Abstract. Efficiently finding Web Services on the Web is a challenge research question in Service Oriented Architecture-SOA. Web services are becoming necessary part of the Semantic Web, and research on Semantic Web Services aims to automate their use. However, finding the adequate service for each individual user becomes a more and more demanding problem.

In this paper we suggest a new approach for Web Service Discovery which we believe that is the first attempts that combines clustering techniques with Latent Semantic Indexing. Empirically the proposed method will improve the quality of automated service discovery.

Keywords. SOA, Web Services, Semantic Services, OWL-S, Service Discovery, GHSOM, Latent Semantic Indexing.

1. Introduction

Web Services are Internet-based, distributed modular applications that adopt open standard interfaces and protocols in order to be used as basic software building blocks in service oriented applications. One of the most promising advantages of the service oriented paradigm is related to service discovery.

SOA supports a directory UDDI (Universal Description Discovery and Integration) in which service providers can publish their services which enables potential clients to find and invoke them over the internet. On the basis of WSDL (Web Services Description Language), UDDI is designed to function in a fashion similar to white pages or yellow pages.

Due to the lack of semantic description of the Web services, the search results returned by the service directories are effectively inadequate.

In this paper we extend our previous XML&Web Services framework [6] that addresses open issues in the development of a Web Services based application. Here we will focus on extending the semantic representation of services, grouping the similar Web Services and ranking them by relevance. The main objective of this research is to develop a more effective mechanism for semantic Web service discovery by combining GHSOM clustering method and Latent Semantic Indexing.

The organization of this paper is as follows: first, we briefly introduce OWL-S services and Neural Network Learning Methods. In section 3, the XML & Web Services Framework is discussed. The Semantically Service Discovery Approach is presented in section 4. A briefly discussion of the research work related to service discovery is given in section 5 and finally, the conclusion and future work can be found in section 6.

2. Introduction to OWL-S and Neural Network Learning Methods

In the following, we briefly introduce the essentials of the semantic Web service description language OWL-S, Self Organizing MAP (SOM) and Growing Hierarchical SOM.

2.1. OWL-S

OWL-S [17] is a specific OWL [16] ontology designed to provide a framework for semantically describing services. This semantic description of Web services will provide an ontology that allows software agents to discover, execute and compose Web services automatically.

OWL-S ontology consists of a service profile, a process model and service groundings.

Profiles: specifies web service descriptions based on their functional and nonfunctional parameters [16]. In service discovery, profiles are applied in two ways: advertisements and request.
During service discovery the Request is compared with Advertisement to find suitable services.

**Process Models:** is used to define process models that describe the execution of a Web service in detail. OWL-S defines three types of processes [2]: Atomic Process, Simple Process and Composite Process.

**Groundings:** enables communication between concrete Web service and abstract specifications of Web service characteristics (profiles and process models).

2.1. Self-Organizing Map (SOM)

SOM also known as Kohonen map is one of the most popular unsupervised neural network model [14]. It provides a mapping from high-dimensional feature spaces onto a two-dimensional space. SOM is also a clustering method where similar data items are represented as vectors of numerical values. SOM also illustrates the relationship among clusters by locating one cluster close to the other related clusters.

But SOM provides only flat, i.e. two-dimensional representation of document’s clusters, which might blind for interpretation when document collection is very large.

To overcome these limitations, GHSOM is developed.

2.2. Growing Hierarchical SOM

The key idea of the GHSOM [14] is to use a hierarchical structure of multiple layers where each layer consists of a number of independent self organizing maps. For every unit in this map, a self organizing-map might be added to the next layer of the hierarchy. The same principle might be applied to any further layer of the hierarchical future map.

This allows starting from a rather small high-level SOM which provides an overview of topics presented in the collection, where subsequent layers are added until a specific topic is presented in sufficient degree of granularity.

3. Overview of XML & Web Services Framework

The framework is illustrated in Fig.2. It concentrates on resolving major research challenges in the field of SOA [6]. XML & Web Services Framework consists of five layers.

### Application Layer

### Semantically Service Discovery Layer

### Web Service Layer

### Data Integration Layer- Query Engine

### Data Layer

**Figure2. XML & Web Services Framework**

**Data Layer:** presents the set of various resources like different databases, Text files, Xml Files etc… to be integrated.

**Data Integration Layer:** will present a unified interface of a collection of data to be integrated.

The integration will be by using wrappers and mediators in the following way:

The system will send the query to Semantic Query Analyzer Service. The query analyzers will search in the field of ontology and return the results to the Data Service. Data Service delivers XML Query Language to Mediator Service, which will integrate all the heterogeneous information from the different sources in Data Layer and send back to Data Service.

**Web Service Layer:** presents a collection of Data Services created in Data Integration Layer.
as well as composite services, created by composing Data Services.

