International Journal of Engineering & Technology, 7 (2.4) (2018) 182-186

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Service identification using k-NN machine learning

Travis Joseph Poulose [1] *, S. Ganesh Kumar [1]

[1] *Assistant Professor Department of Computer Science Engineering, Kattankulathur Campus, SRM Institute of Science and Technology*

**Corresponding author E-mail: tjpoulose03@hotmail.com*

Abstract

Web service categorization is a daunting task since it requires semantic descriptions of those services which are not provided to the majority of those websites. The proposal of a Semantic based automated service discovery requires a request from the user that can be analyzed which then provides the user with a list of related webs services based on the request that instigated the search. The problem with these service categorizations listed in the Universal description Discovery and Integration (UDDI) is the way the information is related to one another. The relations follow a syntactic method. Semantic based service descriptions is necessary for accurate web categorization. With the help of machine learning we can also predict the user's service request automatically based on previous searches and also select the best web service for a particular request that the user has made using a k-nearest neighbor algorithm. By doing this we can distinguish between the various types of user requests, provide services that are suitable for that particular request as well as suggest other services that might potentially suit the needs of the user.

Keywords: *UDDI; Web Service; Clustering; Machine Learning; K-Nearest Neighbor.*

1. Introduction

The concept of machine learning is simply the ability for a computer to learn without being explicitly programmed to. Its applications are endless and range from information retrieval to brain computer interfaces. Machine learning is quickly being implemented in many of the applications that we are using in our day to day lives and continues to expand in other fields as well. Building software that can learn from past experience is the objective of machine learning and this can be accomplished in a number of ways. Data can be analyzed to determine patterns and frequent occurrences which then in turn can build a reasonable assumption about something. It is important to understand that machine learning is more closely related to data mining than it is to artificial intelligence even though the two are connected. Building better and more accurate machine learning software helps programmers get closer to the ultimate goal of building a realistic self thinking machine.

Web Service Discovery and Semantic Relations

Understanding the main difference between a webs service and a web site is relatively simple. A website is simply a set of web pages that are under some domain that was defined by the creator of the website. Websites are static and are only for human consumptions and don't require the use to do anything but gather or locate information that can found on the internet. As for the a web service it is not static like a website, in fact it is completely dynamic and it is not meant for human consumption rather it is meant for the consumptions of other websites. A good example would be a flight booking website that will use a third party web application that is known as a web service to collect data such as the flight timing of all relevant flights based on the search parameters that was entered by the user. These flight timings will then be gathered and displayed on a static website for the user consumption. However collecting data and filtering the relevant information from web services is not that simple since it requires the semantic meaning of various keywords that a web service may contain in its WSDL document. This proves to be a very tedious task since we are not only focusing on the syntactic representation of a word but also the semantics meaning of the word which cannot be determined by just analyzing the string. One of the important areas in the proposed work understands the semantic description of the words that are relevant in the WSDL document of the web service. Therefore if a user was to provide the system with keywords, these keywords will be taken as the parameter for the discovery system and provide the user with the most relevant list of web services that pertain most closely to the search parameter keywords that the user has made as the input. There are many applications to the web service discovery system since it can also provide the relations between words using their semantic relations as opposed to the more conventional approach which is retrieving words that are syntactically related. Combing these concepts with machine learning techniques provide the user with a more personalized approach to the common web discovery process. With the help of machine learning the user can now input search queries which are then stored and used as training data which is then used to help determine the tendencies of the used and can even predict future search queries and provide the user with suggestions based on the queries that were already instigated. This concept of the web service discovery process will further expressed in this paper.

2. Related work

Previous research papers have also dived into this area of web discovery and machine learning techniques however the proposed system involves both concepts. In the related work the, web service discovery and semantic description was determined. The proposed system was divided into three main categories as shown in Fig 1. The first was categorization of the web services in the

UDDI. It was made sure that the categorization process respected the semantic definition of the words used in the web service WSDL rather than consider the syntax of the words. The second step was selection of services based on the user request which used a semantic similarity based matching approach. This approach analyzed the user query and further refined it so that it can retrieve more accurate web services for the user. The final step was the retrieval of the web services after the selection process has been executed.

