International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue (6) NCRTCST-2013, ISSN 2249–071X

National Conference on Recent Trends in Computer Science and Technology (NCRTCST)-2013

Classification of Web Services Using JForty Eight

Venkataiah Vaadaala
Department of CSE,
CMR College of Engineering & Technology, Hyderabad, India
Email: venkat.vaadaala@gmail.com

Dr. R. Rajeswara Rao
Department of CSE,
Jawaharlal Nehru Technological University, Kakinada, Vizayanagaram,
Email: raob4u@yahoo.com

Venkateswara Rao .K[3]
Department of CSE,
CMR College of Engineering & Technology, Hyderabad, India
Email: vrkatevarapu@gmail.com

***Abstract* – Web Services are emerging technologies that enable machine to machine communication and reuse of services over Web. They have innovative mechanism for rendering services over diversified environment. They promise to allow businesses to adapt rapidly to changes in the business environment and the needs of different customers. The rapid introduction of new web services into a dynamic business environment can adversely affect the service quality and user satisfaction. Consequently, assessment of the quality of web services is of paramount importance in selecting a web service for an application. In this paper, we employed well-known classification model decision tree (J48) to predict the quality of a web service based on a set of quality attributes. The experiments are carried out on the QWS dataset. We found that web-service relevance function is most significant attribute in determining quality of a web service. The experiments results shown in this paper are about classification accuracy, sensitivity and specificity. The results in the paper on this dataset also show that the efficiency and accuracy of J48.**

***Keywords* – Web Services, Quality of Services, Decision Tree (J48), WSRF (Web Service Relevance Function), Data Normalization.**

I. Introduction

"A Web service is a software system designed to support interoperable machine-to-machine interaction over a network, loosely coupled, distributed and platform independent services. They are platform and language independent, which is suitable for accessing them from heterogeneous environments. With the rapid introduction of web-services technologies, researchers focused more on the functional and interfacing aspects of web services, which include HTTP and XML-based messaging. They are used to communicate across by using open standards such as HTTP and XML-based protocols including SOAP, WSDL and UDDI [2][9]. WSDL is a document that describes the service's location on the web and the functionality the service provides. Information related to the web service is to be entered in a UDDI registry, which permits web service consumers to find out and locate the services they required. Using the information available in the UDDI registry based on the web services, client developer uses instructions in the WSDL to construct SOAP messages for exchanging data with the service over HTTP attributes [10].

The main objective of the paper is to develop classification model based on J48 to predict the quality of web services based on the number of QoS attributes. The most significant application of the developed model is that we can confidently predict the quality of a new web service (which is in the training set) given its QoS attributes.

For illustration purpose, we take the training set consists different web services. Now, any user will choose one of the web services that has higher value as measured by the QoS attributes, which are essentially non-functional in nature. In this context, if one develops classification model based on J48 to classify the given web service, then the user can use this ranking in order to select a web service.
In this paper, we employ, we focus on the data classification and the performance measure of the classifier algorithms based on TP rate, FP rate generated by the algorithms when applied on the data set [1][2].

Classification analysis is the organization of data in given classes. Also known as supervised classification, the classification uses given class labels to order the objects in the data collection. Classification approaches normally use a training set where all objects are already associated with known class labels. The classification algorithm learns from the training set and builds a model. The model is used to classify new objects. The rest of paper is organized as follows: Section II describes, in detail, the quality related issues in web services. Section III presents the methodology followed in this paper. Section IV Measuring the Performance. Section V presents the experimental work and results Section VI conclude the paper.

II. QoS for Web Services

The active e-business visualization calls for a flawless combination of business processes, Web services, and applications over the Internet. Carrying out QoS on the Internet is a vital and major challenge because of its vibrant and changeable nature. The dynamic electronic business idea requires a perfect arrangement of business procedures, web-services, and functions on the web. Implementing quality of service on the web is an essential and main test due to its exciting and variable character.

QoS concludes a comprehensive selection of processes that are comparable to the needs of service-requester with those of the service-publisher on the basis of the network properties available.

