International Journal of Research in Computer Science
eISSN 2249-8265 Volume 2 Issue 3 (2012) pp. 51-56

www.ijorcs.org

ONTOLOGY MAPPING FOR DYNAMIC MULTIAGENT ENVIRONMENT

M.Jenifer[1], P.S.Balamurugan[2], T. Prince[3]

[1]*Research Scholar, Department of Computer Science, Karpagam University, Coimbatore, India*
Email: jenimetil@gmail.com

[2]*Head, Department of Computer Science, Karpagam University, Coimbatore, India*
Email: balabeme@gmail.com

[3]*Lecturer, Dept. of Computer Science, St. Joseph University in Tanzania, Tanzania, East Africa*
Email: prince.thomaskl@gmail.com

Abstract: *Ontologies are essential for the realization of the Semantic Web, which in turn relies on the ability of systems to identify and exploit relationships that exist between and within ontologies. As ontologies can be used to represent different domains, there is a high need for efficient ontology matching techniques that can allow information to be easily shared between different heterogeneous systems. There are various systems were proposed recently for ontology mapping. Ontology mapping is a prerequisite for achieving heterogeneous data integration on the Semantic Web. The vision of the Semantic Web implies that a large number of ontologies present on the web need to be aligned before one can make use of them. At the same time, these ontologies can be used as domain-specific background knowledge by the ontology mapping systems to increase the mapping precision. However, these ontologies can differ in representation, quality, and size that pose different challenges to ontology mapping. In this paper, we analyzed the various challenges of recently introduced Multi-Agent Ontology Mapping Framework, DSSim and we have integrated an efficient feature called QoS-Web Services Composition with DSSsim. ie we have improved this framework with QoS based Service Compositions Mechanism. From our experimental results, it is established that this developed QoS based Web Services Compositions Mechanism for Multiagent Ontology Mapping Framework minimizing uncertain reasoning and improves matching time, which are encouraging results of our proposed work.*

Keywords: *Uncertain reasoning, Multiagent Systems, QoS, Ontology Mapping, Semantic, Web Service Composition, Web Services.*

I. INTRODUCTION

The continuously increasing semantic metadata on the Web will make it possible to develop mature Semantic Web applications [1] which have the potential to attract commercial players to contribute to this Semantic Web vision. This is an important one because this will assure that more and more people will start using ontologies, which is a precondition for commercial viability of Semantic Web applications. However, the community expectations are also high when one thinks about the potential use of these applications. The vision of the Semantic Web promises a type of "machine intelligence," which can support a variety of user tasks like improved search or Question Answering (QA). For such applications, researchers have developed a wide variety of building blocks that needs to be utilized to achieve wider public acceptance. This is particularly true for ontology mapping [2], which makes it possible to interpret and align heterogeneous and distributed ontologies on the Semantic Web. However, to simulate "machine intelligence" for ontology mapping, different challenges have to be tackled. Consider, for example, the difficulty of evaluating ontologies with a large number of concepts. Owing to the size of the vocabulary, a number of domain experts are necessary to evaluate similar concepts in different ontologies. Once each expert has assessed sampled mappings, individual assessments are discussed, and a final assessment is produced, which reflects a collective judgment. This form of collective intelligence can emerge from the collaboration and competition of many individuals and is considered to be better at solving problems than those from experts who independently make assessments. This is because these experts combine their knowledge and experience to create a solution rather than relying on a single person perspective. Miklos Nagy and Maria Vargas Vera proposed and discussed what problems need to be addressed before one can achieve such machine intelligence for ontology mapping and they introduced a Multiagent Ontology Mapping Framework (DSSim) [3] that addresses these problems. From their results, the developed multiagent ontology mapping framework operates effectively in the Semantic Web environment. However, the mapping performance might be improved by modifying the architecture of DSSim with Web Service Composition, which will maintain the semantic similarity. This will improve the performance mapping of DSSim and reducing uncertain reasoning.

