# Significant analysis of terms in bootstrapping ontology for web services

Swathi, Madhumitha, Venkatachalam

**Abstract**— Ontology construction is important for semantic based web services.Identifying concepts and relationships for a specific given domain is one of the promising technique in ontology construction. Bootstrapping means, dynamically creating ontology for specific domain or web services. Bootstrapping ontology based on set of predefined textual sources such as web services, must address the problem of multiple, largely unrelated concepts. In this paper ontology bootstrapping process for web services is done based on identifying the related concepts. WSDL document contains large set of tokens. The tokens are may be closely related to respective WSDL file or sometimes tokens related with the less importance to the WSDL file. The proposed approach finds the related concept that have more significant for the domain.it uses result of two methods to find the related concept: significant analysis of terms using relevance abstraction identification method, and web context extraction using WordNet for the ontology evolution.

**Index Terms**— Bootstrapping process, Ontology, service oriented modeling, significant analysis, Service Discovery, WordNet, Webservices.

## 1 INTRODUCTION

Ontologies serve as the heart of the semantic web. Creating and maintining ontology is very difficult task. Ontology bootstrapping involves automatic identification of concepts relevant to tha domain and relationship between the concepts.previous works on ontology creation focused on TF/IDF calculation. The TF/IDF does not provide significant of terms for the relevant domain. Instead it considers every term as important for domain and calculates TF/IDF for ontology construction. In this previous work, ontologies are constructed for every concept which is also irrelevant for domain. Universal Description, Discovery and Integration (UDDI) are a directory service where businesses can register and search for Web services. UDDI was originally proposed as a core Web service standard and created to encourage interoperability and adoption of web services. The increasing number of available web services makes it difficult to classify web services using single domain ontology or a set of existing ontologies created for other purposes [1].

The proposed work in this paper constructs ontology for web services based on significance analysis. Concept evocation and ontology evolution is done based on significant score. The significant score identifies the more relevant concepts and relationships for ontology construction. The significant score is calculated using two results:

1) Relevance based abstraction method.
2) Web context extraction using WordNet.

In our implementation first we are finding the related keywords using web context extraction and significant score method. We are calculating similarity values between the words. From those words we can identify related words, these words are used when finding document in web services. The following scenarios in our approach the weight values of term is calculated with relevance with other terms. This relevance based score calculation is effectively improving our TF/IDF based calculations.

## 2 EXSISTING SYSTEM

The existing bootstrapping approach enables the automatic construction of an ontology that can assist, classify, and retrieve relevant services, without the prior training required by previously developed methods. As a result, ontology construction and maintenance effort can be substantially reduced. This bootstrapping process is based on analyzing a web service using three different methods, where each method represents a different perspective of viewing the web service. As a result, the process provides a more accurate definition of the ontology and yields better results. In particular, the Term Frequency/Inverse Document Frequency (TF/IDF) method analyzes the web service from an internal point of view, i.e., what concept in the text best describes the WSDL document content. The Web Context Extraction method describes the WSDL document from an external point of view. i.e., what most common concept represents the answers to the web search queries based on the WSDL content. Finally, the Free Text Description Verification method is used to resolve inconsistencies with the current ontology.

- *Swathi, PG student in sri Krishna college of engineering and technology,India, PH -9095404285. E- mail: swathiangamuthu@gmail.com*
- *Madhumitha , Assistant professor in sri krishana college of engineering and technology, India, PH-9952286944. E-mail : madhuperu@gmail.com.*
- *Venkatachalam, Assistant Professor in sri krishana college of engineering and technology, India, PH-9976468985. E-mail: venkatme83@gmail.com.*

An ontology evolution is performed when all three analysis methods agree on the identification of a new concept or a relation change between the ontology concepts. The relation between two concepts is defined using the descriptors related to both concepts. This approach can assist in ontology construction and reduce the maintenance effort substantially. The approach facilitates automatic building of an ontology that can assist in expanding, classifying and retrieving relevant services, without the prior training required by previously developed approaches.