The web service retrieval process involved calculating various parameters such as the relevance, specificity and span. The appropriate services are selected from the UDDI based on these three parameters meaning the higher the value of the three parameters the more relevant the service is to the user's query. The categorization of the web services use a tiered ontology framework to organize each of its concepts and how they are related to one another.

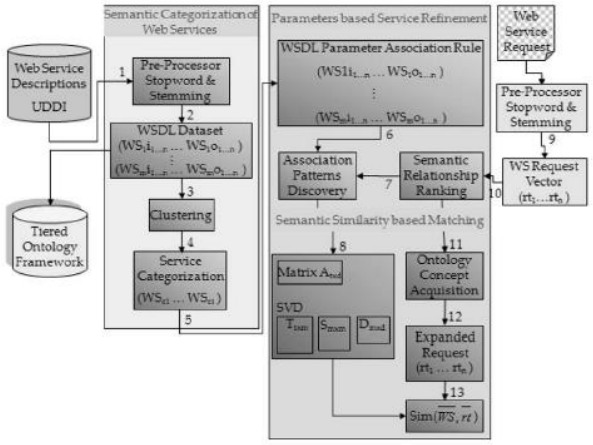

Fig. 1: Architecture of the Web Service Discovery Process in the Related Work.

This paper makes use of the clustering concept and when the categorization of web services takes place each individual web service is then represented as a vector. This vector will hold all the key elements of that particular web service or the service description. This is known as the Service Description Vector (SVD). When the webs services are then grouped further into web services groups. The services are grouped based on the service functionality rather than the category the service is listed under the UDDI.

Categorization of the Web ServicesIn this paper the first step is the organization of the web services listed in the UDDI. This is done by extending ontology concepts and providing the system with a hierarchical clustering methodology. Therefore the service description vector of a particular web services continues to become more and more refined as relevant ontology concepts are added to the vector and at the same time irrelevant and redundant elements are removed from the vector. Keep in mind that the description and keywords that are added to the vector are all semantically related to the function of the web service as opposed to the conventional syntactic relations of words and the elements. The categorization of the web services consists of three stages. 1) Determining what is to be put in the service descriptor vector for each of the web services. 2) Extending the vector with relevant ontology terms and keywords and removing all the redundant and irrelevant key words so that only the functional words are selected using the semantic ranking based system. 3) Creating the clusters of the web services based on the hierarchical of the upper ontology tier so that retrieval of these web services can be taken from the sub-clusters.

Web Service Vector before the Web Service Vector can be determined for a particular Web Service; the respected WSDL document of the particular Service should be extracted and analyzed. The analysis process includes determining the keywords under the <documentation> and <element name> tag so that the important elements can be found and used for the Vector. Also further refinement takes place such that punctuation and redundant words are filtered out of the WSDL document since they hold no value or meaning. The spaces between the words in the document is considered to be the delimiter the end result is a list of important words and elements that can potentially be used in the Web Service Vector. Sumo mappings take place and the software of WordNet is used to negate words like synonyms and combine the ontology terms to the terms in the WSDL document that was filtered out. The SUMO mappings as well as the WordNet software are then used to determine the nouns of the document and associate them to relevant ontology concepts. So in the end the web service is finally given with the a list of relevant terms and nouns and concepts that accurately describe the web service. Therefore when a query is instigated by the user to retrieve a web service, the query is taken as a parameter is is mapped against all the possible web service vectors that are available in the UDDI so that the most relevant web service is retrieved for the user to consume.

