Web Service Discovery via Semantic Association Ranking and Hyperclique Pattern Discovery

Aabhas V. Paliwal
Rutgers Univ., CIMIC
aabhas@cimic.rutgers.edu

Nabil R. Adam
Rutgers Univ., CIMIC
adam@cimic.rutgers.edu

Hui Xiong
Rutgers Univ., CIMIC,
hui@cimic.rutgers.edu

Christof Bornhövd
SAP Labs, LLC
christof.bornhoevd@sap.com

Abstract

Semantic Web technology is a promising first step for automated web service discovery. Most current approaches for web service discovery cater to semantic web services, i.e., web services that have associated semantic descriptions. It is unrealistic, however, to expect all new services to have associated semantic descriptions. Furthermore, the descriptions of the vast majority of already existing services do not have explicitly associated semantics. In this paper we present a novel approach for web service discovery that combines semantic and statistical association metrics. Semantic metrics are based on the semantic aspects of relevant ontology. Statistical association metrics are based on the association aspects of web services instances (their inputs and outputs). Specifically, our approach exploits semantic relationship ranking for establishing semantic relevance, and a hyperclique pattern discovery method for grouping web service parameters into meaningful associations. These associations combined by the semantic relevance are then leveraged to discover and rank web services.

1. Introduction

A large number of web services are being developed as an emerging standard to construct distributed applications in the web. Service requesters have access to a choice of descriptions to various services that provide similar service functionality. Most service descriptions that exist to date are more of a syntactic nature. However, the emerging service description paradigm has more associated semantics through the use of ontologies. Current service discovery approaches often adopt keyword-matching technologies to locate the published web services. This syntax-based matchmaking returns discovery results that may not match the service requester's original intent. The discovery process is constrained by its dependence on human intervention for choosing the appropriate service based on its semantic description. Automation of dynamic web service discovery is made viable by expression of domain semantics or domain specific knowledge [8].

Semantic Web technology is a potential solution for automated service discovery [9]. Consequently, semantic web services, an amalgamation of the semantic web and web services technology, are emerging as a promising methodology for the effective automation of service discovery, composition, and monitoring. The Semantic Web aims at systematizing and merging web service descriptions by using machine-understandable concepts explicitly defined through ontologies that are used to provide metadata for the effective manipulation of available information including discovering information sources and reasoning about their capabilities. A majority of the current approaches for web service discovery cater to semantic web services that have semantic tagged descriptions through various approaches [10]. However, there are two concerns: 1) it is impractical for us to expect all new services to have semantic tagged descriptions; and 2) descriptions of the vast majority of already existing web services do not have associated semantics.

Therefore, in this paper, we exploit a combination of service descriptions and service input and output parameters for accurately discovering the relevant web service meeting the service requestors' functionality. In this way, our approach combines semantics with syntactic characteristic of a Web Services Description Language (WSDL) document.

Indeed, service descriptions, i.e., WSDL document files, contain only implicit information about the corresponding service domain. Also, the web service input and output parameters contain underlying functional knowledge that may be extracted for improving service discovery. This fundamental knowledge conveys the semantic relationships and meaningful association between the operation parameters. These semantic relationships provide the service domain context for service discovery. A service domain is described by an exclusive set of concepts represented by an ontology. For example, to retrieve the web service that provides weather information, we utilize an ontology that describes weather features. For example, the weather ontology may represent semantic relationships between the weather features {temperature, humidity, visibility, precipitation}; that is, precipitation increase exponentially humidity or precipitation correlates wind. The semantic relationships extend the discovery mechanism for finding web service descriptions as they provide the context of the service domain and also impact web service ranking.

Proceedings of the 2006 IEEE/WIC/ACM International Conference
on Web Intelligence (WI 2006 Main Conference Proceedings)(WI'06)
0-7695-2747-7/06 $20.00 © 2006 IEEE

In addition, the relationship between web service input and output parameters may be represented as statistical associations. These associations relay information about the operation parameters that are frequently associated with each other. Along this line, we apply a hyperclique pattern discovery approach [12] for grouping web service input and output parameters into meaningful associations. A hyperclique pattern is a type of association pattern containing items that are strongly associated with each other. That is, every pair of items within a hyperclique pattern is guaranteed to have the uncentered correlation coefficient above a certain level. In the case of web services the items are the input or output parameters and a transaction is the set of input and output parameters for individual web service. Hyperclique pattern discovery can be adapted to capture frequently occurring local operation parameters' structures in web services, and thus can be used as service parameters' features [6].

