However given the complexity and high execution time of WuPalmer mechanism, the proposed algorithm stands out to be the best. The steady improvement in the marginal difference of accuracy using the current approach adds to its credibility. Any means to attempt an even higher accuracy will probably result in a cost driven approach; that would not be preferable for all common levels of use cases.The overall result is efficient than Hungarian algorithm, keyword matching method and the base paper method.The F-measure has increased to 97.47% which is indeed a fair value. This accuracy improvement is the the resultant of the combined impact of both the syntactic and semantic computation of relativity. Hence the proposed method would be more suitable than all the other existing approaches to discover web services effectively. Also, the response time is highly reduced as the proposed approach doesn't use a complex OWL file as in base paper. Instead, semantically annotated WSDL files were used for the experiment.

5 Conclusion and Future Work

Thus the proposed LCH-based Syntactic cum Semantic discovery mechanism for web service discovery using semantic mining helps in efficient web service retrieval in comparison to the other existing methods. Semantic annotation of WSDL files helps in increasing the accuracy of search results and in reducing the time delay caused in case of approaches using complex graphical evaluation. The consideration of relationship among the concepts helps in better discovery of the services. The adoption of Semantic discovery algorithm using LeocockChordorow has been proposed to improve the efficiency of calculating semantic relatedness. The accuracy of service discovery has been found to be higher than the existing approaches. In addition, the classification of services based on QoS metrics has been implemented using machine learning algorithm to help in organized ranking of services and the accuracy of classification has also been found to be high. A separate requirement specific search discovery has been proposed for the developers. Hence the proposed technique of web service discovery will help in improving the quality, relativity and accuracy of search results. In future, the system can be designed to crawl through the services and extract the values of QoS attributes, crawl through the user comments and extract the keywords using POS tagging and NLP techniques. The extracted Quality of Experience metrics could then be used additionally in the process of ranking the services. The functionality of secure login can be further enhanced by encrypting the unique user id and auto mailing the same to the registered mail id. The currently proposed Semantic Discovery Algorithm can be further extended to support IOT enabled environments in the near future.

References

[1] Alshafaey, M.S., Saleh, A.I., & Alrahamawy, M.F. (2021). A new cloud-based classification methodology (CBCM) for efficient semantic web service discovery. *Cluster Computing*, *24*, 2269-2292.

[2] Belhajjame, K., Embury, S.M., & Paton, N.W. (2013). Verification of semantic web service annotations using ontology-based partitioning. *IEEE transactions on services computing*, *7*(3), 515-528.

[3] Cheng, B., Li, C., & Chen, J. (2018). Semantics Mining & Indexing-Based Rapid Web Services Discovery Framework. *IEEE Transactions on Services Computing*, *14*(3), 864-875.

[4] Dantas, J.R.V., & Farias, P.P.M. (2020). An architecture for restful web service discovery using semantic interfaces. *International Journal on Semantic Web and Information Systems (IJSWIS)*, *16*(1), 1-24.

[5] Driss, M., Ben Atitallah, S., Albalawi, A., & Boulila, W. (2022). Req-WSComposer: a novel platform for requirements-driven composition of semantic web services. *Journal of Ambient Intelligence and Humanized Computing*, 1-17.

[6] Fang, M., Wang, D., Mi, Z., & Obaidat, M.S. (2018). Web service discovery utilizing logical reasoning and semantic similarity. *International Journal of Communication Systems*, *31*(10).

[7] Farrag, T.A., Saleh, A.I., & Ali, H.A. (2012). Toward SWSs discovery: Mapping from WSDL to OWL-S based on ontology search and standardization engine. *IEEE transactions on knowledge and data engineering*, *25*(5), 1135-1147.

[8] Huang, K., Zhang, J., Tan, W., Feng, Z., & Chen, S. (2016). Optimizing semantic annotations for web service invocation. *IEEE Transactions on Services Computing*, *12*(4), 590-603.

