A Framework for Semantic Discovery of Web Services

Shalini Batra

Thapar University,
Patiala, Punjab, India
sbatra@thapar.edu

Seema Bawa

Thapar University,
Patiala, Punjab, India
seema@thapar.edu

ABSTRACT

Web services have become a new industrial standard offering interoperability among various platforms but the discovery mechanism is limited to syntactic discovery only. The framework named ADWebS is proposed in this paper for automatic discovery of semantic Web Services, which can be considered as an extension to one of the most prevalent frameworks for semantic Web service, WSDL-S. At the first stage, the framework proposes manual semantic annotations of Web service to provide the functional description of the services in Web Service Description Language(WSDL)'s < document > tag . These annotations are extracted and term-category matrix is formed, where category denotes a class in which a Web service will be added. Next, Semantic relatedness between terms and pre-defined categories is calculated using Normalized Similarity Score (NSS). A non-parametric test, Kruskal Wallis test, is applied on values generated and based on the test results, services are put into one or more pre-defined categories. The user or the requestor of the service is directed to the semantically categorized Universal Description Discovery and Integration (UDDI) repository for discovery of required service. Experimental results on a dataset covering multiple Web services of various categories show a significant improvement over the current state-of-the-art Web service discovery methods.

Categories and Subject Descriptors

General Terms

Design

Keywords

Semantic Web Services; Semantic Web Services Discovery; Normalized Similarity Score; Semantic Annotations

1. INTRODUCTION

Web services are modular, self-describing, self contained applications that are accessible over the Internet. The fundamental premise of Web services is that standardization, predicated on the promise of easy interoperability, resolves many of the long-standing issues business world is facing today. The main pieces coming together for success of Web service include the loosely coupled Web riding on waves of SOAP and XML. WSDL, is used to describe operations and the formats of the input and output messages. Being based on these Web standards makes these services both implementation language and platform independent.

Service providers describe their Web services and advertise them in a universal registry called UDDI. UDDI allows for the creation

of registries that are accessible over the Web. A registry contains content from the WSDL descriptions as well as additional information such as data about the provider. Clients may use one or more registries to discover relevant services [1].

The discovery of suitable Web services for a given task is one of the central operations in Service-Oriented Architectures (SOA), and research on Semantic Web Services (SWS) aims at automating this step. The interaction process for Web services consists of three distinct phases: discovery of possible services, the selection of the most useful service, and subsequent execution. The first two phases are more crucial and hence the focus in this paper is on the discovery and selection activity that is needed to identify the required Web service(s).

Major drawback of current description standards for Web services is that to a large extent, they are restricted to the syntactic aspects of service interaction. They allow the description of how the service can be invoked, which operations may be called, which policies may be supported, etc. However, what the service does and in what order its operations have to be called to achieve certain functionality is only, if ever, described in natural language in the entries or comments of a WSDL description or UDDI entry. Thus, bottlenecks faced when searching for suitable services to achieve a certain task are similar to those faced as in filtering information from the huge amount of data on the current human-readable "static" Web.

The capabilities offered by UDDI for service discovery are rather limited. The lack of machine understandable semantics in the technical specifications and classification schemes used for retrieving services, prevent UDDI registries from supporting fully automated and thus truly effective service discovery. Semantic Web services are Web services whose "properties, capabilities, interfaces, and effects are encoded in an unambiguous, and machine-interpretable form" [2]. Groszof states semantic Web services includes both the infrastructural and the application-specific services and that the term "Semantic Web services" can be parsed as "{SemanticWeb} Services" (e.g., for relatively broad-purpose knowledge translation and inference) or as "Semantic {WebServices}" (e.g., knowledge based service descriptions dealing with discovery, composition, invocation, monitoring, etc.)[3]. Sheth, argues that semantics play an important role in the complete lifecycle of Web services [4].

Three factors which mainly affect service discovery are:

- a) Ability of service providers to describe their services,
- b) Ability of service requestors to describe their requirements, and
- c) “Intelligence” of the service matchmaking algorithm.