II. MULTIAGENT ONTOLOGY MAPPING FRAMEWORK

For ontology mapping in the context of QA over heterogeneous sources, Miklos Nagy and Maria Vargas Vera proposed a multiagent architecture [4]. The main reason for this is, when a particular domain becomes larger and more complex, open, and distributed, a set of cooperating agents is necessary to effectively address the ontology mapping task. In real scenarios, ontology mapping can be carried out on domains with large number of classes and properties. Without the multiagent architecture, the response time of the system can increase exponentially when the number of concepts to map increases.

The main objective of the DSSim architecture is to be able to use it in different domains for creating ontology mappings. These domains include QA, Web Services, or any application that needs to map database metadata, e.g., Extract, Transform, And Load (ETL) tools for Data Warehouses. Therefore, DSSim is not designed to have its own user interface but to integrate with other systems through well-defined interfaces. In the prototype implementation, Miklos Nagy and Maria Vargas Vera have used the Automated Question Answering System (AQUA) [5], which is the user interface that creates first-order logic (FOL) statements based on natural language queries posed by the user. As a consequence, the inputs and outputs for the DSSim system are valid FOL formulas. An overview of the DSSim system is shown in *Figure 1*.

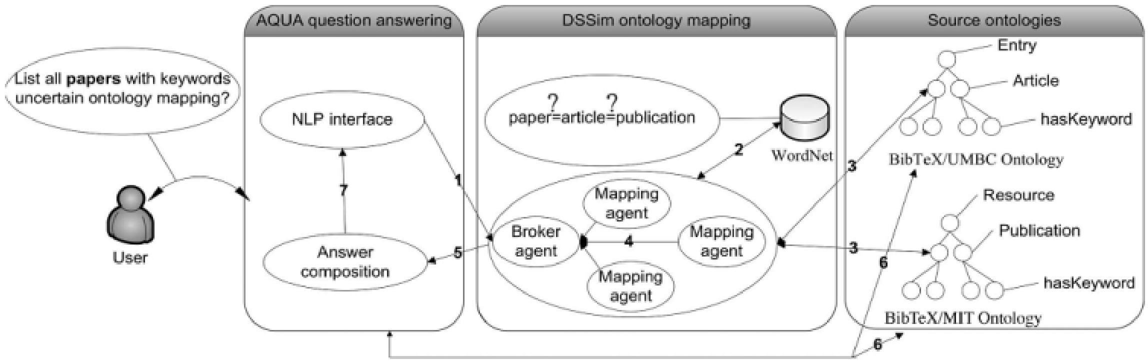

Figure 1: Overview of the DSSim Mapping System

The AQUA system and the answer composition component are described the context of overall framework. The user poses a Natural Language Query to the AQUA system, which converts it into FOL terms. The main components and its functions of the system are as follows.

- Broker agent receives FOL term, decomposes it and distributes the subqueries to the mapping agents.
- Mapping agents retrieve subquery class and property hypernyms from WordNet.
- Mapping agents retrieve ontology fragments from the external ontologies, which are candidate mappings to the received sub-queries. Mapping agents use WordNet as background knowledge to enhance beliefs on the possible meaning of the concepts or properties in the particular context
- Mapping agents build up coherent beliefs by combining all possible beliefs over the similarities of the sub-queries and ontology fragments. Mapping agents utilize both syntactic and semantic similarity algorithms and build beliefs over the correctness of the mapping.
- Broker agent passes the possible mappings into the answer composition component that corresponds to the particular sub-query for which the belief function attains the highest value.
- Answer composition component retrieves the concrete instances from the external ontologies or data sources, which are included in the answer
- Aswer composition component creates an answer to the user's question.

The main novelty of the DSSim's solution is that the authors Miklos Nagy and Maria Vargas Vera approach the ontology mapping problem based on the principles of collective intelligence, where each mapping agent has its own individual belief over the solution. However, before the final mapping is proposed, the broker agent creates the result based on a consensus between the different mapping agents. This process reflects well how humans reach consensus over a difficult issue.

III. SEMANTIC SIMILARITY

For semantic similarity between concepts, relations, and properties, Miklos Nagy and Maria Vargas Vera have used graph-based techniques. DSSim takes the extended query and the ontology input as labeled graphs. The semantic matching is viewed as graph like structures containing terms and their interrelationships.