[9] Kritikos, K., & Plexousakis, D. (2009). Requirements for QoS-based web service description and discovery. *IEEE Transactions on Services Computing*, *2*(4), 320-337.

[10] Kolomeets, M., Benachour, A., El Baz, D., Chechulin, A., Strecker, M., & Kotenko, I. (2019). Reference architecture for social networks graph analysis tool. *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications*, *10*(4), 109-125.

[11] Merin, B., & Banu, W.A. (2020). Discovering Web Services By Matching Semantic Relationships Through Ontology. *IEEE In 6th International Conference on Advanced Computing and Communication Systems* , 998-1002.
[12] Mohanty, R., Ravi, V., & Patra, M.R. (2010). Web-services classification using intelligent techniques. *Expert Systems with Applications*, *37*(7), 5484-5490.
[13] Moradyan, K., & Bushehrian, O. (2015). Web Service Matchmaking based on Functional Similarity in Service Cloud. *Journal of Computing and Security*, *2*(4), 257-270.
[14] Natarajan, B., Obaidat, M.S., Sadoun, B., Manoharan, R., Ramachandran, S., & Velusamy, N. (2020). New clustering-based semantic service selection and user preferential model. *IEEE Systems Journal*, *15*(4), 4980-4988.
[15] Nguyen, T.T.S., Lu, H.Y., & Lu, J. (2013). Web-page recommendation based on web usage and domain knowledge. *IEEE Transactions on Knowledge and Data Engineering*, *26*(10), 2574-2587.
[16] Paliwal, A.V., Shafiq, B., Vaidya, J., Xiong, H., & Adam, N. (2011). Semantics-based automated service discovery. *IEEE Transactions on Services Computing*, *5*(2), 260-275.
[17] Rodriguez-Mier, P., Pedrinaci, C., Lama, M., & Mucientes, M. (2015). An integrated semantic web service discovery and composition framework. *IEEE transactions on services computing*, *9*(4), 537-550.
[18] Sandhya, S., Pabitha, P., & Rajaram, M. (2014). 'Enhanced semantic Web service discovery using machine learning on mapped WSMO services. *International Journal of Engineering &Technology*, *6*(2), 982-991.
[19] Upadhyaya, B., Zou, Y., Keivanloo, I., & Ng, J. (2015). Quality of experience: User's perception about web services. *IEEE Transactions on Services Computing*, *8*(3), 410-421.
[20] Yang, W., & Zhu, Y. (2020). A verifiable semantic searching scheme by optimal matching over encrypted data in public cloud. *IEEE Transactions on Information Forensics and Security*, *16*, 100-115.
[21] Zheng, G., & Bouguettaya, A. (2009). Service mining on the web. *IEEE transactions on services computing*, *2*(1), 65-78.

Authors Biography

J.Brindha Merin is a Asst. Professor(Sr.G) in B.S.Abdur Rahman Crescent Institute of Science and Technology. She did her B.E from AnnaUniversity in Computer Science and Engineering. She completed her M.Tech. from Anna University. She is pursuing her Phd in the field of Data mining and Web Services. She has 11 years of experience in teaching and years of experience in industry.

Dr.W. Aisha Banu is a professor in B.S.Abdur Rahman Crescent Institute of Science and Technology. Her qualifications are as mentioned. Ph.D.(Computer Science and Engineering) from AnnaUniversity,M.E. (Computer Science &Engg.) from AnnaUniversity, B.E. (Computer Science &Engg.) from MadrasUniversity. She has 25 years of teaching experience and her areas of interest include Big Data, Cloud Computing, Social Network Analysis and Information Retrieval Total number of research publications are 30.

K. Fahima Sanobar Shalin is a student pursuing her Bachelor's degree in Computer Science and Engineering at BS Abdur Rahnman Crescent Institute Of Science And Technology. Her deep curiosity to explore the wide spectrum of the ever-evolving Digital world, has always kept her inclined towards programming, data engineering and Data analytics.