More specifically, this paper presents a novel approach for web service discovery that combines semantic and statistical metrics. Semantic metrics are based on the semantic aspects of relevant ontology. Statistical metrics are based on the association aspects of web services instances, i.e., their inputs and outputs. Our method exploits semantic relationship ranking for establishing semantic relevance, and a hyperclique pattern discovery approach that groups web service parameters into meaningful associations. These associations combined with the semantic relevance are then leveraged to discover and rank web services.

2. Proposed Approach

In this work we present a novel discovery methodology for web services that takes advantage of association pattern mining and ranking semantic relationships. Our methodology builds on association pattern mining as described in [11] which does not assume term independence. The basic idea of our approach is to represent parameters of a web service as a vector in which each entry records the terms of the operations' input and output. Thus each of these collections of terms forms a transaction. The set of web services is represented by a collection of a set of transactions. Next we mine this web service collection to find the frequent hyperclique patterns that satisfy a given support level and h-confidence level [12]. This is followed by a pruning of the hyperclique patterns on the basis of the ranking of semantic relationships among the terms. Then for each remaining pattern we retrieve the web services that have the pattern expressed as part of the service description. Given the training set of the LSI classifier based on features extracted from selected WSDL files, we finally project the description vectors and the request vector and utilize the cosine measure to determine similarities and to retrieve the corresponding relevant WSDL service descriptions [5].

Below is a discussion of the semantic relationship ranking followed by a discussion on Hyperclique patterns. Finally we present the steps of our approach in more detail.

2.1 Ranking Semantic Relationship

There exist complex relationships among domain entities. These relationships may also be expressed as semantic associations. The complex relationships are based on property sequences that link the two entities in the semantic association. In [4] semantic associations are expressed as paths spanning across multiple domains that are represented by ontologies. These paths connect at least two entities and may involve multiple intermediate entities and relations. The ranking of the semantic associations facilitates the selection of the most relevant relationships among the entities. The ranking of the semantic association is based on the relevance, specificity and the span of the relationship [2].

2.2 Association Pattern Generation

In this paper, we apply Hyperclique patterns [12] for web service discovery. They are based on the concepts of frequent itemsets [1]. Next, we first briefly review the concepts of frequent itemsets and then describe the basic concepts of Hyperclique patterns.

Table 1: Example Hyperclique Patterns

Hyperclique Patterns	Support	h-Confidence
{temperature, pressure}	9.52%	50%
{mapurl,distanceunits,time,routeoptions}	4.76%	66.67%
{countrycode,countryname,region,ispname,domainname}	4.76%	100%

Table 1. Shows some example hyperclique patterns identified from a real-world web services data set, which includes web service descriptions from various service categories, e.g., 'weather', 'financial', 'business', and 'location', e.g., the hyperclique pattern {mapurl, distanceunits, time, routeoptions} is from the "location" category.

A Hyperclique pattern [12] is a new type of association pattern that contains items that are *highly affiliated* with each other. Specifically, the presence of an item in one transaction strongly implies the presence of every other item that belongs to the same hyperclique pattern. The h-confidence measure captures the strength of this association and, for an itemset $P=\{i_1, i_2, ..., i_m\}$, is defined

Proceedings of the 2006 IEEE/WIC/ACM International Conference
on Web Intelligence (WI 2006 Main Conference Proceedings)(WI'06)
0-7695-2747-7/06 $20.00 © 2006 IEEE

as the minimum confidence of all association rules of the itemset with a left hand side of one item, i.e., $hconf(P) = min\{conf\{i_1 \rightarrow i_2, ..., i_m\}, conf\{ i_2 \rightarrow i_1, i_3, ..., i_m\}, ..., conf\{i_m \rightarrow i_1, ..., i_{m-1}\}\}\}$, where *conf* follows the classic definition of association rule confidence [1]. An itemset P is a Hyperclique pattern if $hconf(P) \geq h_c$, where h_c is the minimum h-confidence threshold.

