International Journal of the Physical Sciences Vol. 6(18), pp. 4466-4472, 9 September, 2011
Available online at http://www.academicjournals.org/IJPS
DOI: 10.5897/IJPS11.323
ISSN 1992 - 1950 ©2011 Academic Journals

Full Length Research Paper

Semantic discovery of web services using principal component analysis

Shalini Batra* and Seema Bawa

Computer Science and Engineering Department, Thapar University, Patiala, Punjab, India.

Accepted 20 July, 2011

With the increase in the number of available web services, searching for appropriate web services fulfilling the service discoverer's functional requirements has become as major challenge. Web services standards, in their present format support only keyword based search and many services which can fulfill the user's requirements are not retrieved. Basic requirement for efficient service discovery is to extract the contextual information provided in the service description. In such situation, there are two options either change the present service standards completely or introduce semantics in the present Web Service Description Language (WSDL). In this paper, we propose a novel method of introducing semantic in WSDL and classifying the services into set of pre-defined domains(categories) utilizing the available semantic information. Classification of services in specific domains will reduce the number of recommended services leading to decrease in service discoverer's search efforts. The results achieved using the proposed approach showed significant precision of the discovered services.

Key words: Semantic web services, semantic web services discovery, normalized similarity score, semantic annotations.

INTRODUCTION

Exponential growth of Web services availability will definitely increase the discoverer's efforts as finding the potential services will become more difficult and tedious. To overcome this problem one of the possible solution is to confine the available services to a specific domain or category. Retrieval of services from Universal Description, Discovery and Interface (UDDI) (Bellwood et al., 2002) will be more efficient if:-

1. Semantic information is available in the service itself and
2. The available semantic information can be utilized to allocate a service to the specific predefined domain (category). Therefore, two major issues which need an important consideration for successful semantic discovery of Web services are:-
1. How to add semantics in the present descriptions of the services?
2. How to allocate services into domains?

Service descriptions are provided in the WSDL in the form of inputs, output, interface and binding, and extracting conceptual information from these descriptions is a cumbersome job. Ontology has emerged as the most efficient candidate to capture the required semantic information. Even the most prevalent semantic Web service discovery frameworks like WSMO (Bruijn et al., 2005), OWL-S (Martin et al., 2004) and WSDL-S (Akkiraju et al., 2005) are opting for ontologies for conceptual matching. Although ontologies provide the required semantic structures but some of the problems with ontology are that there can never be a universal method for defining ontology and, ontology alignment, mapping and matching is not an easy job especially if cross platform ontologies are considered. Crasso et al. (2010) discuss eight shortcomings in the present Web service description and have termed them as 'anti-patterns' which prevent the efficient discovery of the services. They have experimentally shows that discovery is more accurate if all such 'anti-patterns' are removed. We believe that service discovery can be enhanced further if semantic annotations are provided within the service descriptions. Our paper proposes a novel approach for providing semantic descriptions within in the

*Corresponding author. E-mail: sbatra@thapar.edu.

WSDL and allocating the services to the specific domains by extracting the available information.

RELATED WORK

Machine learning techniques can be used to build classifiers for a category by observing the characteristics of a set of documents or corpus. Laura et al. (2011) propose grouping of results of traditional search engines into various categories using semantics techniques and ontologies available on Web. Subramaniam et al. (2010) study various classifiers like Naïve Bayes (Sahami et al., 1998), SVM (Vapnik et al., 1999), Neural Net (McCulloch and Pitts, 1943), etc. for categorization of emails. Similar techniques can be applied to classify Web services into different domains or categories.