International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue (6) NCRTCST-2013, ISSN 2249–071X

By QoS, we talk about not ingenious configuration of web-services like reliability, ease of use, performance and security methods. The table below gives the different non-functional attributes of web services and their units. The attributes of the model in Table are almost similar to the attributes of QWS Dataset used in this paper.

A. About the Dataset

The updated QWS Dataset Version 2.0 includes a set of 2,507 Web services and their QWS measurements that were conducted in March 2008 using our Web Service Broker (WSB) framework. Each row in this dataset represents a Web service and its corresponding nine QWS measurements (separated by commas). The first nine elements are QWS metrics that were measured using multiple Web service benchmark tools over a six-day period. The QWS values represent averages of the measurements collected during that period. The last two parameters represent the service name and reference to the WSDL document [8].

B. Data Normalization

Mostly, all of the quality of service constraints varies from one another in direction as well as in value range of the utility increments. There is no comparison between them. Therefore, calculation of the weighted average of quality of service constraints is not useful. Constraint values must be transformed such that they reflect the true value in a standard range and also providing the same incrementing direction. Let's say that raw value of constraint, Q, is denoted by q, threshold value is denoted by qth and qmin denotes the minimum [3].

Data normalization of a constraint is calculated according to equation (1) if the effectiveness of it increases with the value of the constraint, q. Or else, equation (2) is applied.

$$Q' = \frac{(q-qmin)}{(qmax-qmin)'} \text{ if } qmax\text{- } qmin \neq 0 \qquad (1)$$

$$Q = \frac{(qth-q)}{(qth\ \ qmin)'} \text{ if } qth\text{- } qmin \neq 0 \qquad (2)$$

On the whole, final rank value WSRF (web service relevancy function) is calculated using weighted sum of the quality of service constraints which were normalized, according to equation (3) and we get dataset.

$$V=\sum_{i=0}^{n} Wi\ X\ Qi \qquad (3)$$

C. Service Classification

The service classification characterizes different levels of service contributing qualities. There are four service classifications:

1. Excellent (High quality)
2. Good
3. Average
4. Poor (Low quality)

The classification is differentiated on the on the whole quality evaluation calculated by WsRF. Using WsRF values found for every Web service, we apply a classification format to relate each Web services to a particular service group. The classification can be useful to distinguish between ranges of services that offer the similar functionality. The part of the dataset is shown in Table 2.

III. Methodology

Decision tree algorithm J48:

J48 classifier is a simple C4.5 decision tree for classification. It creates a binary tree. The decision tree approach is most useful in classification problem. With this technique, a tree is constructed to model the classification process. Once the tree is built, it is applied to each tuple in the database and results in classification for that tuple[1][4].

```
Algorithm [1] J48: INPUT:
                D    //Training data

                OUTPUT

                T    //Decision tree
            DTBUILD (*D)

  {
 T=φ;
 T= Create root node and label with splitting attribute;
T= Add arc to root node for each split predicate and label;
            For each arc do
 D= Database created by applying splitting predicate to D;
 If stopping point reached for this path, then
 T'= create leaf node and label with appropriate class;
 else
 T'= DTBUILD(D);
  T= add T' to arc;
  }
```

While building a tree, J48 ignores the missing values i.e. the value for that item can be predicted based on what is known about the attribute values for the other records. The basic idea is to divide the data into range based on the attribute values for that item that are found in the training sample. J48 allows classification via either decision trees or rules generated from them [4][5].

IV. Measuring the Performance

The performance of classification algorithm is usually examined by evaluating the accuracy of the classification. However since classification is often a fuzzy problem, the correct answer may depend on the user. Traditional algorithms evaluation approaches such as determining the space and time overhead can be used but these approaches are usually secondary. Determining

International Journal of Electronics Communication and Computer Engineering
Volume 4, Issue (6) NCRTCST-2013, ISSN 2249–071X

National Conference on Recent Trends in Computer Science and Technology (NCRTCST)-2013

which better best is depends on the interpretation of the problem by users.

Classification accuracy is usually calculated by determining the percentage of tuples placed in a correct class. This ignores the fact that there also may be a cost associated with an incorrect assignment to the wrong class. This perhaps should also determine [1][5].