2.3 Service Discovery

The first step of our approach is to build the service parameters association pattern index. The next step involves pruning the association pattern based on concepts extracted from domain ontology and a confidence threshold. This is followed by building training set of the LSI classifier from the relevant WSDL files, and finally project the description vectors and the request vector utilizing the cosine measure to determine similarities and to retrieve the corresponding relevant WSDL service descriptions. We describe our approach with the help of the following example: *Service Request (SR): Find the temperature and rainfall based on zip code.*

Following is an outline of the key steps of our approach (see Figure 1). A more detailed description of our approach is given in [11], (1)Pre-process service request and form the expanded request vector, (2)Pre-process the available service descriptions set and retrieve associated parameters forming the association pattern itemset, (3) Perform Hyperclique pattern discoveries on the association pattern itemset, (4) Rank the semantic associations between the terms, (5) Prune the association patterns collection and, (6) Perform SVD on the term X document matrix and project description vectors and the request vector utilizing the cosine measure to determine similarity.

2.3.1. Expanded Service Request. The SR is parsed and pre-processed. The preprocessing includes the removal of markups, translation of upper case characters into lower case, punctuation and white space removal served as a term delimiter. The SR pre-processing phase also involves stoplist removal and stemming to strip word endings. The outcome of this preprocessing result in a term vector yielding term frequency. For our example the SR is transformed to {temperature, rain fall, zip code}.

The expanded request is a union of the original terms and the ontology concepts along with their concept hierarchies. [11] lists the main steps involved in the generation of the expanded SR denoted as SR_e.

2.3.2. Service Parameters Retrieval. The WSDL file described in section 2 forms part of the initial WSDL set and its corresponding description and associated parameters are parsed as follows. The WSDL document processing includes the extraction of the associated operation parameters by extracting all terms under the *<element name>* tag. The next step in the WSDL processing involves removal of markups and index entries, removal of punctuation and using white space as term delimiters. The WSDL processing also includes stoplist removal and stemming to strip word endings.

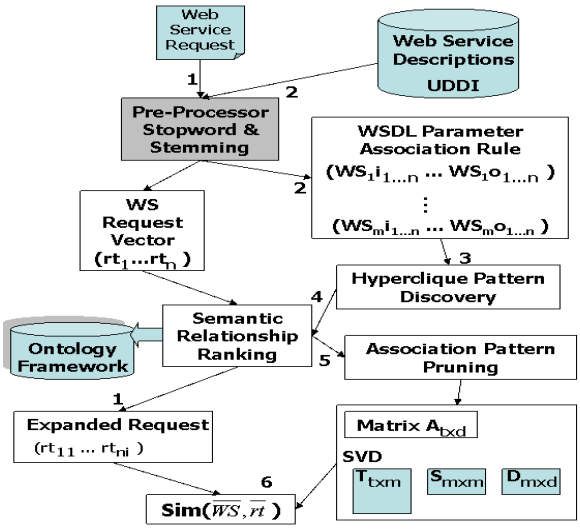

Figure. 1. Our Approach Combining Association Pattern Mining and Ranking Semantic Relationship

2.3.3. Hyperclique Pattern Discovery. In a nutshell, the process of searching hyperclique patterns can be viewed as the generation of a level-wise pattern tree. Every level of the tree contains patterns with the same number of nodes. If the level is increased by one, the pattern size (number of objects in the pattern) is also increased by one. Every pattern has a branch (sub-tree) which contains all the superset of this pattern. Our algorithm for finding hyperclique patterns is breadth-first. We first check all the patterns at the first level. If a pattern is not satisfied with the user-specified support and h-confidence thresholds, the whole branch corresponding to this pattern can be pruned without further checking. This is due to the anti-monotone property of support and h-confidence measures. Consider the h-confidence measure, the anti-monotone property guarantees that the h-confidence value of a pattern is greater than or equal to that of any superset of this pattern. Following this approach, the pattern tree is growing level-by-level until all the patterns have been generated. This algorithm is very efficient for handling large-scale datasets [12].

2.3.4. Ranking Semantic Associations. The complex relationships are based on property sequences that link the two entities in the semantic association.

Two entities e_1 and e_i are semantically associated with each other if there exists one or more relationship REL_j where $1 \leq i < n$ and $1 \leq j < n$. Next for each of these

entities we find the relevance, specificity and the user specified span. The user assigns weights for each of the parameters to refine the request. This also makes the ranking process more flexible. The associated semantic rank is utilized to sort the association pattern collection.

2.3.5. Association Pattern Collection Pruning. A large number of association patterns are generated in the association pattern mining phase. There is a need to disregard patterns containing irrelevant information that will influence the relevant service discovery process. The pruning of the association pattern collection is based on [3]: 1) eliminate the association patterns that have a low semantic relationship ranking between its terms; 2) retain the generic patterns with high confidence.