The similarity comparison between a pair of nodes from two ontologies is based on the analysis of their positions within the graphs. The basic assumption is that, if two nodes from two ontologies are similar, their neighbors might also be somehow similar.

Miklos Nagy and Maria Vargas have considered semantic similarity between nodes of the graphs based on similarity of leaf nodes, which represent properties. That is, two non-leaf schema elements are semantically similar if their leaf sets are highly similar, even if their immediate children are not. The main reason why semantic heterogeneity occurs in the different ontology structures is because different institutions develop their data sets individually, and as a result, these metadata sets contain many overlapping concepts. Assessing the aforementioned similarities in DSSim, Miklos Nagy and Maria Vargas Vera have adapted and extended the SimilarityBase and SimilarityTop algorithms [6] used in the current AQUA system for multiple ontologies. Miklos Nagy and Maria Vargas Vera aim was that the similarity algorithms (experts in terms of evidence theory) could mimic the way a human designer would describe a domain, based on a well-established dictionary. What also needs to be considered is, when two graph structures are obtained from both the user and query fragments, the representation of the subset of the source ontology can be a generalization or specialization of specific concepts present in the graph. This can be derived from the external source and needs to be handled correctly. DSSim adapts and extends the SimilarityBase and SimilarityTop algorithms, which have been proved effective in the current AQUA system for multiple ontologies.

A. DSSim Procedure

The procedure of the DSSim Mapping System is demonstrated as follows.

1. *List all papers with keywords uncertain and ontology mapping*
   a. *List all the papers and Keywords*
   b. *Decompose the given keywords interms of Keyword, Publication, and URLs*
   c. *Communicate to WordNet for Mapping and to find ontologies*
   d. *Iterate the Mapping procedure and find the evidence*
2. *Compare the Evidence with different publications to Agent 1, Agent 2 and etc.*
   a. *Finally send to Broker Agent as Answer*
3. *Send this Answer to requested User*

IV. EXISTING ISSUES OF DSSIM FOR LARGER DATASETS

There are thirteen participants took part in the OAEI 2008 campaign, but only one of them was present in all four tracks. Thus, it is not possible to determine which participant has the best performance over all the datasets provided.

Nevertheless, in this work, from our analysis report, the following four systems namely Lily, SAMBO, RiMOM, and DSSim have shown very good performance in the benchmark, anatomy, FAO, Directory, and VLCR datasets, which is shown in the Figure. 2.

System	Benchmark	Anatomy	FAO	Directory	mldirectory	Library	vlcr	conference
Anchor-Flood	√	√						
AROMA	√	√						
ASMOV	√	√	√	√				
CIDER	√			√				
DSSim	√	√		√	√	√	√	√
GeRoMe	√							
Lily	√	√	√	√	√	√		√
MapPSO	√		√	√	√			
RiMOM	√	√	√	√	√			
SAMBO	√	√	√					
SAMBOdtf	√	√	√					
SPIDER	√							
TaxoMap	√	√		√		√		

Figure 2: Test sets (OAEI 2008) with different Systems

From our literature survey, we have noted that DSSim is performing well for semantic similarity. But still its performance might be improved interms of matching time and uncertain reasoning, i.e. the performance of this DSSim system could be improved further with the an efficient QoS based Web Services Composition Mechanism (QoS-WSC).

This modified QoS-WSC based DSSim will support for the largest datasets, i.e. the matching time of the enhanced QoS-WSC based DSSim will be lesser as compared to the existing DSSim System.

V. QOS-WSC BASED DSSIM

In this section, we would like to discuss the various existing Web Services Compositions [7], [8], [9], [10] approaches namely Semantic Approach. From an initial set of available services, we can define Service Composition as follows.

Definition 1 (Web Service Composition) : Web Service Composition aims at selecting and interconnecting Web Services provided by different partners in order to achieve a particular goal. Automating Web Service Composition aims to overcome the problem where no single service can satisfy the goal specified by the service consumer.