Semantic annotation is considered as a promising technology to add and manage the knowledge associated with a set of resources. Annotating specific domains with accuracy from an automatic or semiautomatic viewpoint has raised a challenge for the current state of the art of semantic technologies (Gómez et al., 2011). To adequately exploit the capacities of the Web services, it is necessary to provide semantic annotation of its contents. Hess and Kushmerick (2003) suggest the use of machine learning to generate suggestions for annotating Web services. Patil et al. (2005) developed MWSAF, a Web service annotation framework where recommendations are generated for automatically annotating WSDL documents. According to Fenza et al. (2008) semantic annotation help in overcoming interoperability limitations and in fact enhance interpretation of service capabilities. Bai and Liu (2011) propose matching of semantic annotations of a Web service using fuzzy set theory. Many researchers use Web as knowledge base to find similarity between related words. Such approaches can also be used for finding semantic similarity between Web services or to classify services into various domains by determining contextual information in the services and domains. Turney (2006) define 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. (2006) use a similar approach to measure the similarity between words and apply their method in a graph-based word clustering algorithm. Chen et al. (2006) compute semantic similarity between two words using text snippets returned by a Web search engine. Bollegala et al. (2007) combine both page counts and text snippets returned by a Web search engine to measure semantic similarity between words. Cilibrasi and Rudi (2007) compute semantic relatedness using Google Similarity Distance, called the Normalized Google Distance (NGD), where Google™ is used to determine similarity between two related words by counting their frequency of occurring together in Web documents. Salahli (2009) use the related terms of two words to determine the semantic relatedness between the words. Web services similarity is defined using a WordNet-based distance metric by Wu and Wu (2005). A web service matcher used by Zhuang et al. (2005) assign a similarity score to matching elements between two WSDL documents using Web as a live corpus. Stroulia and Wang (2005) employ WordNet to expand the query and WSDL files with the synonyms, direct hypernyms, hyponyms, and siblings senses, and the syntactic similarity between the WordNet-powered VSM feature vectors and the semantic distances between the identifiers of the WSDL files was calculated. Kokash (2006) propose combination of syntactic, semantic and structural similarities of different elements in a single measure to ameliorate retrieval performance. Many authors have proposed the use of external resources such as thesaurus and dictionaries to perform the task of service matching (Aleksovski et al., 2006a; Aleksovski et al., 2006b; Zhang and Bodenreider, 2005; Giunchiglia et al., 2004). Gligorov et al. (2007) proposed the idea of approximate ontology mappings by using Google based similarity measure, that is, Google distance as a weighting heuristic. Once the semantic relatedness is found, some statistical metrics can be applied to allocate the service into one of the predefined domains. One such metrics is principal component analysis (PCA). It is probably the oldest and best known of the techniques of multivariate analysis. It was first introduced by Pearson (1901), and developed independently by Hotelling (1933). Computation of the principal components reduces to the solution of an eigenvalue-eigenvector problem for a positive-semidefinite symmetric matrix.

The approximate structure and variances of the first few PCs can be deduced from a correlation matrix, provided that well-defined groups of variables are detected, including possibly single-variable groups, whose within-group correlations are high, and whose between-group correlations are low (Friedman and Weisberg, 1981; Jackson, 1991; Jolliffe, 2002).

PROPOSED METHOD

We propose addition of semantic data in the present WSDL by annotating the Web services using an additional tag (Batra and Bawa, 2010b). Once the terms defining the service capabilities are available, the next important issue is how to find semantic similarity between the semantic annotations and different domains. Since ontology approach suffer from some serious bottlenecks, we propose using Normalized Similarity Score (NSS), and Measures of Semantic Relatedness (MSRs) for assessing the semantic similarity (Batra and Bawa, 2009, 2010a).

NSS (k)

[k is the total number of records in the database, LC is the list of categories]

1. Set N = T_r [T_r is number of terms in the service considered]
2. Do for each Category C[i] in LC
3. Set j = 0
4. Do for each terms T_r[j]
5. V[i][j] = NSS (C[i], T_r [j])

6. Set j = j+1 [End of for loop]
7. Set i = i+1 [End of for loop]

Algorithm 1: Algorithm to find normalized similarity score (NSS) between terms in WSDL's annotation tag and categories

Once the semantic relatedness is calculated, the next important issue is how to classify the services into respective categories. If services are added into different categories, the entire discovery process is reduced to finding on of the candidate category from the list of categories. This will reduce discoverer's search time and increase the search efficiency as the set of recommended services will decrease drastically.