A. Confusion matrix:

A confusion matrix illustrates the accuracy of the solution to a classification problem. Given n classes a confusion matrix is a m x n matrix, where $C_{i,j}$ indicates the number of tuples from D that were assign to class $C_{i,j}$ but where the correct class is C_i. Obviously the best solution will have only zero values outside the diagonal [1].

A confusion matrix contains information about actual and predicted classifications done by a classification system. Performance of such systems is commonly evaluated using the data in the matrix. The following table shows the confusion matrix for a two class classifier.

The entries in the confusion matrix have the following meaning in the context of our study [5]:

1. *a* is the number of correct predictions that an instance is negative,
2. *b* is the number of incorrect predictions that an instance is positive,
3. *c* is the number of incorrect of predictions that an instance negative, and
4. *d* is the number of correct predictions that an instances positive [6].

Table 1: QWS Attributes and Units

ID	Parameter Name	Description	Units
1	Response Time	Time taken to send a request and receive a response	Ms
2	Availability	Number of successful invocations/total invocations	%
3	Throughput	Total Number of invocations for a given period of time	Invokes/second
4	Success ability	Number of response / number of request messages	%
5	Reliability	Ratio of the number of error messages to total messages	%
6	Compliance	The extent to which a WSDL document follows	%
7	Best Practices	The extent to which a Web service follows	%
8	Latency	Time taken for the server to process a given request	Ms
9	Documentation	Measure of documentation (i.e. description tags) in WSDL	%
10	WsRF	Web Service Relevancy Function: a rank for Web Service Quality	%

Table 2

S. No	Response Time (Ms)	Avail-ability (%)	Throu-ghput (%)	Sociability (Invokes/ second)	Reliability (%)	Compliance (%)	Latency (Ms)	Documen-tation (%)	WsRF (%)	Classi-fication
1	332.4	89	1.4	96	73	78	2.6	96	0.00053	POOR
2	302.75	89	7.1	90	73	78	187.75	32	0.00051	AVERAGE
3	482	85	16	95	73	100	1	2	0.000713	AVERAGE
4	128.17	98	12	100	67	78	22.77	89	0.00129	GOOD
5	107	87	1.9	95	73	89	58.33	93	0.000911	AVERAGE
6	107.75	80	1.7	85	67	78	18.21	61	0.000803	AVERAGE
7	225	98	1.3	90	67	100	40.8	4	0.00439	AVERAGE

Some standards and terms:

1. True positive (TP): If the outcome from a prediction is p and the actual value is also p, then it is called a true positive.
2. False positive (FP): However if the actual value is n then it is said to be a false positive.
3. Precision and recall: Precision is the fraction of retrieved instances that are relevant, while recall is the fraction of relevant instances that are retrieved. Both precision and recall are therefore based on an understanding and measure of relevance. Precision can be seen as a measure of exactness or quality, whereas recall is a measure of completeness or quantity. Recall is nothing but the true positive rate for the class [7].

In this paper, we have used weka (Waikato environment for knowledge analysis) tool for J48 algorithm calculating efficiency based on accuracy regarding correct and incorrect instances generated with confusion matrix. We have used here web-services.arff for web service classification available on web URL [12] This web-service relation consists of attributes Reliability, Throughput, availability, Response time, latency, documentation, WsRF, etc , 300 instances.

V. Experimental Work and Results

We have performed classification using algorithm J48 decision tree algorithm on web-services .arff dataset in weka tool. It is provide inbuilt algorithms for J48.

A. Results for classification using J48:

Reliability attribute has been chosen randomly for QWS data set. J48 is applied on the data set and the confusion matrix is generated for class gender having two possible values i.e. YES or NO.