## 3 MOTIVATION

Dynamic creation of ontology is very difficult task. The main problem in web service is largely unrelated concepts. Literature works on ontology bootstrapping done with only limited domains. Since UDDI registries are not based on limited domain, it has dynamic registration of web services by various business concerns in world. The second problem is that ontologies are created by expanding the existing ontology. Due to this, new concepts cannot be identified and also memory is wasted. The already created concepts may be out dated but still it is not deleted. Some concepts may need only little update based on advanced concepts. But the previous work will create new ontology itself instead of little update in existing ontology.

## 4 RELATED WORKS

### 4.1 Bootstrapping in semantic web

The process of creating semiautomatic alignment methods is called as "Parameterizable Alignment Methods" (PAM). The bootstrapping approach is performed for acquiring the parameters that drive such a PAM. This approach called as APFEL for "Alignment Process Feature Estimation and Learning". The learnt PAM may be applied to ontologies of specific domains. From the learned classifiers they derive whether concepts in two schemas correspond to each other[ 2].The bootstrapping ontology for informational retrieval uses the ranked objects in attribute concepts formulates (keyword by keyword) context for concept, bootstraps the learning of domain-specific concept hierarchies using FCA, and incorporates the learnt concept hierarchies and WordNet for content-based document classification[3].The ontology evolution approach proposed by BOEMIE puts significant effort in maintaining the consistency of the ontology while trying on the same time to identify and eliminate redundant information[4]. Bootstrapping and populating specialized domain ontologies uses tree-mining algorithms that identify key domain concepts and their taxonomical relationships. Experimental evaluation for the News and Hotels domain indicates that our algorithms can bootstrap and populate domain specific ontologies with high precision and recall[5 ].

### 4.2 Ontology creation and evolution

Recent work has focused on ontology creation and evolution and in particular on schema matching. Many heuristics were proposed for the automatic matching of schemata (e.g., Cupid [9], GLUE [10], and OntoBuilder [11]), and several theoretical models were proposed to represent various aspects of the matching process such as representation of mappings between ontologies [12], ontology matching using upper ontologies [13], and modeling and evaluating automatic semantic reconciliation [14]. However, all the methodologies described require comparison between existing ontologies. The realm of information science has produced an extensive body of literature and practice in ontology construction, e.g., [15]. Other undertakings, such as the DOGMA project [16], provide an engineering approach to ontology management. The existing bootstrapping work automatically evolves an ontology for web services from the beginning by considering tf/idf ranking. Were as our bootstrapping work based on significant score. In addition, a survey on the state-of the art web service repositories [17] suggests that analyzing the web service textual description in addition to the WSDL description can be more useful than analyzing each descriptor separately. TF/IDF and web content extraction methods overcomes the NLP disadvantages by using web context recognition. The survey mentions the limitation of existing ontology evolution techniques that yield low recall. Our solution overcomes the low recall by using relevance based identification methods using significant score.

## 5 PROPOSED BOOTSTRAPPING ONTOLOGY MODEL

The bootstrapping ontology model proposed in this paper is based on the continuous analysis of WSDL documents and employs an ontology model based on concepts and relationships [1]. The innovation of the proposed bootstrapping model centers on 1) the combination of the use of two different extraction methods, significant score analysis and web based concept generation, and 2) the verification of the results using a Free Text Description Verification method by analyzing the external service descriptor. We utilize these three methods to demonstrate the feasibility of our model. It should be noted that other more complex methods, from the field of Machine Learning (ML) and Information Retrieval (IR), can also be used to implement the model. However, the use of the methods in a straightforward manner emphasizes that many methods can be "plugged in" and that the results are attributed to the model's process of combination and verification. Our model integrates these three specific methods since each method presents a unique advantage— internal perspective of the web service by the significant score, external perspective of the web service by the Web Context Extraction, and a comparison to a free text description, a manual evaluation of the results, for verification purposes.