Classification of services in to candidate category

According to Crasso et al. (2008), categorization in Web services is done by providers either by manually assigning a category to their services from a number of predefined options such as the United Nations Standard Products and Services Code (UNSPSC) and the North American Industry Classification System (NAICS) or discoverers may look up third-party services by manually browsing categories or performing keyword-based search. In the proposed approach services are categorized into one or more predefined category(s) by a statistical metric called principal component analysis (PCA). PCA is a linear transformation technique which involves a mathematical procedure that transforms a number of possibly correlated variables into a smaller number of uncorrelated variables called principal components. Eigen vectors are calculated to define principal component weights, and eigen values represent variances of principal components. The first principal component accounts for as much of the variability in the data as possible, and each succeeding component accounts for as much of the remaining variability as possible (Details of experiment provided in 'preliminary evaluation'). Using this metric, a service can be added to more than one category using soft categorization as discussed in Batra and Bawa (2010b)

PRELIMINARY EVALUATION

Experimental dataset

Five categories *zip code, stock market, weather, country information and currency* have been considered as candidate categories and eight terms *city, temperature, pressure, humidity, precipitation, visibility, clouds, wind speed, wind direction latitude, longitude and elevation* are extracted from annotations tag of WSDL.

Evaluation metric

To evaluate the proposed approach's effectiveness and efficiency, its precision and recall parameters are compared. The precision ratio is the fraction of WSDLs retrieved relevant to user needs (precision = number of relevant operations/ total number of operations) and measures how well the approach rejects irrelevant services. The recall ratio is the fraction of query-related WSDLs successfully retrieved (recall = number of relevant operations / total number of relevant operations) and measures how well relevant services are found. For each experiment, we built a query and examined the WSDL documents of the retrieved set. While there are some different methods for evaluating the performance of a retrieval system, we measured the performance in terms of the proportion of relevant services in the retrieved list and their positions relative to non relevant ones (Precision-at-n). Precision-at-n measure allows computing precision at different cut-off points (Korfhage, 1997). For example, if the top 10 documents are all relevant to the query and the next 10 are all non-relevant, we have 100% precision at a cutoff of 10 documents but a 50% precision at a cut-off of 20 documents. Formally:

$$\text{Precision at } n = \text{RetReln} / n \quad (1)$$

where RetReln is the total number of relevant services retrieved in the top n. We evaluated Precision-at-n for each experiment with three values of n: n = 2, n = 4 and n = 5.

EXPERIMENTAL RESULTS

Once the terms are extracted using a PHP script, the terms along with the Uniform Resource Locater (URL) are stored in table. NSS of these terms is calculated and after determining the Eigen value of the correlation matrix and applying PCA, the service is permanently allocated to one or more category(s). The biplot achieved by applying principal component analysis on Table 1 is shown in Figure 1. The biplot clearly indicates that service with data considered in Table 1 will go to '*Weather'* category. Similarly all services can be allocated to one or more candidate categories permanently using the PCA.

The parameter used for measuring the effective our approach was that we compared the performance of the proposed approach, that is, service retrieved with categorization to the service retrieved without categorization. The comparisons were made on the top two, four and six services retrieved. We conducted many experiments and examined the URLs of the retrieved services on different categories and results achieved are shown in Figure 2. In our experiment, average precision-at-2 was around 95 % for the proposed approach.

Frontend

To minimizing discoverers' effort, the proposed solution provides a drop down list of available categories and terms associated with each category along with a text box for adding the query. A "Google-like" query interface relieves user from learning any other query language (Figure 3). Initially, the user or service discoverer is directed to choose a category and once a category is selected, he can either select the terms from dropdown list containing terms relevant to that category or he can add any query in the text box provided (Figure 4). Once the user selects a category and provides the relevant query terms the list of URLs within that category are retrieve (Figure 5). Thus the entire problem of finding relevant services is reduced to looking for similar services within a category.

DISCUSSION

As discussed previously, classification of Web service

Table 1. The normalized similarity score of all categories with all terms extracted from a WSDL file.