3. Overview of proposed approach

In the recent paper that was discussed in the previous section, similar steps were taken. The categorization of the UDDI through a clustering method as well as organizing the data based on the semantic relationships rather than the syntactic representation of the data. After this there is the retrieval of the web services based on the query that was instigated by the user. The proposed system is just an extension of the previous paper. The use concept of machine learning is taken into consideration. Machine learning is implemented in the user search query stage. For every search query that is made by the user in the system, all queries will then be taken as training data to better the personal response from the system the next time the user searches for another Web Service. The machine learning algorithm used is the k nearest neighboring algorithms and the system will provide more and more accurate suggestions the more the user utilizes the system and searches. Every user will have his/her own profile so that the training data is different for every user. The profiles will match the tendencies of the specific user using the engine. As the user continues to search for the services the parameters for the search gets collected and this will be used to determine the future tendencies of the user using k nearest neighbor algorithm. Fig 2 displays the architecture of the system, which includes the web categorization, service retrieval, and the additional step of the machine learning implementation. The algorithm is further optimized by displaying the concepts on a graph and the mapping and the relationships of these concepts using such metrics such as precision, recall and f-recall. These parameters can determine how closely related theses particular concepts are from one another using the k nearest neighbor algorithm. The distance of the concepts using the ontology that has been created and continues to expand for every new concept that is uploaded in the system, helps with the distance calculations between the concepts. The formula that is used us a basic Euclidean formula. Parameters are taken from each of the concepts and are compared with each other to see how far or close they are together in the ontology that was generated. The distance will determine the similarity parameters such as the precision and the recall as mentioned earlier.

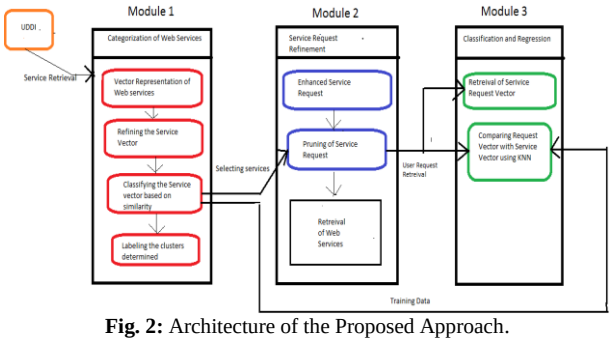

Fig. 2: Architecture of the Proposed Approach.

K-Nearest Neighbor Algorithm used in Web Service

Discovery-NN is one of the most commonly used algorithms that is used when dealing with machine learning concepts. It is so effective because it follows a classification method that enables the user of the algorithm to analyze various forms of data and then classify them based on the similarity using the distance between these concepts. This becomes even more useful for the proposed approach because we are dealing with web services and the ontology in which they reside. These web services are also related in some way and the strength of the relationship between these concepts is based on the distance between them in the ontology. Therefore the use of a K nearest neighbor algorithm is a best fit for this type of implementation. In addition to the relevance of this method for this particular project the implementation of this algorithm is simple and easy to follow. The main objective for the proposed project is defining and classifying the services based on the function of the service rather than any other parameter. This means that the all the concepts that were inserted in the ontology will be accurately places and the use of the K nearest neighbor algorithm will work more effectively. Also by enhancing the service request as well we in turn receive a better matching with relevant services.

