Classification of Web Services Using Bayesian Network

Ramakanta Mohanty1*, V. Ravi2, M. R. Patra3

1Computer Science Department, MLR Institute of Technology, Hyderabad, India; 2Institute for Development and Research in Banking Technology, Hyderabad, India; 3Computer Science Department, Berhampur University, Berhampur, India.

Email: *ramakanta5a@gmail.com, rav_padma@yahoo.com, mrpatra12@gmail.com

Received December 17th, 2011; revised January 31st, 2012; accepted February 15th, 2012

ABSTRACT

In this paper, we employed Naïve Bayes, Markov blanket and Tabu search to rank web services. The Bayesian Network is demonstrated on a dataset taken from literature. The dataset consists of 364 web services whose quality is described by 9 attributes. Here, the attributes are treated as criteria, to classify web services. From the experiments, we conclude that Naïve based Bayesian network performs better than other two techniques comparable to the classification done in literature.

Keywords: Web Services; Quality of Services (QoS); Bayesian Network; Naïve Based Bayesian; Markov Blanket and Tabu Search

1. Introduction

Services are tendered and availed in almost all the business and industries. The growth and proliferation of IT across industries and business appear to have fuelled the requirement as well as the delivery of services profoundly. Delivering services has been an attractive business proposition for many industries lately. The latest development in the systems is a new paradigm, called web services [1]. Web Services heralded another significant mile stone in the history of IT. Earlier, Internet catered mostly to the business to Customer (B2C) category of the users on the web. As against this, Web Services enable B2B interaction as well Web. They are independent of platform and natural languages, which is suitable for accessing 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 by employing pervasive standards based web technologies. Web services are based on XML and three other core technologies: WSDL, SOAP, and UDDI. WSDL is a document which describes the services’ 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. With the help of 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 [2].

In this study, we address this problem of efficiently identifying a set quality attributes by employing Bayesian Networks vz. Naïve Bayes, Markov blanket and Tabu search. Naïve Bayes is special form of Bayesian networks that is widely used for classification [3] and clustering [4], but its potential for general probabilistic modeling (i.e., to answer joint, conditional and marginal queries over arbitrary distributions) remains largely unexploited. Naïve Bayes represents a distribution as a mixture of components, where within each component all variables are assumed independent of each other. The Markov blanket of a variable Y, (MB(Y)), by definition, is the set of variables such that Y is conditionally independent of all the other variables given MB(Y). A Markov Blanket Directed Acyclic Graph (MB DAG) is a Directed Acyclic Graph over that subset of variables. When the parameters of the MB DAG are estimated, the result is a Bayesian Network, defined in the next section. Recent research by the machine learning community [5-7] has sought to identify the Markov blanket of a target variable by filtering variables using statistical decisions for conditional independence and using the MB predictors as the input features of a classifier. However, learning MB DAG classifiers from data is an open problem [8]. There are several challenges: the problem of learning the graphical structure with the highest score (for a variety of scores) is NP hard [8] for methods that use conditional independencies to guide graph search, identifying conditional independencies in the presence of limited data is quite unreliable and the presence of multiple local

*Corresponding author.
Classification of Web Services Using Bayesian Network

2. Quality Issues in Web Services

QoS plays an important role in finding out the performance of web services. Earlier, QoS has been used in networking and multimedia applications. Recently, there is a trend in adopting this concept to web services [9]. The basic aim is to identify the QoS attributes [10-12] for improving the quality of web services through replication services [10], load distribution [13], and service redirection [14]. To measure the QoS of a web service, attributes like Response Time, Throughput, Availability, Reliability, Cost, and Response Time are considered.

2.1. QoS Attributes

According to Kalepu et al. [15], quality of service (QoS) is a combination of several qualities or properties of a service, such as: 1) Availability; 2) Reliability; 3) Price; 4) Throughput; 5) Response Time; 6) Latency; 7) Performance; 8) Security; 9) Regulatory; 10) Accessibility; 11) Robustness/Flexibility; 12) Accuracy; 13) Serviability; 14) Integrity and 15) Reputation. QoS parameters determine the performances of the web services and find out which web services are best and meet user’s requirements.