Category / Terms	Zip code	Weather	Country	Stock market	Currency
City	0.639	0.822	0.983	0.473	0.244
Temperature	0.826	0.635	0.948	0.241	0.11
Pressure	0.664	0.98	0.862	0.668	0.226
Humidity	0.924	0.992	0.891	0.533	0.547
Precipitation	0.368	0.89	0.503	0.028	0.013
Visibility	0.15	0.983	0.476	0.059	0.032
Clouds	0.078	0.517	0.062	0.058	0.02
Wind speed	0.444	0.905	0.243	0.021	0.009
Wind direction	0.017	0.878	0.259	0.014	0.007
Longitude	0.861	0.988	0.934	0.217	0.54
Latitude	0.587	0.959	0.887	0.26	0.273
Elevation	0.472	0.594	0.702	0.062	0.024

Here coloums indicate the categories and rows indicate the terms extracted from a file.

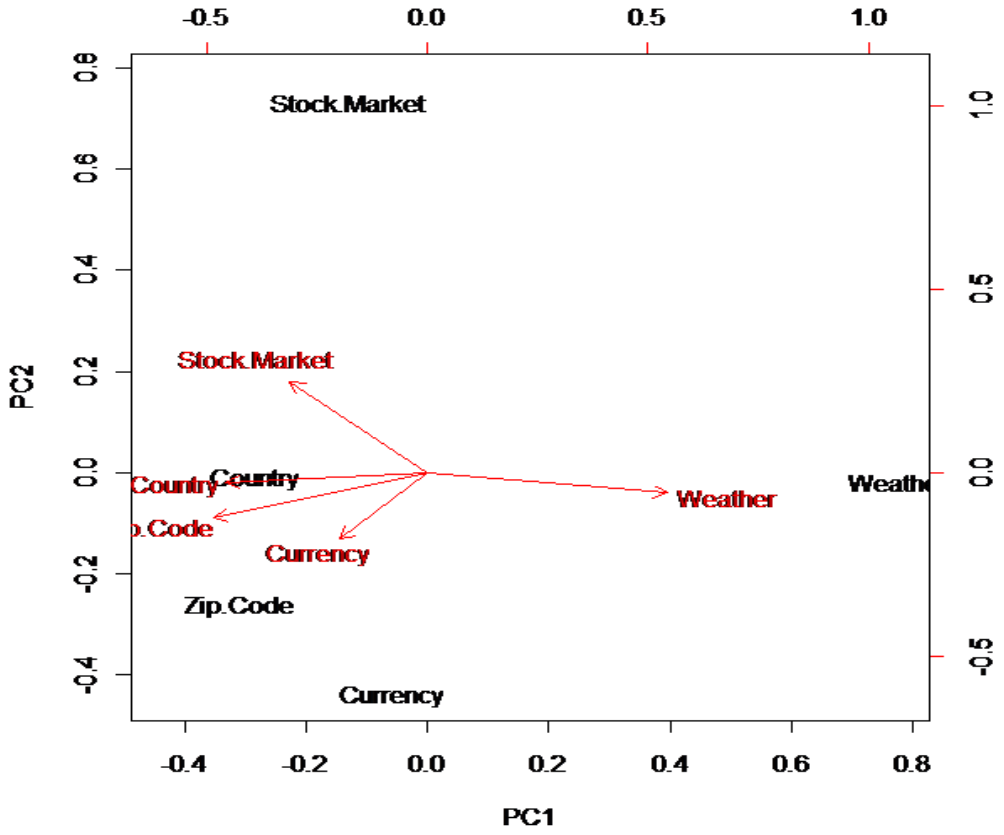

Figure 1. Biplot of principal component anlaysis (PCA) applied to values in Table 1.

into one or more candidate category(s) definitely reduces the discoverer's effort and increase the efficiency of the search. Since services are allocated to one or more category (s), the accuracy will definitely increase as the services lying within the category selected by the user will be retrieved. Overhead introduced in the proposed approach is that when publishing a new Web service addition of semantic metadata in the annotation tag is mandatory but the time required to annotate a service be comfortably ignored. The processing time required to