K-NN is often used when people are trying to search for concepts and elements that are very closely related with one another. When two or more concepts are more closely related to one another that means that the distance between those concepts is less. However before making the comparison between the concepts there is another task that must be performed so that the Euclidean formula can be executed. There must be a vector representation of each of the web services. This vector representation will define all the concepts and elements that describe the web service. The parameters for the web service can then be used as the variables for the Euclidean formula to determine the appropriate distance the services. A Concept Search is when we try to search these web services based on the semantic representation of them. Concept Searches are very useful in the real word since it helps with many companies such as legal companies which include law suits. For example all emails and relevant data must be gathered for a particular case when it comes to a lawsuit to ensure appropriate actions towards the defendant or whoever is involved in the case. When a system is dealing with model that cannot be accessed by humans then the K-NN algorithm is extremely useful since it requires only few parameters that must be defined and more importantly it is highly accurate and does not require a human readable model. Since this algorithm relies mainly on the distance between the concepts, it is very important to make sure that relevant information about the elements involved in algorithm is readily available. This information is important since it is needed to provide the measurements for each of the elements. Although the K-NN algorithm is a lazy learner algorithm and sacrifices computation time, it is still effective for this system since is accurately provides relationships between concepts. A basic understanding of the K-NN algorithm is displayed in fig 3.

```
k-Nearest Neigbour
Classify(X,Y,x)//X:trainingdata, Y:class labels of X, x:unknown sample
for i=1 to m do
Compute distance d(Xj,x)
end for
Compute set I containing indices for the k smallest distances d(Xj,x).
return majority label for {Yi where i ϵ I}
```

Fig. 3: Pseudo Code of the K nearest Neighbor Algorithm

Stemming and Stopping of elements before the web application can include a service into an ontology, the service must be refined. This include removing all redundant words and synonyms and refining the elements of the document in further my including the root words where necessary so that the true meaning of the words are taken into consideration rather than just analyzing the elements as string. It is important to understand that the system must follow semantic guidelines to endure that the ontology and refinement of the ontology remains consistent so that user will receive the most relevant service based on the request that was ensued. When a service is uploaded into the server, the entire document is never fully accounted for before entering the server and ultimately being part of the ontology. The document must go through two main filtering techniques before it is deemed fit to enter the ontology creation process. These two procedures are the stemming and stopping filtering techniques. Stemming is the first step of the procedure. Once the document is accessed and initially uploaded into a server, the document is then analyzed word for word. The entire static website is converted into a text file, which then analyzes every word. It important to list all the words in the document before proceeding, this ensures that no word is left out or excluded during the next phase of the filtering process. As every word is analyzed, the stopping elements in the document are removed. These elements include and are not limited to “the”,”and”,”or”,”on”,” at.” The stopping words include all prepositions and transitional words. These words are not considered for evaluation due to the lack of importance to the main ideas and functional relevancies of the web document. After the first filtering process takes place the list of words are much shorter and refined and easier to identify and deal with in the next area of the filtering process. The stemming filter included determining the root words for the listed words. It is important to keep in mind that not all words will have a root word but for all the main conceptual words there will be words that have root meaning. The root word gives us meaning in the dictionary. When we are considering the root word we consider the etymology of the word which then gives us a basic understanding of the word, where it came from and what it means. This is done by using and third party application called WordNet. WordNet is an application that words the user with a list of words in the English dictionary which include the definition the etymology and other important facts about each individual word. The WordNet application is used during the filtering process in order to remove redundant words and to display the root word for the stemming step of filtering. Fig 4. Shows how filtering occurs in both the stemming and stopping steps.

Topical Terms Cataloging Use For: Cataloguing Created: Broader Terms: Information organization Technical services (Libraries) Narrower Terms: Library catalog management Names, Personal (Cataloging) Shelflisting Copyright cataloging Collection level cataloging Multiple versions (Cataloging)	Cataloging Cataloguing Information organization Technical services Libraries catalog management Names Personal Cataloging Shelflisting Copyright cataloging Collection	service manag title uniform recatalog catalogu copyright convers cooper level descript book person version catalog technic librari

Fig. 4: Filtering Process from the Dataset to the Set without Stop-Ping Words and Finally to the Set with Only the Root Words.

Figure 4. Filtering process from the dataset to the set without stopping words and finally to the set with only the root words.