A. Web Service Composition

A web service is a software system identified by a URL, whose public interfaces and bindings are defined and described using XML. Its definition can be discovered by other software systems. These systems may then interact with the web service in a manner prescribed by its definition, using XML-based messages conveyed by internet protocols. This definition has been published by the World Wide Web consortium W3C, in the Web Services Architecture document. The web service model consists of three entities, the Service Provider, The Service Registry and the Service Consumer. The *Figure 3* shows a graphical representation of the traditional Web Service Model [7], [8].

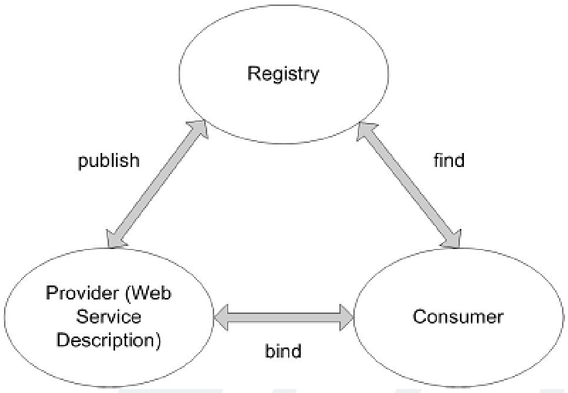

Figure 3: Web Service Model

The Service Provider creates or simply offers the Web Service. The Service Provider needs to describe the web service in a standard format, which in turn is XML and publish it in a central Service Registry.The Service Registry contains additional information about the Service Provider, such as address and contact of the providing company, and technical details about the service. The Service Consumer retrieves the information from the registry and uses the service description obtained to bind to and invoke the web service.

The appropriate methods are depicted in Figure 4. by the keywords ‘publish’, ‘bind’, and ‘find’. In order to achieve communication among applications running on different platforms and written in different programming languages, standards are needed for each of these operations.Web Services Architecture is loosely coupled, service oriented. The Web Service Description Language WSDL uses the XML format to describe the methods provided by a web service, including input and output parameters, data types and the transport protocol, which is typically HTTP, to be used. The Universal Description Discovery and Integration standard UDDI suggests means to publish details about a Service Provider, the services that are stored and the opportunity for service consumers to find service providers and web service details. Besides UDDI, other standards have been developed as well. The Simple Object Access Protocol SOAP is used for XML formatted information exchange among the entities involved in the web service model.

When composing web services, the business logic of the client is implemented by several services. This is analogous to workflow management, where the application logic is realised by composing autonomous applications. This allows the definition of increasingly complex applications by progressively aggregating components at higher levels of abstraction. A client invoking a composite service can itself be exposed as a web service.

Since it is a widely used approach to use conventional programming languages to link components to a composite web service and thereby bridge heterogeneous middleware platforms, it becomes necessary to develop a Service Composition Middleware to support composition in terms of abstractions and infrastructure as well.

B. Web Service Composition Approach in DSSim

The general and typical functionalities of Web Service Composition is disucssed in the previous section. In this section, we have followed the same procedure for decomposing key words / publications and we have done mapping process with newly created and maintained WSC Databases if the contents are exists in WSC. Otherwise, the ordinary DSSim procedure will be followed to match the keywords and mapping.

The proposed architecture is shown in the Figure 4. As showin the figure, once the user lists the papers and keywords, this proposed system will refer either the request is already approached this matching system or not. If the request is new, the Web Service Composition will be created and requested QoS will be assigned to WSC. Then the DSSim procedure will be followed and finally the WSC Database will be updated for future use.

If the request is laready performed one, the mapping procedure will be executed with the content of WSC, which will map the ontology with short period, which will improve the system performance in terms of content matching.

Further, this proposed mechanism will compare and monitor the uncertain reasoning with the given requested QoS and if the uncertain reasoning level is very low as compared with the demanded QoS, the uncertain reasoning will be removed from the database, which is improving the classification and prediction accuracy.