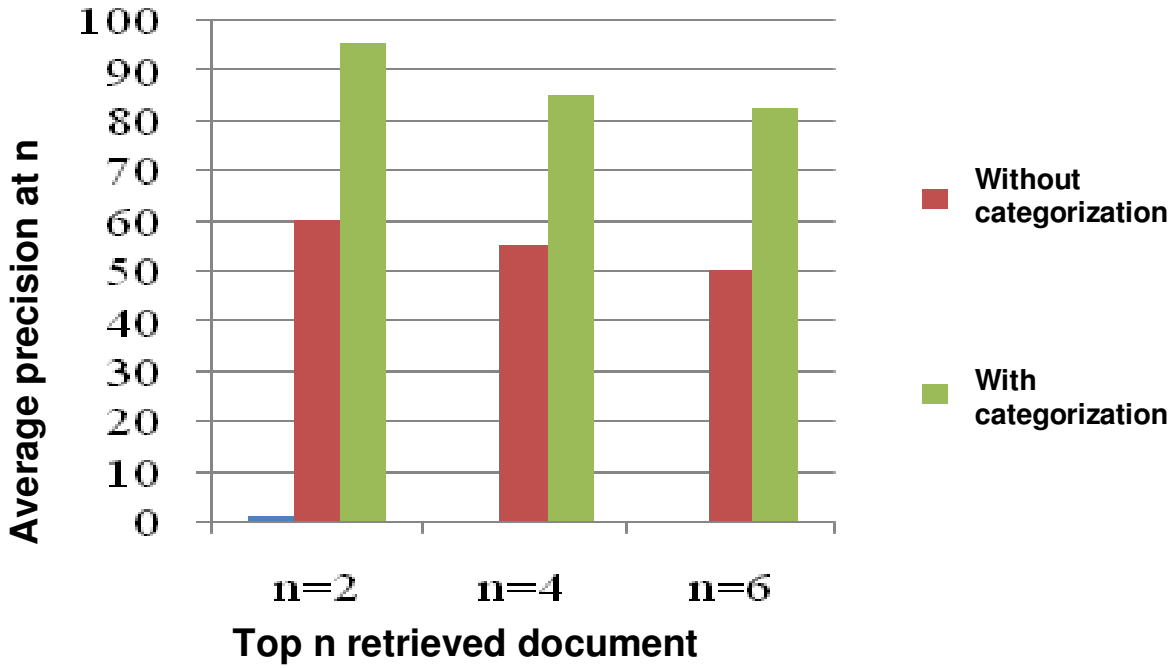

Figure 2. The average precision at n for different values of n are indicated by the bar chart which shows that precision rate is quite high when discoverer opts for finding Web services with categorization compared to using Web services without categorization.

Figure 3. Front end of the proposed framework where the service discoverer has the option of selecting a category from the set of categories presented in the form of a drop down list.