1.1 Role of Semantics in Life Cycle of a Web Service

Contextual information that establishes relationships between the data and the real world aspects it applies to is called metadata. Thus, metadata is data that describes information about a piece of data, thereby creating a context in terms of the content and functionality of that data. The process of associating metadata with resources (audio, video, structured text, unstructured text, web pages, images etc) is called annotation and semantic annotation is the process of annotating resources with semantic metadata. Semantic annotations can be coarsely classified as being formal or informal. Formal semantic annotations follow representation mechanisms, drawing on conceptual models represented using well defined knowledge representation languages. Such machine processable formal annotations on Web resources can result in vastly improved and automated search capabilities, unambiguous resource discoveries, information analytics, etc. To semantically annotate a Web service means explicating the exact semantics

of the Web service data and functionality elements that are crucial towards the use of the Web service. This is done by annotating the Web service elements with concepts in domain models or ontologies. This enables unambiguous and automated service discovery and composition.

2. Related Work

Web service discovery is normally defined as a matching process in which available services' capabilities can satisfy a service requester's requirements. The capability of a Web service is often implicitly indicated through a service's name, a method's name and some descriptions included in the service and it can be described as an abstract interface by using standard WSDL. With the help of the standard descriptions of Web services, various approaches can be used to find services on the Web, such as using Web search engines like Google, Yahoo, etc, service portals [5] and service registries like UDDI [6].

As Web Services discovery is an important and difficult task in the development cycle of service oriented application, many algorithms, tools and mechanisms are proposed to solve this problem. WS Discovery mechanisms include a series of registries, indexes, catalogues, agent based and Peer to Peer-P2P solutions. Researchers have proposed a number of ideas to enhance the accuracy of Web service discovery by applying data mining approaches [8] [7], singular vector decomposition [9], graph based methods, various ontology based discovery frameworks, agent- based, logic based methods and others [10],[12],[13].

Generally, information matching can be accomplished on two levels:

- In syntactic matching the similarity of data is found using syntax driven techniques. Usually, the similarity of two concepts is a relation with values between 0 (completely dissimilar) and 1 (completely similar).
- In semantic matching the key intuition is the mapping of meanings. There are several semantic relations of two concepts: equivalence more general less general, mismatch and overlapping. Nevertheless, they can be mapped into a relation with values between 0 and 1 [11].

The main foundation of the investigation related to semantic matching of services is in the studies of Software Engineering that Zaremski and Wing carried out in [14],[15]. There have been few proposals for Web services discovery based on OWL ontologies [16] and many researchers have proposed the usage of ontology [17], [18], [19] to annotate the elements in Web services. By modeling Web services with ontologies ,the semantic representation of concepts and their relations can be exploited and thus semantic matching can be performed. Semantic descriptions of Web Services can be obtained with the use of DAML-S [11] or OWL-S [21] languages. Paolucci et. al. [18] present a framework to allow WSDL and UDDI perform semantic matching where Web Services are modeled as ontologies, or Service Profiles as they are called, with the use of the DAML-S. Hess and Kushmerick [19] suggest the use of machine learning to generate suggestions for annotating Web services. In a related effort, Patil and colleagues have developed MWSAF, a Web service annotation framework [22]. In their work, they generate recommendations for automatically annotating WSDL documents. To accomplish this they match XML schema used by the WSDL files with ontologies by creating canonical schema graphs. A survey of semantic annotations platforms is presented by Reeve and Han [23]. Some recent work by Paolucci has proposed annotating web services manually with additional semantic information, and then using these annotations to compose services [1].

Regarding the Web as a live corpus has become an active research topic recently and much work has been carried out on measuring semantic similarity using Web content. Simple, unsupervised models demonstrably perform better when n-gram counts are obtained from the Web rather than from a large corpus [24] [25]. Resnik and Smith [26] extracted bilingual sentences from the Web to create a parallel corpora for machine translation. Turney [27]defined a point-wise mutual information (PMI-IR) measure using the number of hits returned by a Web search engine to recognize synonyms. Matsuo et. al, [28] used a similar approach to measure the similarity between words and apply their method in a graph-based word clustering algorithm. Author proposed the use of Web hits for extracting communities on the Web. They measured the association between two personal names using the overlap (Simpson) coefficient, which is calculated based on the number of Web hits for each individual name and their conjunction (i.e., AND query of the two names).