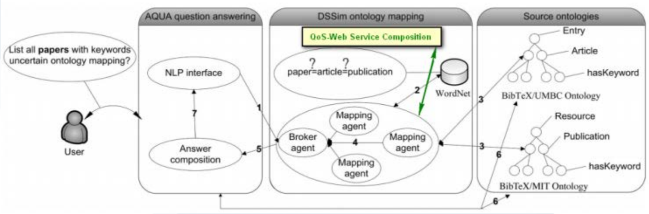

Figure 4: Overview of the QoS-WSC based DSSim Mapping System

C. The Procedure of the Proposed QoS-WSC based DSSim

The procedure of the Proposed QoS-WSC based DSSim Mapping System is demonstrated as follows.

Algorithm: Adaptive Web Service Selection

1. *If **new_query** then goto step2 else step3*
2. *Initialize **Web Service Composition Database** and required **QoS***
3. *List all papers with keywords uncertain and ontology mapping*
    a. *List all the papers and Keywords*
    b. *Decompose the given keywords interms of Keyword, Publication, and URLs*
    c. *Communicate to WordNet for Mapping and to find ontologies or **new ontologies***
    d. *Iterate the Mapping procedure and find the evidence from **WSC if exist or update WSC***
4. *Compare the Evidence with different publications*
    a. *If publications/keywords with **WSC** then send Answer*
    b. *Else compare with the new Agents Agent 1, Agent 2 and etc.*
    c. *Finally send to i. Broker Agent, ii. **update WSC***
5. ***Take this answer and compare with Requested QoS and send best matching answer to requested user***

VI. EXPERIMENTAL ANALYSIS

The evaluation uses recall, precision, and F-measure, which are useful measures that have a fixed range. They are also meaningful from the mapping point of view.

A. Precision

A measure of the usefulness of a hit list, where hit list is an ordered list of hits in decreasing order of relevance to the query and is calculated as

$$Precision = \frac{\left|\{Relevant\_Items\} \cap \{\text{Retrieved\_Items}\}\right|}{\left|\{Retrieved\_Items\}\right|}$$

B. Recall

A measure of the completeness of the hit list and shows how well the engine performs in finding relevant entities and is calculated as

$$Recall = \frac{\left|\{Relevant\_Items\} \cap \{\text{Retrieved\_Items}\}\right|}{\left|\{Relevant\_Items\}\right|}$$

C. F-Measure

The weighted harmonic means of precision and recall and is calculated as

$$F - Measure = \frac{2 * (Precesion * Recall)}{(Precision + Recall)}$$

D. Harmonic Mean (H-Mean)

Harmonic mean is used to calculate the average of a set of numbers. Here the number of elements will be averaged and divided by the sum of the reciprocals of the elements. The Harmonic mean is always the lowest mean.

$$H - Mean = \left[\frac{N}{\left(\frac{1}{a_1} + \frac{1}{a_2} + \frac{1}{a_3} + \ldots + \frac{1}{a_N} \right)} \right]$$

From the experimental results, both the DSSim and QoS-WSC based DSSim are getting almost the same level of content matching/mapping accuracy, which is shown in the Fig. 6. But however, this proposed QoS-WSC based DSSim removes more uncertain reasoning keys and the matching speed is more as compared with DSSim which is shown in the Figure 5.