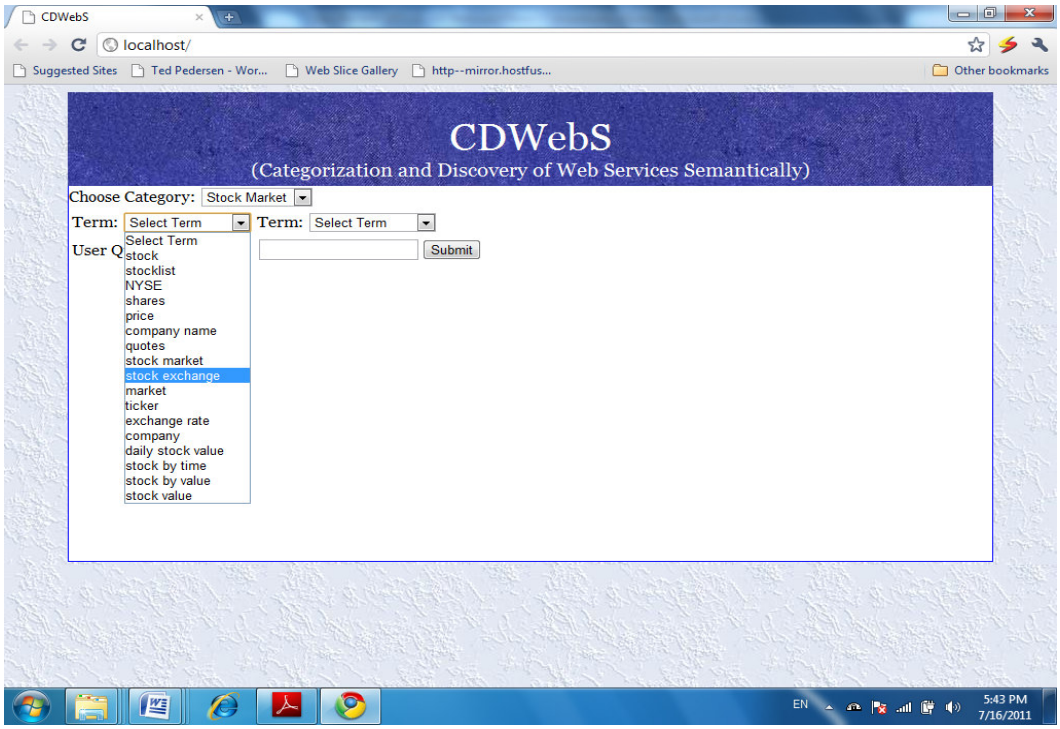

Figure 4. Front end with category selected as “Stock Market”, all terms related to stock market category displayed in drop down list.

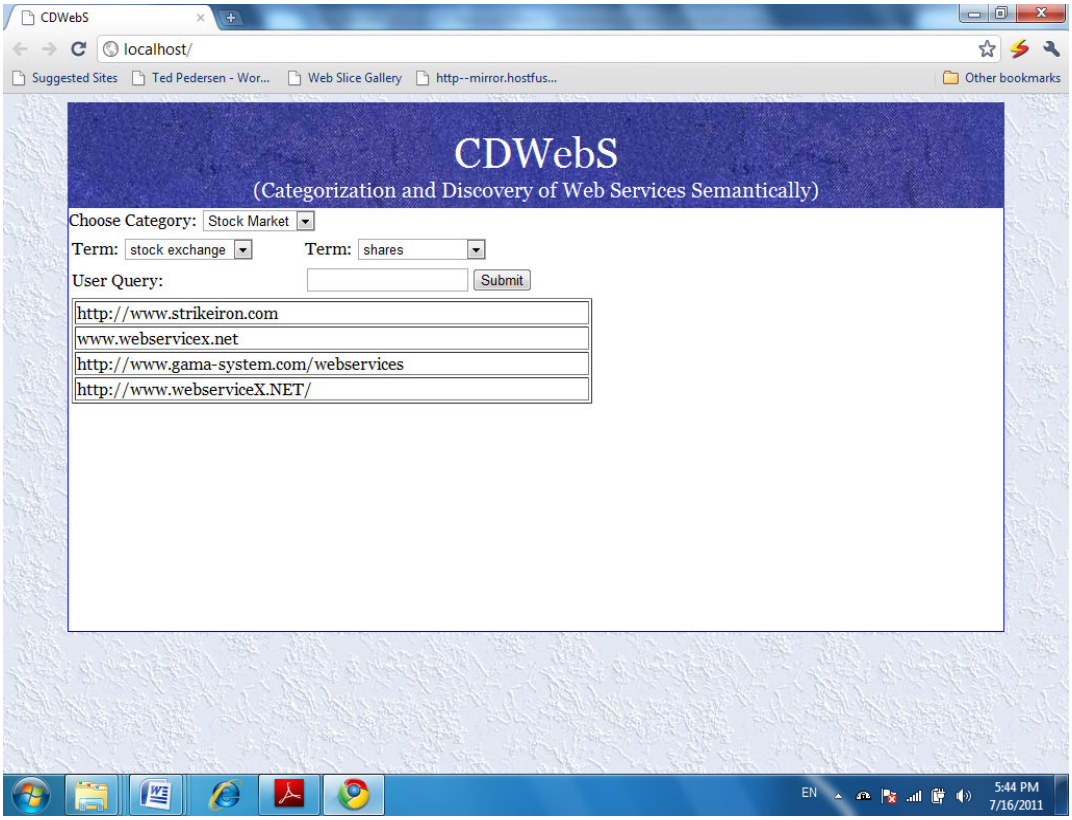

Figure 5. Front end with category selected as “Stock Market”, terms selected as ‘Stock Exchange’ and ‘shares’ and the list of URLs displayed.