Chen et al. [29] have proposed to exploit the text snippets returned by a Web search engine as an important measure in computing the semantic similarity between two words. Danushka Bollegala et al. [30] has proposed a method that exploits page counts and text snippets returned by a Web search engine to measure semantic similarity between words. Cilibrasi and Rudi [31] developed the method that defines the relatedness between the words via Google Similarity Distance. They have proposed to compute the semantic relatedness using the normalized google distance (NGD), in which they used GoogleTM. to determine how closely related two words are on the basis of their frequency of occurring together in Web documents. They use the World Wide Web as the database and Google as the search engine. Salahli [32] use the related terms of two words to determine the semantic relatedness between the words.

3. Proposed Framework ‘ADWebS’

Automatic Discovery of Web Services Semantically (ADWebS) proposes a relational similarity measure that uses aWeb search engine to measure the similarity between implicitly stated

semantic relations in two word-pairs. Formally, given a pre-defined set of categories i.e. (C1, C2, C3,Cn) and set of terms in each Web service (t1,t2,t3,...,tn), a term-category matrix is designed by calculating the NSS of each term with every category, that returns a similarity score in the range [0, 1]. The proposed relational similarity measure first finds semantic similarity between each category and the term, and then ranks the services into a predefined category using Kruskal Wallis Test (Shown in Figure 1).

Figure 1: Top view of the Proposed Framework ‘ADWebS’

3.1 Steps Followed in ADWebS

First step, in the proposed framework ADWebS, is addition of functional descriptions of the Web services in documentation tag of WSDL. It will be mandatory for every service publisher to give a set of n terms, which best describe the service functionality, i.e. $N = n_1, n_2, n_3, \dots$ where n_1, n_2, n_3 , etc. are the terms describing the service or, in other words, N is the set correspond to most probable query terms for the Web service published. Next step is to extract the terms from the WSDL file and generate the term-category matrix.

Definition: Term-Category Matrix: A term category matrix, $Q = V(C_{ij})$; $1 \leq i \leq m$; $1 \leq j \leq n$, refers to a collection of category term-values for a set of candidate services, such that, each row of the matrix corresponds to the value of a particular term (defined in Web service) and each column refers to a particular category. In other words, $V(C_{ij})$, represents the value of the i th term’s NSS for the j th category. These values are obtained from the terms of the candidate service providers and services are mapped to a particular category or categories.

Given a Web service with most useful terms in the document tag, this part of the WSDL of a Web Service is extracted and term-category matrix is generated. Once the term-category matrix is generated semantic relatedness of each term is calculated with each category using Normalized Google Distance (NGD) and Normalized Similarity Score (NSS).

3.2 Calculating Normalized Similarity Score (NSS)

Given a text corpus, individual words have more or less differing contexts around them. The context of a word is composed of words co-occurring with it within a certain window around it. Distributional measures use statistics acquired from a large text corpora to determine how similar the contexts of two words are. These measures are also used as proxies to measures of semantic similarity as words found in similar contexts tend to be semantically similar. This is known as the distributional hypothesis [34] and such measures have traditionally been referred to as measures of distributional similarity. Budanitsky and Hirst [35] point out that if two words have many co-occurring words then similar things are being said about both of them and so they are likely to be semantically similar. Conversely, if two words are semantically similar then they are likely to be used in a similar fashion in text and thus end up with many common co-occurrences. For example, the semantically similar *bug* and *insect* are expected to have a number of common co-occurring words such as *crawl*, *squash*, *small*, *woods*, and so on, in a large enough text corpus.

Page-count-based metrics are used by search engines to find use association ratios between words that are computed using their co-occurrence frequency in documents. The basic assumption of page count-based metrics is that high association ratios indicate a semantic relations between words[36].

Motivated by Kolmogorov complexity, Cilibrasi and Vitanyi [31] proposed a page-count-based similarity measure, called the Normalized Google Distance (NGD), defined as:

$$NGD(x, y) = \frac{\max\{\log f(x), \log f(y)\} \log f(x, y)}{\log M \min\{\log f(x), \log f(y)\}} \quad (1)$$

where M is the total number of Web pages searched by Google; $f(x)$ and $f(y)$ are the number of hits for search terms x and y , respectively; and $f(x, y)$ is the number of web pages on which both x and y occur.

If the two search terms x and y never occur together, but do occur separately, the normalized Google distance between them is infinite, meaning they are not semantically related at all. In another extreme case where both terms always occur together, their NGD is zero, which is an indicator that they are semantically closely related. While theoretically the scope of NGD is [0,1] experimental results show that most of the time the values fall between 0 and 1 [31].