Figure 3. Outline of the data integration [6]

Semantically Service Discovery Layer: will efficiently find Web services from Web Service Layer according to Application Layer requirements.

In our previous work [6] we have stated that various approaches can be used to locate Web Services, but research, at this layer, will be focused on the service discovery problem using clustering method and ontology-based method, where this two method will be combined, in order to establish semantic connections among the underlying composite services and Application Layer.

4. Semantically Service Discovery - SSD Architecture

The Service Discovery Layer is illustrated in Fig4. Semantic Web service discovery takes as input a user query, which is used as search criterion and returns a set of web services matching this query.

We first assume that OWL-S service descriptions will be made available by service providers and SSD can access a local service registry storing ontology descriptions (OWL-S) and WS-Descriptions (WSDL). The Local Registry is fetched from UDDI or uploaded directly by service providers.

Figure 4. Outline of Semantically Service Discovery Approach

The process of discovery consists of the following phases:

- A local register server will be created, so that all the web services which will be provided by the service providers will be stored at this register.
- The searching process will be initialized with a query typed by the user which will be used as search criteria.
- Applying GHSOM to the service set, in order to construct a self organizing semantic map for service retrieval.
- Utilizing Latent Semantic Indexing for scoring and ranking the documents by their relevance.
- Returning the achieved result in the user interface.
4.1. Overview of our SSD Approach

The overall process of discovering web services includes: web services information processing, clustering heterogeneous services according to semantic similarity and ranking web services by their relevance.

4.1.1 Web services information preprocessing

Our approach starts with the preprocessing of inputs (OWL-S files). Terms are extracted from scanning the OWL-S files where outputs are tokenized documents in XML format. We continue with interpreting all tokenized documents as numerical vectors suitable for neural network processing. For this purpose we use vector representation according to the Vector Space Model (VSM), obtaining in such a manner a service matrix for a service corpus [1].

4.1.2 Clustering heterogeneous Web services according to their semantic similarity

The second step is to design neural network (GHSOM), which is able to cluster the representation vectors, arranged as nodes in a hierarchy and is able to discover hierarchical clusters.

This is done by using the obtained service matrix for GHSOM training [14]. In the resulting map architecture the OWL-S files are grouped into various topical branches, with topics in each branch being arranged on two-dimensional maps and keywords serving as labels for the respective topics. Each hierarchy presents a group of the services related according to their semantic similarity.

Fig5. GHSOM and LSI in combination
4.1.3 Ranking web services by their relevance.

LSI ranking of web services begins with the construction of document term matrix and query column matrix.

The GHSOM is queried against the query vector and resulting neighborhood of query vector will be used for constructing a LSI term document matrix [1]. The LSI document matrix, in this case service matrix, will be decomposed and truncated to all required matrixes by using Singular Value Decomposition- SVD [1], which results in finding new coordinates of query and document vectors in the reduced k-dimensional space. The cosine similarity values are used for service ranking. That is, sorting services in decreasing order which in fact is our desired result.

5. Related Work

Recently, there have been a few proposals for Web services discovery based on OWL ontologies and Web services discovery by using clustering algorithms.

Work [5] proposed an approach of using ontologies to compute similarity degrees between terms of queries in multiple-source information systems.


The WSMO framework [17] provides ontology translation to support automatic interoperability between Web services.

The two approaches described in [3, 4] are both based on ontology concept to add semantic information into UDDI by means of tModels.

In [10], the discovery of services is addressed as a problem of matching queries specified using a variant of Description Logic (DL) with service profiles specified in OWL-S. The matching process is based on the computation of subsumption relations between service profiles and supports different types of matching (exact, plug-in, subsume, intersection, and disjoint matching).

The Web Service Modelling Framework - WSMF [5] aims at providing an industry scale framework for semantic web service discovery, execution and composition. WSMO is built on four key concepts: Ontologies, web services, goals and mediators. Ontologies are used to formally conceptualize properties and capabilities of web services.

Current approaches use GSOM and LSI separately, and as such they suffer from a number of drawbacks. When used GSOM technique alone, new services will be clustered and classified appropriately, however during the searching process, the result may be again enormous number of services, among which many could be irrelevant. On the other hand, when used LSI only, the technique will encompass all documents in resulting matrixes which in turn will be of large dimensions used by this technique. Working and transforming such matrixes to gain the appropriate web services will result in huge processing time, making the searching too slow. Therefore, we propose an integrative approach, where GSOM and LSI are used in combination, in order to gain relevant services effectively and efficiently.

6. Conclusion and future work

In this paper, we follow up on our previous work of proposing a new Xml&Web Services Framework in order to enable the user to use the framework to deploy a series of pre-developed functions as a web service. We have extended the Semantically Service Discovery Layer proposing a new method for efficiently service discovery.

Further research should concentrate on the building tool for evaluation of the effectiveness of our method and to embed the SSDL in UDDI registries.

9. References