Confusion Matrix:

a b ← classified as

33 72 | a = YES
25 170 | b = NO

For above confusion matrix, true positives for class a='YES' is 33 while false positives is 72 whereas, for class b='NO', true positives is 170 and false positives is 25 i.e. diagonal elements of matrix 33+170 =203 represents the correct instances classified and other elements 25+72 = 97 represents the incorrect instances.

$$\text{True positive rate} = \frac{\text{Diagonal Element}}{\text{Sum of relevant row}}$$

$$\text{False positive rate} = \frac{\text{Non-diagonal element}}{\text{Sum of relevant row}}$$

Hence,
TP rate for class a = 33/(33+72) = 0.314
FP rate for class a = 25/(25+170) = 0.128
Average TP rate = 0.677
Average FP rate = 0.491
Precision = diagonal element/sum of relevant column
Precision for class a = 33/(33+25) = 0.568
Precision for class b = 170/(170+72) = 0.702
F-measures = 2*precision*recall/(precision + recall)
F-measure for class a = 2*0.568*0.314/(0.568+0.314) = 0.404
F-measure for class b= 2*0.702*0.871/(0.702+0.871) = 0.778

VI. CONCLUSION

In selecting a web service for use, it is important to consider non-functional properties of the web service so as to satisfy the constraints or requirements of users. We present web services quality prediction model, which takes non-functional in account. From above experimental work we can conclude that correct instances generated by J48 is 203, as well as performance evolution on the basis of reliability is:

Classification Accuracy	
Reliability	J48
YES	63%
NO	37%

This proves that the, J48 is a simple classifier technique to make a decision tree. Efficient result has been taken from QWS dataset using weka tool in the experiment. The experiments results shown in the study are about accuracy. J48 gives classification accuracy for class reliability in QWS dataset having two values Yes and No. We can prove that J48 is t efficient classifier. Finally we conclude that this study can be used to classify a new web service into one of the four predetermined classes based on the models.

ACKNOWLEDGMENT

We are grateful to Dr. E. Al-Masri and Dr. Q.H. Mahmoud for providing us dataset related to the web services classification.

REFERENCES

[1] Margaret H. Danham,S. Sridhar, " Data mining, Introductory and Advanced Topics", Person education , 1st ed., 2006
[2] Ramakanta Mohanty, V.Ravi, M.R.Patra , "Web-services calssification using intelligent techiniques" , Expert systems with application s 37 (2010) 5484-5490 ElsEveir, Vol. 3, issue 7, July 2010, ISSN 0957-4174
[3] Wenke Lee, Salvatore J. Stolfo, Kui W. Mok, "A Data Mining Framework for Building Intrusion Detection Models"
[4] Jeberson R. Retna Raj, Sasipraba T., "Web Service Selection Based on QoS Constraints", IEEE, 2010.
[5] Aman Kumar Sharma, Suruchi Sahni, "A Comparative Study of Classification Algorithms for Spam Email Data Analysis", IJCSE, Vol. 3, No. 5, 2011, pp. 1890-1895
[6] Anshul Goyal, Rajni Mehta, "Performance Comparison of Naïve Bayes and J48 Classification Algorithms", IJAER, Vol. 7, No. 11, 2012, pp
[7] Xiang yang Li, Nong Ye, "A Supervised Clustering and Classification Algorithm for Mining Data With Mixed Variables", IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, Vol. 36, No. 2, 2006, pp. 396-406
[8] Tina R. Patil, Mrs. S. S. Sherekar, " Performance analysis of naïve bayes and J48 classification algorithm for data classification",International Journal Of Computer Science And Applications, Vol. 6, No.2, Apr 2013, ISSN: 0974-1011
[9] Al-Masri, E., and Mahmoud, Q. H., "QoS-based Discovery and Ranking of Web Services", IEEE 16th International Conference on Computer Communications and or QWS Dataset Version 2.0)
[10] Kalepu, S., Krishnaswamy, S., & Loke, S. W. (2004). Verity: A QoS metric for selecting web services and providers. In Proceedings of the fourth international conference on web mining systems engineering workshops (WISEW'03) (pp. 131–139).
[11] Kokash, N. (2005). Web service discovery with implicit QoS filtering. In Proceedings of the IBM PhD student symposium, in conjunction with ICSOC'05 (pp. 61–66), Netherlands. Al-Masri & Mahmoud, 2008; http://www.uogue/ph.ca/~qmahmoud/qws/index.html