Subsequently, the normalized distance values are converted into similarities called Normalized Similarity Score(NSS):

$$NSS(x, y) = 1 - NGD(x, y) \quad (2)$$

Calculating the NSS of all terms with all categories means that semantic relatedness of each term with each category has been found. So, the next step is to add the Web service into one of the predefined category.

To categorize Web Services into one or more predefined categories Kruskal-Wallis Test has been used in ADWebS.

Soft Categorization Vs Hard Categorization

After applying Kruskal- Wallis test it is clear that depending upon the maximum value achieved for a category, the Web Service will be placed in that particular slot. At this point there are two alternates: hard categorization or soft categorization i.e. a service can belong to only one category (Hard categorization) or it can belong to more than one categories (Soft categorization). If it is assumed that a Web service W1 is categorized to Category C1 using Kruskal-Wallis test and the terms of that Web service W1 are also semantically closer to Category C2 (predicted by observing the values generated through Krushkal Wallis test) it would not be optimal to totally ignore all the other categories. So, soft categorization has been considered in this work i.e. a Web service can be fall in more than one category and criterion used is Eq. (3).

Based on Krushal Wallis test, let Web service WS1 is categorized in category C1 since C1 has maximum value. Range fixed to categorize a Web service into more than one category is fixed to 25 %of the maximum value achieved for terms extracted from WS1 i.e.

$$WS_{Range} = 25\% (C_{max}) \quad (3)$$

Hence, all categories which have values lying in the limit of WSRange will register the Web service WS1 in their category too and a Semantic UDDI repository is generated.

3.3 Ranking the Services of a Particular Category

Basic assumption for user's request is that it is targeted to semantic UDDI repository where Web services have been categorized beforehand to a particular set, thus user's query is limited to a particular category i.e. user selects one of the category from the set of pre defined-categories. To rank a particular request to a set of Web services of a selected category i.e.to match complex WSDL given their similarity scores corresponding to their functional annotations, the matching average strategy is used.

4. Empirical Evaluations

A collection of Web services has been retrieved from X-Methods, free UDDI repository and WSDL files are annotated with the terms giving the functional description of the Web Service considered . Terms have been extracted from document tag of the published WSDLs and put in a file along with the URL of the Web Service to which they belonged and the NSS of each term is calculated with all categories .In this proposed work five categories have been considered: *Zip Code, Stock Market, Weather, Country Information and Currency*.

Table 1 shown below has five categories represented in columns and six terms latitude, longitude, humidity, temperature, wind speed and pressure represented in rows, extracted for Web service say WS1. This value varies between 0 and 1 and more is the value, closer is the association of a word to the respective category.

Table 1. NSS of all Categories with all terms

Category / Words	Zip Code	Country info.	Stock Market	Weather	Currency
Pressure	0.563	0.368	0.566	1	0.902
Temperature	0.151	0.684	0.252	0.513	0.776

Wind speed	0.374	0.03	0	0.745	0.327
Rainfall	0.155	0.302	0	0.737	0.584
Country	0.755	0.396	0.246	0.983	0.804
City	0.797	0.352	0.564	0.935	0.805

Now Kruskal Wallis test is applied and values achieved clearly put WS1 under the category weather and hence this Web service will fall under the category weather . On similar lines a Web service which will have most words related to stock markets will obviously have maximum NSS for stock market category. Since value achieved for category Currency lies in the range of C_{max} , URL of WS1 will also be added in category Currency as per Eq 3.

5. Principles Guiding WSDL-S and ADWebS

The WSDL-S charter [37] recommend that certain principles guide any work to define a framework for Web services semantics. WSDL-S is guided by the following principles:-

i) Build on existing Web Services standards:

In WSDL-S it is believed that any approach to adding semantics to Web Services should be specified in an upwardly compatible manner so as to not disrupt the existing install-base of Web Services. Considering this principle as the most important consideration ADWebS framework too has been designed on the existing structure of Web services which includes WSDL, SOAP, UDDI and XML. The WSDL structure including interface, operation, message, binding, service and endpoint remain intact and functional description is provided in the document tag of WSDL.

ii) The mechanism for annotating Web services with semantics should support user's choice of the semantic representation language:

There are a number of potential languages for representing semantics such as OWL, WSML, and UML. Each language offers different levels of semantic expressivity and developer support. By keeping the semantic annotation mechanism separate from the representation of the semantic descriptions, the approach in WSDL-S offers flexibility to the developer community to select their favorite semantic representation language. On similar principles, users in ADWebS have been provided full flexibility in providing any number of terms i.e. publisher can even add multiple terms to represent the same concept if query is expected from a sysnet.

iii) The mechanism for annotating Web services with semantics should allow the association of multiple annotations written in different semantic representation languages:

Designers of WSDL-S believe that the mechanism for annotating Web Services with semantics should allow multiple annotations to be associated with Web Services. This principle is already cover in second point as publisher has full choice of adding multiple annotations in ADWebS and it is depended only on the search engine to accept different semantic representation languages.

iv) Support semantic annotation of Web Services whose data types are described in XML schema:

WSDL-S design believes that the semantic annotation of service inputs and outputs should support the annotation of XML schemas. WSDL 2.0 supports the use of other type systems in addition to XML Schema, so constructs in semantic models, such

as classes in OWL ontologies, could be used to define the Web service input and output data types. However, an approach that does not address XML schema-based types, will not be able to exploit exiting assets or allow the gradual upgrade of deployed WSDL documents to include semantics. In ADWebS full support has been provided for semantic annotation of Web services and XML schema-based types have been allowed.

v) Provide support for rich mapping mechanisms between Web Service schema types and ontologies:

Giving the importance of annotating XML schemas in Web service descriptions, WSDL-S proposes that attention should be given to the problem of mapping XML schema complex types to ontological concepts. Again, an agnostic approach to the selection of schema mapping languages is called for. For example, if the domain model is represented in OWL, the mapping between WSDL XSD elements and OWL concepts can be represented in any language of user's choice. Among the choices are meta representation languages such as: RDF, OWL, SPARQL and MOF or XML-based transformation languages such as XSLT, XQuery or any other language as long as the chosen language is fully qualified with its own namespace. The major difference between WSDL-S approach and ADWebS approach is that ADWebS does not relies on ontologies instead it considers search engine as knowledge corpus and relies on distributional hypothesis for semantic discovery of annotated Web services. In ADWebS correlation has been established between terms and pre defined categories by finding the normalizing co occurrence ratio, a measure of semantic relatedness, so rich mapping has been provided between the functional description of Web Services and pre defined categories. Based on this categorization, Semantic UDDI has been generated.

6. Advantages of ADWEBS over WSDL-S

Two major benefits foreseen are:

i) No use of domain ontologies.

Usage of domain ontologies as knowledge corpus has some inherited problems: Although standard vocabularies like ontologies provide semantic descriptions they don't adapt well for heterogeneous domains like Web services since a very broad coverage vocabulary is required for such domains. Gruber [39] has clearly stated that "It is impossible to develop a single ontology that fully covers a domain or will satisfy the needs and preferences of each user". Secondly, design of ontology is dependent on the representation language's expressiveness and if the language is weak then the designed ontology will also be weak. Finally, since there is no standard methodology pre-defined for development of ontologies, many times it is difficult to understand the intended meaning of concepts and the associations existing between the interrelated concepts of the ontology.

ii) Use of World Wide Web as Knowledge Corpus
Web services carry out operations to support a real-world service, e.g., the ordering of goods. Thus, Web services exist on the boundary of the world inside an information system and the external world. Lexical resources of knowledge cannot efficiently find semantic relatedness between concepts coming from millions of Web users and diverse text corpus. Web search engines can serve as an efficient interface to extract semantics since its sheer mass of users and documents with different intentions averages out to give the true semantic meaning used by people world over. Since there is no doubt that the relative page counts approximate the true societal word- and phrases usage and Google is an able

extractor, Normalized Google Distance is used as a measure of semantic relatedness between terms and categories.