IJSER

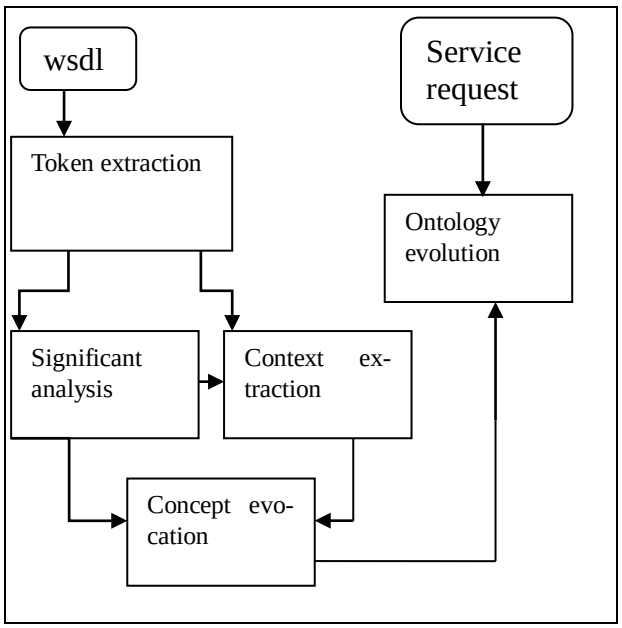


Figure 1 Bootstrapping process and ontology construction

### 5.1 Token extraction

The token extraction is first step in bootstrapping process. The WSDL descriptors are called as set of tokens. The tokens are extracted separately from WSDL document.this list serves as a baseline for ontology creation All elements classified as name are extracted, including tokens that might be less relevant. The first word being capitalized is extracted as single token. The sequence of words is expanded using capital letters. The tokens are filtered using a list of stopwords, removing words with no substantive semantics.

### 5.2 Significant analysis

The main purpose of applying statistical methods for abstraction identification is to rank candidate abstractions based on a particular criterion that gives higher scores to likely abstraction candidates. The most common statistical technique is to infer the significance of a candidate term (and thus its likelihood of signifying an underlying abstraction) from the number of times it occurs in the document.

In the previous paper work, simple frequency profiling is hard to beat; however, one way to improve upon it is to apply additional knowledge such as the standard distributional properties of candidate terms by performing corpus-based frequency profiling. These properties can be determined if there exists a large enough normative corpus within which the term occurs a representative number of times. The rate of occurrence thus predicted by the normative corpus can be compared with the actual rate of occurrence in the analyzed document, and the difference used to infer the strength of the term's relevance to the domain. The more overrepresented a term is within a domain document (as compared to the representation in the corpus), the more likely it is to signify an important domain abstraction.

Corpus-based frequency profiling works as follows. Assume we are interested in the significance of word w in the domain document. The domain document contains a total of nd words, and the normative corpus contains nc words. w occurs wd times in the domain document and wc times in the normative corpus. wd and wc are called the observed values of w. Based on the occurrences of w in the domain document and the normative corpus, we can define two expected values for w:

Ed = nd(wd + wc)/ (nd + nc)
Ec = nc(wd + wc)/ (nd + nc)

The log-likelihood value for w is then

$$LLw = 2( W_d \ .ln\ W_d /E_d + W_c \ .ln\ W_c /E_c ) \quad (1)$$

Given a log-likelihood value for each term in the domain document, the terms can be ranked, placing the term with the highest LL value, and thus most likely to represent an underlying abstraction, at the top. This corpus based frequency profiling is the primary technique used successfully by W Matrix . It is also used by RAI, but the results of RAI are modified by the technique described below in order to cope with multiword terms.

There is a particular challenge associated with multiword terms since most techniques, including corpus-based frequency profiling, rely on identifying individual words, and count these individually. There are collocation analysis techniques that can infer lexical affinities; however, since most association measures are defined to measure the pair-wise adhesion of words (wi,wj) only, they cannot be used for measuring the association between more than two words. In requirements engineering, it is fairly common to encounter domain terms, such as software requirements specification, that comprise more than two words. Correctly handling such sequences is therefore an important challenge, since several researchers claim that in specialized domains over 85% domain-specific terms are multiword units. In RAI, we apply simple syntactic patterns that posit multiword terms as common combinations of adjectives and nouns, adverbs and verbs, and prepositions. Key problem is that although multiword terms can be identified, in abstraction identification we want to rank terms in order of the relevance of their signified abstractions. In terms of pure frequency, it is common for important multiword terms to occur relatively infrequently in a document. Worse, no normative corpus of which we are aware contains large numbers of multiword terms. This is because most such terms are specific to particular domains and hence are unlikely to find their way into a corpus whose role is to serve as a guide to general usage of a language (e.g., English). Hence, while the corpus-based frequency profiling technique described above works well for terms that are single words, in practice it doesn't help with multiword terms.