Ontology Creation It is important to make use of an ontology so that we can relate the endless amounts of concepts and functions related to the services with each other. That is what an ontology basically is. It related all relevant in a pool with each other. This gives us a conceptual idea of how everything is related to everything. However the construction of an ontology is not that simple since we focus on the semantic and functional descriptions of the services rather than the syntax. That is why the stemming and stopping filtering is important instead of analyzing the entire document as a whole. If the latter was done then there would be noise in our data and that would result to inaccurate ontology creation and refinement in the following steps. Ontology will create classes, each of these classes will consist of objects. These objects in this context will be the services. Ontology generation is impossi-

ble without ontology services. One of the many services includes OWL (Web Ontology Language). In this proposed research paper OWL will be used to assist the generation of the ontology. OWL is one of the few services provided in the Semantic Web technology stack (W3C's) which include RDF, RDFS and SPARQL. OWL provides the user with communication between the services and the user will be able to interact with other ontology. In this paper the ontology that is generated is based on the filtering of words from the stemming and stopping steps of the documents. Therefore when the first service is uploaded into the local directory an ontology is created for the first time. As more and more services are uploaded and searched by the user the ontology is more and more refined and implements further elements into the ontology, therefore becoming much larger and complex for every search or upload that is instigated. The OWL creation uses programming language and classes as well as objects to organize the data and make relations with one another. Fig 5 shows the snippet of how the ontology is created and how classes and the objects are placed in the classes.

```
<? Xml version="1.0"?>
<rdf:RDF
xmlns="http://a.com/ontology#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xml:base="http://a.com/ontology">
<owl:Ontology rdf:about=""/>
<owl:Class rdf:ID="inform">
<rdfs:subClassOf>
<owl:Class rdf:about="# multiple"/>
</rdfs:subClassOf>
</owl:Class>
<owl:Class rdf:ID=" copyright">
```

Fig. 5: OWL Programming Language to Label Data and Organize Them into Classes.

Mapping and Search Analysis

After the search parameters are given by the user, the system will provide the user with a list of web services that are related to the search that was instigated. This is the part of the system where the machine learning aspect comes into play. In order to determine the relationship between the search parameters and the web services associated with the search keywords a mapping mechanism should take place. In order to map the two concepts with one another, there are certain parameters that first must be determined in order to visually understand how close the two concepts are actually related with one another. There are three main calculations that must be determined which are precession, recall and F-recall. Precession involves calculating the percentage of the true positives in a pool of all positives, while the recall can be calculated by determining the percentage of true positives in a pool of both true positives and true negatives. True positives in the case of are all the elements in search that are that are related to the elements in the webservices that are retrieved. The true positives are then collected and the percentage is then determined. This is done with the true positives elements out of all the elements in the pool. These elements are already determined when the search query became part of the overall ontology. The elements in of the web service has also been part of the ontology that was created, therefore the precession and the recall can then be calculated more efficiently. A graph can be used to visually represents how closely related the service is with the search that was instigated by the user. In the final representation of the relationship we can then map the recall with the precession as displayed in figure 6.

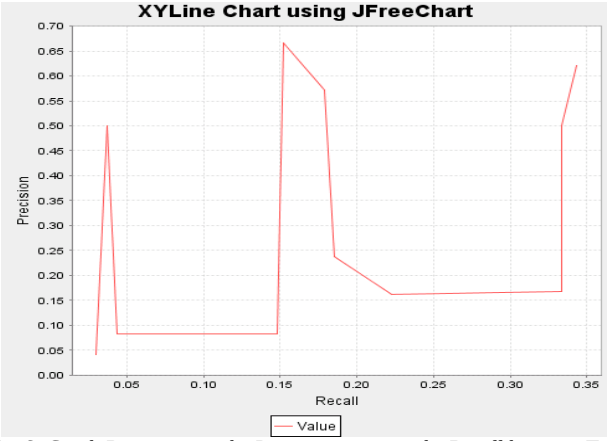

Fig. 6: Graph Representing the Precession against the Recall between Two Concepts.