categorize a service can be considered as another overhead but since only one time processing is required to categorize the services in the UDDI; this can also be ignored. As the number of categories will increase the size of term-category matrix will naturally increase and hence time required to add a service to a category will increase but it will not affect the service discoverer in any form instead it will increase the search scope of the discoverer.

Conclusions

Semantic Web service discovery mechanism is an upcoming challenge for the research community and many proposals have come up for efficient service discovery. Our framework provides an incremental approach to present WSDL standard and results indicate that classification of service is indeed an good option as it can always be assumed that the discoverer is clear about his service requirements and categorization will defiantly decreases his search effort as his search is now limited to a specific domain. Data set considered is for empirical evaluations is quite small and all these experiments have been performed on a single machine with local host and in future we will implement it with large number domains categories and multiple Web services.

REFERENCES

Akkiraju R, Farrell J, Miller J, Nagarajan M, Schmidt M, Sheth A, Verma K (2005). Web Service Semantics WSDL-S. W3C Member (http://www.w3.org/Submission/WSDL-S/)

Aleksovski Z, Klein M, Kate W, Harmelen F (2006a). Matching unstructured vocabularies using a background ontology. In Proceed. of Knowl. Eng. and Knowl. Manage. (EKAW), pp. 182–197.

Aleksovski Z, Klein M, Kate W, Harmelen F (2006b). Exploiting the structure of background knowledge used in ontology matching. In Ontol. Match. Workshop at Int. Semantic Web Confere.(ISWC).

Bai L, Liu M (2011). Fuzzy sets and similarity relations for semantic web service matching. Comp. Math. with Appl., 61 : 2281–2286.

Batra S, Bawa S (2009). Semantic categorization of web services. Int. J. Recent Trends in Eng.,2(2):19-23.

Batra S, Bawa S (2010a). Web services categorization using normalized similarity score. Int. J. Comp. Theory Eng.,2(1):139-142.

Batra S, Bawa S (2010b). A framework for semantic discovery of web services. In 5th International Workshop on Ubiquitous and Collaborative Computing, Scotland (UK), Sep. 2010. Published by British Comput. Society (eWIC).

Bellwood T, Clement L, Uddiversion D (2002). UDDI version 3.0 http://uddiorg/pubs/uddi-v3.00- published-20020719.htm.

Bollegala D, Matsuo, Mitsuru Y, Ishizuka (2007). Measuring semantic similarity between words using web search engines. In Int. World Wide Web Confe. Committee (IW3C2), May 8-12, Banff, Alberta, Canada.

Bruijn J, Bussler C, Domingue J, Fensel D, Hepp M, Keller U, Kifer M, Konig-Ries B, Kopecky J, Lara R, Lausen H, Oren E, Polleres A, Roman D, Scicluna J, Stollberg M (2005). Web Service Model. Ontol. (WSMO). W3C Member Submission 3. June 2005. (http://www.w3.org/Submission/WSMO/)

Chen H, Lin M, Wei Y (2006). Novel association measures using web search with double checking. In Proceed. of the COLING/ACL, pp. 1009-1016.

Cilibrasi R, Vitanyi P (2007). The Google similarity distance. IEEE Trans. Know. Data Eng., 19(3): 370-383.

Crasso M, Zunino A, Campo M (2008). Easy web service discovery: A query-by-example approach. Sci. Comp. Prog., 71:144–164.

Crasso M, Rodriguez JM, Zunino A, Campo M (2010). Improving web service descriptions for effective service discovery. Sci. Comp. Prog., 75: 1001-1021.

Fenza G, Loia V, Senatore S (2008). A hybrid approach to semantic web services matchmaking. Int. J. Approx. Reason., 48:808–828.