To solve this problem, we synthesize a significance value for all terms using a heuristic based on the number of words of which the term is composed, and the LL value for each word. In its simplest form, the significance value for a term t ={w1, w2; . . .; wi} is given by the formula:

$$S_t = \sum_i LL_{w_i}/l \quad (2)$$

It calculates the mean of the LL values for all the component words comprising a multiword term. However, we hypothesize that not all the words contribute equally to the significance value of the multiword term of which they are a component. Our hypothesis is based on an assumption that such a term is typically composed of a headword and one or more modifiers. Thus, in the term sailing ship, the headword is the noun ship and the adjective sailing is a modifier that denotes sailing ship as a type or class of ship. We assume that the headword is the most significant component of the term; thus the term ship is more significant than sailing, and the LL value of ship should carry more weight than the LL of sailing.To accommodate our hypothesis, the significance equation is modified to incorporate a weight, ki, that assigns a weight to each word that is a component of the term (based on its position)

$$S_t = \sum_i K_i LL_{w_i}/\ l \quad (3)$$

Relevance-driven Abstraction Identification (RAI), has been designed to support abstraction identification in RE. It combines a number of existing natural language processing (NLP) techniques in a novel way to enable it to handle both single and multiword terms, ranked in order of confidence. One of the main contributions of our work is the evaluation method that we use for RAI, which avoids the problems associated with employing expert human judgment for evaluating how well map onto the problem domain's underlying abstractions.

### 5.3 Context Extraction

The tokens are passed to the WordNet and the similar key terms are extracted. We define a context descriptor ci from the web services. Each descriptor can define a different point of view of the concept. The semantic similar terms are identified and these terms form base for ontology construction. The term price may be similar to money, cost Etc. The service may not know similar meanings while the user search for information. So we extract similar terms from WordNet for improving semantics.

### 5.4 Concept Evocation

Concept evocation identifies a possible concept based on context intersection. An ontology concept is defined by the descriptors that appear in the intersection of both the web context results and the RAI results. The context, C, is initially defined as a descriptor set extracted from the web and representing the same document. As a result, the ontology concept is represented by a set of descriptors, ci, which belong to both sets.

### 5.5 Ontology evolution

The concepts extracted used for ontology evolution. The class is identified as class or subclass , then the relationship between the classes are identified.The ontology evolution consists of four steps including:
1. building new concepts,
2. determining the concept relations,
3. identifying relations types, and
4. resetting the process for the next WSDL document.

Building a new concept is based on refining the possible identified concepts. The evocation of a concept in the previous step does not guarantee that it should be integrated with the current ontology. Instead, the new possible concept should be analyzed in relation to the current ontolgy.the algorithmis explained below.

**Step 1:** For Each Web Services

**Step 2:** Extract Tokens from WSDL

**Step 3:** RAI $_{result =}$ apply RAI algorithm to $D_{wsdl}$

**Step 4:** WebContext $_{result}$ =apply Web Context algorithm to $D_{ws}$

**Step 5:** PossibleCon $_i$ = RAI $_{result}$ $\cap$ WordNet $_{result}$

**Step 6:** if (PossibleCon $_i$ $\subseteq$ $D_{desc}$)

**Step 7:** Con $_i$ = WordNet $_{result}$ $\cap$ RAI $_{result}$

**Step 8:** PossibleRel $_i$= WordNet $_{result}$ $\cup$ RAI $_{result}$

**Step 9:** For each concept pair con $_i$ ,con $_j$

**Step 10:** If (con $_i$ $\subseteq$ con $_j$)

**Step 11:** con $_i$ subclass con $_j$

else

Re( con $_i$ ,con $_{j)}$ = PossibleRel $_i$ $\cap$ PossibleRel $_{j.}$

## 6 IMPLEMENTATION RESULT

The precision is calculated for the concepts generated by the different methods. Every method listed the concepts that were analyzed to evaluate how many of them are meaningful and could be related to at least one of the services. The precision is defined as the number of relevant (or useful) concepts divided by the total number of concepts generated by the method. The precision is analysed for increasing number of web servicesAs a result we can find the significant analysis gives the highest precision compare to the other methods.