Users of web services are not human beings but programs that send requests for services to web service providers. QoS issues in web services have to be evaluated from the perspective of the providers of web services (such as the airline-booking web service) and from the perspective of the users of these services (in this case, the travel agent site) [16]. There are other models available related to the quality of web services issues. A QoS model [16] represented in Table 1 shows that the main classification of QoS attributes is based on internal attributes, which are independent of the service environment, and external attributes that are dependent on the service environment. The attributes of the model in Table 1 are almost similar to the attributes of QWS Dataset used in this paper.

2.2. Description of QWS Dataset

QWS dataset [17] consists of different web service implementations and their attributes as presented in Table 2. The classification is measured based on the overall quality rating provided by all the attributes. The functionality of the web services can be helpful to differentiate between various services. The attributes G1 to G10 are used as explanatory variables and the attribute G11 is used as the target variable. However, attributes G12 and G13 are ignored as they do not contribute to the analysis.

The web services [1,2] in the QWS dataset are classified into four categories, such as: 1) Platinum (high quality); 2) gold; 3) silver and 4) bronze (low quality). The classification is measured based on the overall quality rating provided by WSRF. It is grouped into a particular web service based on classification. The functionality of the web services can be helpful to differentiate between various services [14].

3. Overview of Bayesian Networks

A Bayesian network is a directed acyclic graph model that represents conditional independencies between a set of variables [18,19]. It has two constituents: One is a network graphical structure which is a directed acyclic graph with the nodes of variables and arcs of relations. The other is the conditional probability table associated with each node in the model graph. Machine learning techniques are able to estimate the structure and the conditional probability table from the training data. Based on the Bayesian probability inference, the conditional probability can be estimated from the statistical data and propagated along the links of the network structure to the target label. By setting a threshold of confidence, the final probability value can be used as the indication for the classification decision. The Bayesian formula can be mathematically expressed as below:

\[
P(H_j | E) = \frac{P(E | H_j) \times P(H_j)}{\sum_{j=1}^{n} P(E | H_j) \times P(H_j)}
\]

(1)

According to the basic statistical theory, e.g., the Chain Rule and independency relation derived from the network structure, the joint probability of \( E \) can be
Table 1. QoS model of web services [4].

<table>
<thead>
<tr>
<th>QoS Factor</th>
<th>Internal attributes (Metrics)</th>
<th>External Attributes (Metrics)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reliability</td>
<td>Correctness (accuracy</td>
<td>Availability and consistency</td>
</tr>
<tr>
<td></td>
<td>and precision)</td>
<td></td>
</tr>
<tr>
<td>Performance</td>
<td>Efficiency (Time and Space</td>
<td>Load management</td>
</tr>
<tr>
<td></td>
<td>Complexity)</td>
<td>(Throughput, waiting and</td>
</tr>
<tr>
<td></td>
<td></td>
<td>response time security)</td>
</tr>
<tr>
<td>Integrity</td>
<td></td>
<td>Security</td>
</tr>
<tr>
<td>Usability</td>
<td>Input and output</td>
<td></td>
</tr>
</tbody>
</table>

Table 2. Attributes of QWS dataset.

<table>
<thead>
<tr>
<th>ID</th>
<th>Attribute Name</th>
<th>Description</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>G1</td>
<td>Response Time</td>
<td>Time taken to send a request and receive a response</td>
<td>ms</td>
</tr>
<tr>
<td>G2</td>
<td>Availability</td>
<td>Number of successful invocations/total invocations</td>
<td>%</td>
</tr>
<tr>
<td>G3</td>
<td>Throughput</td>
<td>Total number of invocations for a given period of time</td>
<td>Invokes/second</td>
</tr>
<tr>
<td>G4</td>
<td>Successability</td>
<td>Response/Number of request messages</td>
<td>%</td>
</tr>
<tr>
<td>G5</td>
<td>Reliability</td>
<td>Ratio of the number of error messages to total messages</td>
<td>%</td>
</tr>
<tr>
<td>G6</td>
<td>Compliance</td>
<td>The extent to which a WSDL document follows WSDL Documentation</td>
<td>%</td>
</tr>
<tr>
<td>G7</td>
<td>Best Practices</td>
<td>The extent to which a web service follows</td>
<td>%</td>
</tr>
<tr>
<td>G8</td>
<td>Latency</td>
<td>Time taken for the server to process a given request</td>
<td>ms</td>
</tr>
<tr>
<td>G9</td>
<td>Documentation</td>
<td>Measure a documentation (i.e., description tags) in WSDL</td>
<td>%</td>
</tr>
<tr>
<td>G10</td>
<td>WSRF</td>
<td>Web service relevance function: a rank for web service Quality</td>
<td>%</td>
</tr>
</tbody>
</table>

calculated by the production of local distributions with its parent nodes, i.e.