Friedman S, Weisberg HF (1981). Interpreting the first eigenvalue of a correlation matrix. Educ. Psychol. Meas., 41, 11–21.

Giunchiglia F, Shvaiko P, Yatskevich M (2004). S-match: an algorithm and an implementation of semantic matching. In Proceed. of the European Semantic Web Symposium (ESWC), pp. 61–75.

Gligorov R, Aleksovski Z, Kate W, Harmelen F (2007). Using Google Distance to weight approximate ontology matches. WWW 2007, Banff, Alberta, Canada.

Gómez-Berbís JM, Palacios RC, Cuadrado JL, Carrasco IJ, Crespo AG (2011). SEAN: Multi-ontology semantic annotation for highly accurate closed domains. Int. J. Ph. Sci., 6(6):1440-1451.

Hess A, Kushmerick N (2003). Learning to attach semantic metadata to web services. In 2nd Int.Semantic Web Conference (ISWC), Florida, USA.

Hotelling H (1933). Analysis of a complex of statistical variables into principal components. J. Educ. Psychol., 24, 417–441, 498–520.

Jackson JE (1991). A User's Guide to Principal Components. Wiley, New York.

Jolliffe IT (2002). Principal Component Analysis. Second Edition, Springer.

Kokash N (2006). A comparison of web service interface similarity measures. In 3rd European AI research. Symposium, IOS Press, Riva del Garda, Italy, pp. 220-231.

Korfhage RR (1997). Information Storage and Retrieval. John Wiley & Sons.

Laura RT, Sergio PI, Bergamaschi S, Mena E (2011). Using semantic techniques to access web data. Info. Sys., 36 :117–133.

Martin D, Burstein M, Hobbs JJ (2004). OWL-S: Semantic Markup for Web Services, W3C Member Submission (Available from http://www.w3.org/ Submission/OWL-S/).

Matsuo Y, Sakaki T, Uchiyama K, Ishizuka M (2006). Graph-based word clustering using web search engine. In Proceedings of EMNLP.

Pearson K (1901). On lines and planes of closest fit to systems of points in space. Phil. Mag., 6(2): 559–572.

Patil A, Sheth A, Verma K, Sivashanmugam K, Oundhakar S, Miller J (2005). METEOR-S WSDI: A scalable infrastructure Of registries for semantic publication and discovery of web services. J. Info. Tech. and Mgt., Special Issue on Universal Global Integrat.,6(1): 17-39.

Sahami M, Dumais S, Heckerman D, Horvitz E (1998). A Bayesian approach to filtering junk e-mail. In Learning for Text Categorization - Papers from the AAAI Workshop, pp. 55-62. Available at: ftp://ftp.research.microsoft.com/pub/ejh/junkfilter.pdf.

Salahli MA (2009). An approach for measuring semantic relatedness between words via related terms. Mathematical and Comp. App., Assoc. for Sc. Res., 14(1):55-63.

Stroulia E, Wang Y (2005). Structural and Semantic Matching for Accessing Web Service Similarity. Int. J Cooperat. Info. Sys., 14(4):407-437.

Subramaniam T, Jalab HA, Taqa AY (2010). Overview of textual anti-spam filtering techniques. Int. J. Phy. Sc., 5(12):1869-1882.

Turney P (2006). Similarity of semantic relations. Associat. for Computat. Linguist., 32(3).

UNSPSC http://www.unspsc.org/Search.asp

Vapnik VN, Druck H, Wu D (1999). Support Vector Machines for spam categorization. IEEE Trans. on Neural Networks, 10(5): 1048-1054.

Wu J, Wu ZH (2005). Similarity-based Web service matchmaking. In International conference on services computing. IEEE Computer Society, Orlando, FL, USA, pp. 287-294.

Zhang S, Bodenreider O (2005). Alignment of multiple ontologies of anatomy: Deriving indirect mappings from direct mappings to a reference. In AMIA Symposium Proceed., pp. 864–868.

Zhuang Z, Mitra P, Jaiswal A (2005). Corpus-based Web Services Matchmaking. In AAAI Conference.