7. Conclusions

The general understanding is that ADWebS approach will give more meaning full and technical annotations to the Web services in hand. Major advantage foreseen by applying this approach is that it will overcome the heterogeneity in the data representation. Since the developer or publisher of a Web Service is the best judge of his service functionalities and capabilities, terms most closely to Web service functionality will be provided in the documentation part of WSDL which can be easily extracted, which will serve as the input dataset for grouping the similar Web services. Another major advantage is that there is no need for following usual preprocessing steps which include detagging, stop word removal, stemming etc as the terms are directly extracted from the WSDL's Document tag without any repetition of same words. Because of the vastly numerous documents and the high growth rate of the Web, it is difficult to analyze each document separately and directly. Web search engines provide an efficient interface to this vast information. The need of the hour is to extracts semantics as it were the semantics used in the society (of all these web users) and not just the bias of any individual user or document.

This is only possible if semantics is extracted using the Web, since its sheer mass of users and documents with different intentions averages out to give the true semantic meaning as used in society. Our contention is that the Web is such a large and diverse text corpus, and Google such an able extractor, that the relative page counts approximate the true societal word- and phrases usage, starts to be supported by current real linguistics research.

References

- [1] Cardoso,J.:Semantic Web Services: Theory, Tools, and Applications: IGI Global (2007)
- [2] McIlraith, S.,Son T., Zeng,H.: Semantic Web Services. In: IEEE Intelligent Systems, 16(2), pp.46-53 (2001)
- [3] Groszof., B.:Semantic Web Services: Obstacles and Attractions.In: 12th International Conference on World Wide Web, Budapest, Hungary (2003)
- [4] Sheth., A.: Semantic Web Process Lifecycle: Role of Semantics in Annotation, Discovery, Composition and Execution. In: Invited talk at WWW Workshop on e-Services and the Semantic Web, Budapest, Hungary(2003)
- [5] XMethods.<http://www.xmethods.com/>
- [6] T. Bellwood et al.: UDDI version 3.0 <http://uddiorg/pubs/uddi-v3.00-> published-20020719.htm (2002)
- [7] Nayak R.: Using Data Mining In Web Services Planning, Development and Maintenance. International Journal of Web services Research, vol. 5, pp. 62-80 (2008)
- [8] Nayak R., Lee B.: Web Service Discovery with additional Semantics and Clustering. In: IEEE/WIC/ACM International Conference on Web Intelligence (WI'07), Silicon Valley, USA (2007)
- [9] Sajjanhar, A., Hou, J., Zhang, Y.: Algorithm for Web Services Matching. In: Proc. APWeb 2004, LNCS vol. 3007, pp 665-670, Springer Verlag, (2004)