\[
P(E) = \prod_{i=1}^{n} P(E_i | \text{Parent of } (E_i))
\]  

(2)

In the above formula, \( E \) denotes a set of variable values, i.e. \( E = \{E_1, E_2, \cdots, E_n\} \). H is termed as hypothesis. H is called the prior probability and \( P(H|E) \) is called posteriori probability of \( H \) given \( E \). If \( E_i \) has no parent nodes, Parent Of \( (E_i) \) is equal to \( P(E_i) \).

3.1. Naïve Bayes

Naïve Bayes models are so named for their “naïve” assumption that all variables \( X_i \) are mutually independent given a “special” variable \( C \). The joint distribution is then given compactly by \( P(C, X_1, \cdots, X_n) = P(C) \prod_{i=1}^{n} P(X_i|C) \). The univariate conditional distributions \( P(X_i|C) \) can take any form (e.g., multinomial for discrete variables, Gaussian for continuous ones). When the variable \( C \) is observed in the training data, naïve Bayes can be used for classification, by assigning test example \( (X_1, \cdots, X_n) \) to the class \( C \) with highest \( P(C|X_1, \cdots, X_n) \) [2]. When \( C \) is unobserved, data points \( (X_1, \cdots, X_n) \) can be clustered by applying the EM algorithm with \( C \) as the missing information, each value of \( C \) corresponds to a different cluster, and \( P(C|X_1, \cdots, X_n) \) is the point’s probability of membership in cluster \( C \) [14]. Naïve Bayes models can be viewed as Bayesian networks in which each \( X_i \) has \( C \) as the sole parent and \( C \) has no parents. A naïve Bayes model with Gaussian \( P(X_i|C) \) is equivalent to a mixture of Gaussians with diagonal covariance matrices. While mixtures of Gaussians are widely used for density estimation in continuous domains, naïve Bayes models have seen very little similar use in discrete and mixed domains. However, they have some notable advantages for this purpose. In particular, they allow for very efficient inference of marginal and conditional distributions. To see this, let \( X \) be the set of query variables, \( Z \) be the remaining variables, and \( k \) be the number of mixture components (i.e., the number of values of \( C \)). We can compute the marginal distribution of \( X \) by summing out \( C \) and \( Z \):

\[
P (X=x) = \sum_{c}^{k} \sum_{z}^{c} P(C=c, X=x, Z=z)
\]

\[
= \sum_{c}^{k} \sum_{z}^{c} P(c) \prod_{j=1}^{k} P(x_j|c) \prod_{j=1}^{k} P(z_j|c)
\]

\[
= \sum_{c}^{k} P(c) \prod_{j=1}^{k} P(x_j|c) \sum_{z_j} P(z_j|c)
\]

where the last equality holds because, for all \( j \),

\[
\sum_{c} P(z_j|c) = 1
\]

Thus the non-query variables \( Z \) can simply be ignored when computing \( P(X=x) \), and the time required to compute \( P(X=x) \) is \( O(|X|k) \), independent of \( |Z| \). This contrasts with Bayesian network inference, which is worst-case exponential in |Z|. Similar considerations apply to conditional probabilities, which can be computed efficiently as ratios of marginal probabilities: \( P(X=x|Y=y) = P(X=x, Y=y)/(Y=y) \). A slightly richer model than naïve Bayes which still allows for efficient inference is the mixture of trees, where, in each cluster, each variable can have one other parent in addition to \( C \). The basic graph of Bayesian network is presented in Figure 1 and the graph of QWS dataset for naïve Bayes are depicted in Figure 3.
3.2. Markov Blanket

The Markov condition implies that the joint distribution $P$ can be factorized as a product of conditional probabilities, by specifying the distribution of each node conditional on its parents [20]. In particular, for a given DAG $S$, the joint probability distribution for $X$ can be written as

$$P(X) = \prod_{i=1}^{n} P(X_i | Pa_i)$$

(3)

where $Pa_i$ denotes the set of parents of $X_i$ in $S$; this is called a Markov factorization of $P$ according to $S$. The set of distributions represented by $S$ is the set of distributions that satisfy the Markov condition for $S$. If $P$ is faithful to the graph $S$, then given a Bayesian Network $(S, P)$, there is a unique Markov blanket for $Y$ consisting of $PaY$, the set of parents of $Y$, $chY$, the set of children of $Y$, and $Pa chY$, the set of parents of children of $Y$.