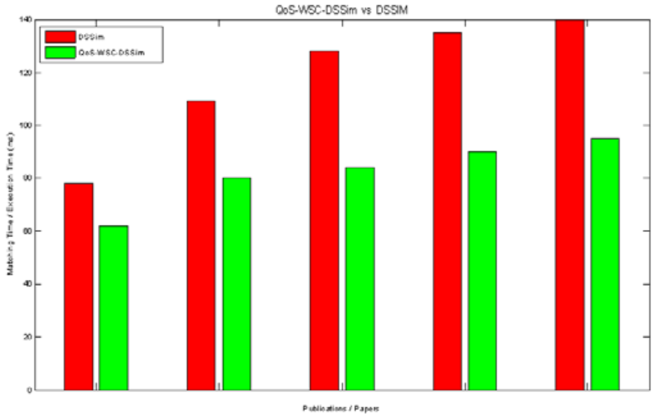

Figure 5: Matching Time of QoS-WSC-DSSim

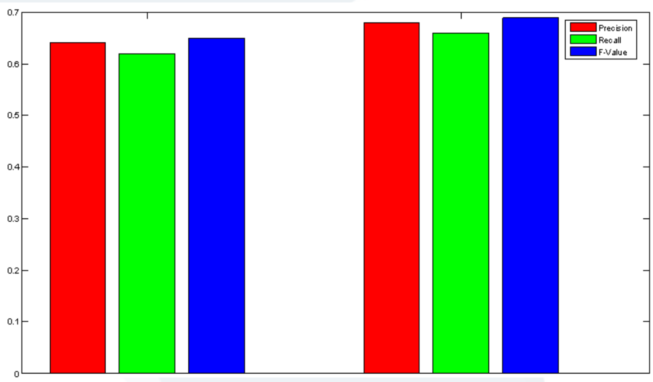

Figure 6: F-Values of DSSim and QoS-WSC-DSSim

As shown in the Figure 6, the matching time of QoS-WSC-DSSim almost constant for finding more key words/publications as compared with DSSim as in the case of content available in the Web Services Composition. ie for large volume of content matching, our proposed work is performing better as compared with the DSSim Ontology System .

VII. CONCLUSIONS

In this research work, the recently developed Multiagent Ontology Mapping Framework, DSSim has been studied thoroughly and implemented. From our results, we have observed that the performance of this work could be improved with QoS based Service Compositions Mechanism and thus we have developed QoS-Web Services Compositions based Mechanism for Multiagent Ontology Mapping Framework. From our experimental results, it is established that this developed QoS-Web Services Compositions based Mechanism for Multiagent Ontology Mapping Framework minimizing uncertain reasoning and maximizing content matching/mapping speed, which are encouraging our proposed work.

VIII. REFERENCES

[1] N. Shadbolt, W. Hall, and T. Berners-Lee, "The semantic web revisited," IEEE Intell. Syst., vol. 21, no. 3, pp. 96–101, Jan./Feb. 2006. doi:10.1109/MIS.2006.62

[2] J. Euzenat and P. Shvaiko, Ontology Matching. Heidelberg, Germany: Springer-Verlag, 2007.

[3] M. Nagy, M. Vargas-Vera, and E. Motta, "DSSim—Managing uncertainty on the semantic web," in Proc. 2nd Int. Workshop Ontology Matching, 2007, pp. 156–168.

[4] M. Nagy, M. Vargas-Vera, and E. Motta, "Multi-agent ontology map-ping framework in the aqua question answering system," inProc. 4th MICAI—Advances in Artificial Intelligence, 2005, pp. 70–79.

[5] M. Vargas-Vera and E. Motta, "Aqua—Ontology-based question answer-ing system," in Proc. 3rd MICAI, 2004, vol. 2972, pp. 468–477.

[6] M. Vargas-Vera and E. Motta, "An ontology-driven similarity algorithm," Knowl. Media Inst., Open Univ., Milton Keynes, U.K., Tech. Rep. KMI-TR-151, 2004.

[7] Zibin Zheng, Hao Ma, Michael R. Lyu and Irwin King, "QoS-Aware Web Service Recommendation Collaborative Filtering", IEEE Transactions On Service Computing, Vol. 4, No. 2, April-June 2011. doi:10.1109/TSC.2010.52

[8] L.-J. Zhang, J. Zhang, and H. Cai, Services Computing. Springer and Tsinghua Univ., 2007.

[9] L. Zeng, B. Benatallah, A.H. Ngu, M. Dumas, J. Kalagnanam, and H. Chang, "Qos-Aware Middleware for Web Services Composition,"IEEE Trans. Software Eng., vol. 30, no. 5, pp. 311-327, May2004.

[10] Z. Zheng, H. Ma, M.R. Lyu, and I. King, "Wsrec: A CollaborativeFiltering Based Web Service Recommender System," Proc. Seventh Int'l Conf. Web Services (ICWS '09), pp. 437-444, 2009. doi:10.1109/ICWS.2009.30

How to cite

M.Jenifer, P.S.Balamurugan, T. Prince, "Ontology Mapping for Dynamic Multiagent Environment". *International Journal of Research in Computer Science*, 2 (3): pp. 51-56, April 2012. doi:10.7815/ijorcs.23.2012.029