- [10] Yu J., Su H., Zhou G., and Xu K.: SNet: Skip Graph based Semantic Web Services Discovery. In: ACM Symposium on Applied computing, Seoul, Korea pp. 1393-1397 (2007)
- [11] Kokash, N., Birukou, A., D'Andrea, V.: Web Service Discovery Based on Past User Experience. In: the proceeding of 10th International Conference on Business Information Systems, Poznan, Poland. pp. 95- 107 (2007)
- [12] Al-Masri E. and Mahmoud Q. H.: QoS-based Discovery and Ranking of Web Services. In: IEEE 16th International Conference on Computer Communications and Networks (ICCCN) pp. 529-534 (2007)
- [13] Bose, A., Nayak, R., Bruza, P.: Improving Web Service Discovery by using Semantic Models. In: Bailey, James and Maier, David and Thalheim, Benhard, Eds. Proceedings 9th International Conference on Web Information Systems Engineering (WISE 2008) 5175/2008, Auckland, New Zealand pp. 366-380 (2008)
- [14] Zaremski Moormann A., Wing Jeannette M.: Signature Matching: A Tool for Using Software Libraries. In: ACM Transactions on Software Engineering and Methodology (TOSEM), Vol. 4, No. 2, (1995).
- [15] Zaremski Moormann, A., Wing Jeannette M.: Specification Matching of Software Components. In: ACM Transactions on Software Engineering and Methodology (TOSEM), Vol. 6, No. 4, pages 333-369 (1997)
- [16] Maximilien, E., Singh, E.: A Framework and Ontology for Dynamic Web Services Selection. In: IEEE Internet Computing, 8(5), pp 84-93 (2004)
- [17] Sivashanmugam, K., Verma, K., Miller J.A.: Adding Semantics to Web services Standards. In: Proceedings of the International Conference on Web services ICWS'03 pp 395-401 (2003)
- [18] Paolucci, M., Kawamura, T., Payne, T., Sycara, K.: Semantic Matching of Web Services Capabilities. In: The First International Semantic Web Conference (ISWC)(2002)
- [19] Hess, A., Kushmerick, N.: Learning to Attach Semantic Metadata to Web Services. In: 2nd International Semantic Web Conference (ISWC2003), Sanibel Island, Florida, USA (2003)
- [20] Ankolekar, A., Burstein, M., Hobbs J., et. al.: DAML-S: Semantic Markup for Web Services. In: Proceedings of the International Semantic Web Working Symposium (SWWS), July 30-August 1 (2001)
- [21] Martin, D., et al.: OWL-S: Semantic Markup for Web Services. In: W3C Member Submission, 22 November 2004. Available from <http://www.w3.org/Submission/OWL-S/>.
- [22] Verma, K., Sivashanmugam, K., Sheth, A., Patil, A., Oundhakar S., Miller, J.: METEOR-S WSDI: A Scalable Infrastructure of Registries for Semantic Publication and Discovery of Web Services. J. of Information Technology and Management, Special Issue on Universal Global Integration, Vol. 6, No. 1 pp. 17-39. Kluwer Academic Publishers (2005)
- [23] Reeve, L., Han H.: Survey of Semantic Annotation Platforms. In: Proceedings of the ACM symposium on Applied Computing (2005)
- [24] Keller U., Lara, R., Lausen, H., Polleres, A., Fensel, D.: Automatic Location of Services. In: Proceedings of the 2nd European Semantic Web Conference, LNCS 3532, (2005).
- [25] Lapata, M., Keller, F.: Web-based Models of Natural Language Processing. In: ACM Transactions on Speech and Language Processing, 2(1):131, (2005)
- [26] Resnik P., Smith, N. A.: The Web as a Parallel Corpus. In: Computational Linguistics, 29(3):349-380, (2003)
- [27] D. Turney: Similarity of Semantic Relations. Volume 32, Number 3. in: Association for Computational Linguistics. (2006)
- [28] Matsuo, Y., Sakaki, T., Uchiyama, K., Ishizuka, M.: Graph-Based Word Clustering using Web Search Engine. In: Proceedings of EMNLP (2006)
- [29] Chen, H., Lin, M., Wei, Y.: Novel Association Measures using Web Search with Double Checking. In: Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics pp. 1009-1016 (2006)
- [30] Bollegala, D., Matsuo, Y., Ishizuka M.: Measuring Semantic Similarity between Words Using Web Search Engines. In: International World Wide Web Conference Committee (IW3C2) WWW 2007, May 8-12, 2007, Banff, Alberta, Canada. ACM 9781595936547
- [31] Cilibrasi R., Vitanyi, P.: The Google Similarity Distance. In: IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 3, pp. 370-383, (2007)
- [32] Salahli M.: An Approach for Measuring Semantic Relatedness between Words via Related Terms. In: Mathematical and Computational Applications, Vol. 14, No. 1, pp. 55-63, . Association for Scientific Research. (2009)
- [33] Xin Dong, Halevy, A., Madhavan, J., Nemes, E., Zhang, J.: Similarity Search for Web Services. In: Proceedings of the 30th VLDB Conference, Toronto, Canada, (2004)
- [34] Firth J. R.: A Synopsis of Linguistic Theory 1930-55. In: Studies in Linguistic Analysis (special volume of the Philological Society). Oxford, pp. 1-32, The Philological Society (1957)
- [35] Budanitsky, A., Hirst, G.: Semantic Distance in WordNet: An Experimental, Application-oriented Evaluation of Five Measures. In: Workshop on WordNet and Other Lexical Resources, in the North American Chapter of the Association for Computational Linguistics (NAACL-2000), Pittsburgh (2001)
- [36] Church K., Hanks, P.: Word Association Norms, Mutual Information and Lexicography. In: Computer Linguist., vol. 16, no. 1, pp. 22-29 (1990)
- [37] Akkiraju R., Farrell J., Miller J., Nagarajan M., Schmidt M., Sheth A., Verma K.: Web Service Semantics - WSDL-S. W3C Member Submission 7 November 2005 (2005)
- [38] <http://www.w3.org/Submission/WSDL-S/>
- [39] Tom R. Gruber: A translation approach to portable ontologies. In: Knowledge Acquisition, 2(5):199-220 (1993)