2.3.6. Service Request Projection. As shown in Figure 1, step 6 involves the selection of web service descriptions (WSDLs) based on the pruned association pattern collection. These documents are then parsed and processed to form the term-document matrix. Consequently, the term-document items are transformed using an "ltc" weighting [8].

The "ltc" matrix is used as input to the SVD algorithm. The SVD program calculates the best reduced dimension approximation for the transformed term-document matrix. This reduced dimensional representation is used for determining the appropriate web service. The cosine similarity between the term-term, request-description is used as a measure of similarity for further analysis of this representation [5].

This step involves projecting the description vectors and the request vector and utilizing the cosine measure to determine similarity. Following this, the corresponding web services are ranked as most appropriate based on a higher similarity measure.

3. Conclusion & Future Work

In this paper we presented a novel approach for web service discovery that combines semantic and statistical association metrics. Semantic metrics are based on the semantic aspects of ontology. Statistical association metrics are based on the association aspects of web services instances, specifically their inputs and outputs. Our methodology combines semantic relationship ranking, for establishing semantic relevance, and a hyperclique pattern discovery approach that groups web service parameters into meaningful associations. These associations combined by the semantic relevance are then leveraged to discover and rank web services.

Our research investigated two methods for web service discovery from a collection of web services. The association pattern mining augmented with ranking of semantic associations proved to be effective in the retrieval of more relevant web services.

As part of our future work we intend to extend our ontology framework by including more generic ontologies to develop a set of upper level ontologies. We also intend to investigate additional mapping tools to better express a service request to search for relevant concepts. Our future work also includes exploring better algorithms to determine the span and depth of the concepts to be selected for expanding the request. We plan to explore identifying efficient selection of dimensions to optimize the outcome of LSI. Finally, as part of the service discovery process we will explore associating semantic weights to the retrieved set of web services for more effective ranking of the results.

References

1. Agrawal, R., Imielinski, T., & Swami, A. (1993). "Mining association rules between sets of items in large databases". Proceedings of the ACM SIGMOD International Conference on Management of Data, 207-216.
2. Aleman-Meza, B., Halaschek-Wiener, C., Arpinar, I. B., Ramakrishnan, C., Sheth, A. P., "Ranking Complex Relationships on the Semantic Web," IEEE Internet Computing, vol. 09, no. 3, pp. 37-44, May/Jun, 2005.
3. Antonie, M.-L. and Zaane, O. R., "Text Document Categorization by Term Association" , IEEE ICDM'2002.
4. Anyanwu, K., Maduko, A., and Sheth, A. 2005. "SemRank: ranking complex relationship search results on the semantic web". In Proceedings of the 14th international Conference on World Wide Web (Chiba, Japan, May 10 - 14, 2005). WWW '05. ACM Press, New York, NY, 117-127.
5. Deerwester, S., Dumais, S. T., Landauer, T. K., Furnas, G. W. and Harshman, R. A., "Indexing by latent semantic analysis." Journal of the Society for Information Science, 41(6), 391-407
6. Dong, X., Havey, A., Madhavan, J., Nemes, E., and Zhang. J., "Similarity search for web services". In Proceedings of the 30th VLDB Conference, Toronto, Canada, 2004.
7. Dumais, S. T. "LSI meets TREC: A status report." In: D. Harman (Ed.), The First Text REtrieval Conference (TREC1), National Institute of Standards and Technology Special Publication 500-207 , pp. 137-152
8. Garofalakis, J., Panagis, Y., Sakkopoulos, E., Tsakalidis, A., "Web Service Discovery Mechanisms: Looking for a Needle in a Haystack?", International Workshop on Web Engineering, 2004
9. McIlraith, S., Son, T., and Zeng, H. "Semantic web services". IEEE Intelligent Systems 2001
10. McIlraith, S., Martin, D. "Bringing semantics to web services". IEEE Intelligent Systems, 2003.
11. Paliwal, A.V., Adam, N., and Bornhövd, C., RUTGERS University - CIMIC Technical Report, "Web Service Discovery and Composition: An Overall Approach ", 2006.
12. Xiong, H., Tan, P., & Kumar, V. "Mining Strong Affinity Association Patterns in Data Sets with Skewed Support Distribution", IEEE International Conference on Data Mining (ICDM), 387-394.