4. Future work

As mentioned earlier the discovery of web services involve semantically making the relation between the concepts. Avoiding all syntactic representation of data is frowned upon since it does not provide the ideal functionality of the service. In the proposed paper as well as related work we provide that type of information analysis, and try to determine the meaning of words rather than just analyzing every word as just a string and then making the comparison. In this paper we looked into the area of ontology creation and the discovery of web services, we have also developed a systematic method of creating a ontology when a service is being uploaded. Future work involve developing a more refined version of the ontology with more accurate depictions of elements, objects and the classes they belong to. Creating a more refined version of the ontology will in turn provide better analysis when implementing the machine learning techniques to the data which is not limited to classification algorithms such as K-NN algorithm. We can further expand the concept that has been implemented which can involve providing suggestions to the user for every search that has been instigated. This means that the user can search for the relevant services without actually having to type the parameters. For every search a user can make, more refined results can be displayed which are directly relevant to the user's tendencies. This type of procedure might involve more accurate and stable machine learning techniques which don't requires so much data as its parameters. Therefore the scope for the discovery of web services continues to expand in this current generation of computer science and technology. Machine learning technique can be found in various social media outlets which include Facebook and Youtube. These are the type of social media applications that continue to expand their methods on machine learning techniques to further understand their users tendencies likes and dislikes to certain content.

5. Conclusion

After doing a survey on various papers involving document clustering and semantic analysis with the help of machine learning It is clear that a classification and regression approach may be most suitable for the system at hand. By determining and ranking the characteristics of web services the request can be taken and compared to the data set of services using a k nearest neighbor algorithm so that the most suitable services will be generated for the respected request. Previous searches can then be stored for future service request enhancement and more accurate and relevant services can be retrieved when more service requests are taken from the user. Web Services can then be compared to other services in the system through mapping and parameters which involve precession and recall. These are the type of parameters that can be used to display a visual representation of how the services are

related to one another conceptually and functionally wise. The mapping procedure utilizes a classification algorithm to better understand the closeness or distance between the services using the ontology that include both concepts. Ultimately the end result is the provision of the related services based on the search parameters that was given by the user. The discovery of web services has been accomplished and comparisons between services and can be done by mapping its relations using an ontology that was created using a Web Language (OWL)

References

[1] Jian Ma, Wei Xu, Y.-H. S. E. T. S. W. and Liu, O. (2012). "An ontology based mining system." IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICSâˆ˜A ˇ TPART A: SYSTEMS AND HUMANS, 42.

[2] Aviv Segev, M. and Quan Z. Sheng, M. (2012). "Bootstrapping Ontologies for web services." IEEE TRANSACTIONS ON SERVICES COMPUTING, five.

[3] Tamer Ahmed Farrag, A. I. S. and Ali, H. A. (2013). "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).

[4] Danushka Bollegala, Y. M. and Ishizuka, M. (2011). "Automatic discovery of personal name aliases from the web." IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, 23.

[5] Abinayal, G. and Jaishanthu, B. (2014). "Equipped search results using machine learning from web databases." ICICES.

[6] Marouane Kessentini, Hanzhang Wang, J. T. D. and Ouni, A. (2009). "Improving web services design quality using heuristic search and machine learning." IEEE 24th International Conference on Web Services, 40, 883–891.

[7] Jian Wang, Panpan GAO, P. G. Y. M. K. P. C. K. H. (2017). "A web service discovery approach based on common topic group's extraction." IEEE.

[8] Rupasingha A. H. M. Rupasingha, I. P. and Kumara, B. T. G. S. (2015). "Calculating web service similarity using ontology learning with machine learning." IEEE.

[9] Aabhas V. Paliwal, Basit Shafiq, M. J. V. H. X. N. A. (2012). "Semantic based automated service discovery." IEEE.

[10] Payal A. Jadhav, D. P. N. C. and Wagh, K. P. (2016). "Integrating performance of web service engine with machine learning approach." IEEE.

[11] S. Ganesh Kumar, K Vivekanandan(2017) , "Intelligent Model View Controller Based Semantic Web Services call through Mishmash Text Featuring Technique" in Journal of Computational and Theoretical Nanoscience.