Semi-automatic Algorithm Based on Web Service Classification

Jie Yang, Xianzhong Zhou
School of Management and Engineering, Nanjing University,
Nanjing 210093, china
Dg1115014@gmail.com

Abstract. The number of web services on the Internet is increasing continuously. Determining a web service’s domain is very helpful for many issues like web service discovery, composition and semantic annotation. Based on predefined domains, we implemented semi-automatic web service classification with supervised machine learning technique in the paper. First we exploited the process of web service classification with choosing element from web service description document, preprocessing, mapping to a digital vector and applying classifier. Then service classification accuracy of single element and multiple elements based on four machine learning algorithms were tested. At last, experimental results showed that C4.5 classifier had a higher accuracy and efficiency in web service classification. In addition, multi-element classification did not perform better than single element.

Keywords: Web Services; Classification; Machine Learning

1 Introduction

With the drastic growth of web services on the Internet, web service discovery is a hot topic now. It is time consuming to traverse a whole repository to find a matched service. To improve efficiency of web service discovery, we adopt machine learning technique to classify web services. UDDI is a most widely shared registry, and providers publish web services into it. UDDI provides classification category based on taxonomies. Providers define category for a web service manually. And in some websites, web services are listed by submission time or providers. These are not very helpful for web service discovery and semantic annotation. In this paper, our goal is to classify web services with the identical or similar functionalities into one category semi-automatically and annotate web services semantically.
2 Web Service Classification

As the statement in Introduction, web services with the identical or similar functionalities are classified into one category. This section will explain the process of web service classification, which can be seen in Figure 1. Web service classification is based on wsdl document. As WSDL is the most widely accepted and used web service description language.

![Fig. 1. Steps of Web Service Classification](image)

### 2.1 Choosing element

Choosing element is the first step of web service classification. Elements include static and dynamic. For example, QoS (Quality of Service) is dynamic element. QoS refers to the quality of a web service, and may include availability, reliability, response time, etc. These measures change with web service’s invoked times frequently. We choose to utilize static elements rather than dynamic, because static elements represent web service’s functionality better than dynamic. Words are extracted from contents of chosen elements in description document.

### 2.2 Preprocessing.

This step is to preprocess extracted words. There are four aspects: splitting, eliminating stop words, stemming and removing specific tags. Splitting: combined words are split into verbs and nouns. Because of naming convention, verb and noun are combined in operation names, service names and messages of web services. Eliminating stop words: words are filtered by stop list. The stop list contains a few common words (including preposition, article, etc.), which are recognized that they are not useful in information retrieval. And in web service classification, stop words could not represent any category either. Stemming, verbs are taken back to infinitive; nouns are taken from plurals to singles; English words are taken to American words. Removing tags and specific words: they are not helpful for classification. For example, “web”, “service” and service provider’s name frequently occur in wsdl files, but they
cannot discriminate web services. Tags like `<input>`, `<output>`, exist in almost every WSDL file.

3 Experimental Analysis and Results

We will give the measurement of web service classification in this section.

The accuracy of classification is defined in formula (1) and formula (2). Classification accuracy of collection is the average value of all categories’ accuracy.

\[
\text{accuracy}(T) = \frac{a(C_1) + a(C_2) + \cdots + a(C_n)}{n}
\]  

(1)

\[
a(C_i) = \frac{N(x|m)}{m}
\]

(2)

\(a(C_i)\) is the accuracy of \(C_i\). \(m\) is the number of unclassified web services which belong to \(C_i\). \(N(x|m)\) is the number of web services which are classified into \(C_i\) in \(m\).

Our experimental dataset is based on OWLS-TC4, which provides many web service documents. The contents of web service’s elements (service name, operation, input and output), are extracted. After preprocessed, four bags of words are formed. And we will obtain each word’s tf-idf value in each category. Based on the gained data, machine learning algorithms are employed to implement classification.

![Graph](a.png)
4 Conclusion

The paper implemented web service classification with machine learning algorithms. The process includes choosing classification element, preprocessing, mapping to a vector and adopting classifier. According to the experiments, C4.5 classifier performs better than Naïve Bayes, SVM and BPNN. Multi-element classification doesn’t act better than single element. And when dataset is small, it is a good way to choose a web service’s category by user. In future work, we will introduce classification into web service discovery to satisfy user’s functional requirements.

References