For example, consider the two DAGs in Figures 1 and 2, above. The factorization of $P$ entailed by the Bayesian Network $(S, P)$ is

$$P(Y, X_1, \cdots, X_6) = P(Y | X_1) \cdot P(X_4 | X_2, Y) \cdot P(X_1 | X_4) \cdot P(X_3 | X_4) \cdot P(X)$$

(4)

The factorization of the conditional probability $p(Y | X_1, \cdots, X_6)$ entailed by the Markov blanket for $Y$ corresponds to the product of those (local) factors in equation (2) that contain the term $Y$.

$$P(Y | X_1, \cdots, X_6) = C_0 \cdot P(Y | X_1) \cdot P(X_1 | X_4) \cdot P(X_3 | X_4) \cdot P(X)$$

(5)

The graph of QWS dataset is depicted in Figure 5 for Markov blanket of Bayesian networks.

3.3. Tabu Search

A Heuristic is an algorithm or procedure that provides a shortcut to solve complex decision problems. Heuristics are used when you have limited time and/or information to make a decision. For example, some optimization problems, such as the travelling salesman problem, may take far too long to compute an optimal solution. A good heuristic is fast and able find a solution that is no more than a few percentage points worse than the optimal solution. Heuristics lead to good decisions most of the time, but not always. There have been several Meta-heuristic applications recently in Machine Learning, Evolutionary Algorithms, and Fuzzy Logic problems. Tabu Search is a meta-heuristic strategy that is able to guide traditional local search methods to escape the trap of local optimality with the assistance of adaptive memory [21].
on selected concepts that cut across the fields of artificial intelligence and operations research.

Tabu list ($TL$) is given by

$$TL = \{s^{-1} : s = s_i, i > k - t_i \}$$

where $k$ is the iteration index and $s^{-1}$ is the inverse of the move $s$; i.e., $s^{-1}(s(x)) = x$. In words, $TL$ is the set of those moves that would undo one of those moves in the $t$ most recent iterations. $t$ is called the Tabu tenure.

The use of $TL$ provides the “constrained search” element of the approach, and hence the solution generated depends critically on the composition of $TL$ and the way it is updated. Tabu search makes no reference to the condition of local optimality, except implicitly where a local optimum improves on the best solution. A best move, rather than an improving move is chosen at each step. This strategy is embedded in the OPTIMUM function. Tabu search analysis of QWS attribute is depicted in Figure 5.

4. Results and Discussions

We employed Naïve Bayes, Markov blanket and Tabu search techniques to classify web services. We note that the average accuracy of Naïve Bayes classifier is 85.62%, followed by Tabu search of 82.45% and Markov blanket of 81.36% presented in Table 3. In this context, we employed Back propagation trained neural network to find the importance of different attributes in web services. We found that we found that WRRF plays a vital role for classifying the web services. We excluded the WSRF from dataset and experimented. We observe that the average accuracy of Naïve Bayes is 75.01% and Marocv Blanket is 65.48% and Tabu search is 71.38% presented in Table 4. As Bayesian network is a very good classifier to classify classification type of problems. In this context, the result obtained by Bayesian classifier is not superior to the results obtained by [22] to classify the accuracy of web services. As Bayesian network has not been applied to classify web services, we employed this approach in the present study.

5. Conclusion

We presented Naïve Bayes, Markov blanket and Tabu search to classify web services. We observed that Naïve Bays approach predicted better accuracy than Markov blanket and Tabu search. Secondly, Bayesian belief network is employed first time to classify web services in the present study. Future directions include more exploration of Markov blanket approach for rule generation and quality of attributes to decide the classification of web services.

6. Acknowledgements

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

REFERENCES