Next, recall for the concepts generated by the methods is analyzed. Recall is defined as the number of classified web services according to the list of concepts divided by the number of services. In the recall a set of an increasing number of web services was analyzed as like previous precision result. The last concept generation experiment compared the recall and the precision for each method. Fig. 11 depicts the recall versus precision results. The previous methods recall is not

perfect as our proposed approach recall.

## 7 CONCLUSION

The bootstrapping process in the previous work is based on TF/IDF count does not consider the significance of the terms in the WSDL document. Our proposed work consider the significance of terms by using corpus. The value of the concept relations is obtained by analysis of the union and intersection of the concept results. The approach enables the automatic construction of an ontology that can assist, classify, and retrieve relevant services, without the prior training required by previously developed methods. As a result, ontology construction and maintenance effort can be substantially reduced. Since the task of designing and maintaining ontologies remains difficult, our approach in this paper improves efficiency in ontology construction.

## REFERENCES

[1] Aviv Segev, and Quan Z. Sheng " Bootstrapping Ontologies For Web Services" IEEE Transactions On Services Computing, 2012

[2] M. Ehrig, S. Staab, and Y. Sure, "Bootstrapping Ontology Alignment Methods with APFEL," Proc. Fourth Int'l Semantic Web Conf. 2005.

[3] G. Zhang, A. Troy, and K. Bourgoin, "Bootstrapping Ontology Learning for Information Retrieval Using Formal Concept Analysis and Information Anchors," 2006.

[4] S. Castano, S. Espinosa, A. Ferrara, V. Karkaletsis, A. Kaya, S.Melzer,R. Moller, S. Montanelli, and G. Petasis, "Ontology Dynamic with Multimedia Information: The BOEMIE Evolution Methodology 2007.

[5] H. Davulcu, S. Vadrevu, S. Nagarajan, and I. Ramakrishnan, "OntoMiner: Bootstrapping and Populating Ontologies from Domain Specific Web Sites," Sept./Oct. 2003

[6] D. Kim, S. Lee, J. Shim, J. Chun, Z. Lee, and H. Park, "Practical Ontolgy Systems for Enterprise Application," Proc. 10th Asian Computing Science Conf. (ASIAN '05), 2005.

[7] N. Oldham, C. Thomas, A.P. Sheth, and K. Verma, "METEOR-S Web-Service Annotation Framework with Machine Learning Classification," 2004.

[8] A. Heß, E. Johnston, and N. Kushmerick, "ASSAM: A Tool for Semi-Automatically Annotating Semantic Web Services," 2004.

[9] J. Madhavan, P. Bernstein, and E. Rahm, "Generic Schema Matching with Cupid,"2001. [10] A. Doan, J. Madhavan, P. Domingos, andA.Halevy, "Learning to Map between Ontologies on the Semantic Web," (WWW '02),

[11] A. Gal, G. Modica, H. Jamil, and A. Eyal, "Automatic OntologyMatching Using Application Semantics," AI Magazine, 2005.

[12] J. Madhavan, P. Bernstein, P. Domingos, and A. Halevy, "Representing and Reasoning about Mappings between Domain Models," 2002.

[13] V. Mascardi, A. Locoro, and P. Rosso, "Automatic Ontology Matching via Upper Ontologies: A Systematic Evaluation," IEEE Trans. KDE 2009.

[14] A. Gal, A. Anaby-Tavor, A. Trombetta, and D. Montesi, "A Framwork for Modeling and EvaluatingAutomatic Semantic Reconciliation," 2005.

[15] B. Vickery, Faceted Classification Schemes. Graduate School of Lbrary Service, Rutgers, The State Univ., 1966.

[16] P. Spyns, R. Meersman, and M. Jarrar, "Data Modelling versus Ontology Engineering," ACM SIGMOD Record, 2002.

[17] M. Sabou and J. Pan, "Towards Semantically Enhanced Web Service Repositories," Web Semantics, 2007.

IJSER