

Ontology Mapping with domain specific agents in the AQUA Question Answering system

Miklos Nagy, Maria Vargas-Vera and Enrico Motta

Knowledge Media Institute (KMi),
The Open University,
Walton Hall, Milton Keynes, MK7 6AA, United Kingdom
miklos.nagy@jrc.nl; {m.vargas-vera, e.motta}@open.ac.uk

Abstract. This paper describes a domain specific multi-agent ontology-mapping solution in the AQUA query answering system. In order to incorporate uncertainty inherent to the mapping process, the system uses the Dempster-Shafer model for dealing with incomplete and uncertain information produced during the mapping. A novel approach is presented how specialized agents with partial local knowledge of the particular domain achieve ontology mapping without creating global or reference ontology. Our approach is particularly fit for a query-answering scenario, where answer needs to be created in real time that satisfies the query posed by the user.

1 Introduction

An important aspect of ontology mapping is how the incomplete and uncertain results of the different similarity algorithms can be interpreted during the mapping process started to become a well-acknowledged research direction. As the latest research started moving towards a more automated mapping process it has been recognized that current approaches do not fully investigate the nature of the produced similarity information and mainly rely on a human domain expert to make a judgment about the correctness of the established mapping. However in the context of question answering like the AQUA [1,2] system the dynamic nature of the source information (e.g. web enabled databases) does not make it possible that a domain expert help is necessary every time the source changes to follow up the modifications in the existing mapping. Our novel approach to address this problem utilizes a multi agent framework where the different mapping agents possess local sub-domain specific knowledge about particular entities (e.g. material, specimen, etc.). From the end user perspective our system addresses the problem of data integration of scientific databases containing vast number of experimental Semantic Web enabled data in order to facilitate better knowledge sharing and reuse between the scientific communities. Although these databases are accessible, the seamless data exchange between different databases is still an unsolved problem in spite of the fact that different XML based languages were defined by the different scientific communities e.g. MatML(Materials Markup Language)[3] on the field of material science to facilitate a standardized XML based

data exchange. This solution solved a number of interoperability issues but makes the assumption that both parties agreed the syntax of the data exchange. This assumption fails when one would search for existing experimental data available on the WWW since neither the syntax nor the semantics of the requested data is known before the submission of the query. The problem is that different research institutions, companies use different standards and naming conventions in their logical data model for the same data, additionally these data model is not always even accessible on the WWW. Hence a vast number of experimental data are remaining inaccessible, or unanalyzed that probably hides the undiscovered correlations of science disciplines. The mapping agents use the Dempster-Shafer theory of evidence [4] to assess and combine the belief in the correctness of the different similarity algorithms. Our approach also does not assume the existence of global or reference ontology that is the superset of the different source ontologies and contains the existing mappings a priori. This approach makes it possible to perform query answering effectively with multiply source ontologies. In our first experimental system we consider query answering over Web enabled S&T (Scientific and Technical) or engineering databases those are described with their own domain specific ontologies.

The paper is organized as follows:

Section 2 presents the architecture of the mapping framework and describes how mapping agents on the different levels are carrying out the mapping. Section 3 introduces the similarity algorithm used by the framework to assess syntactic and semantic similarities between the posed query and the local ontologies. Section 4 describes how the problem of uncertain information created by the similarity mapping process is resolved and handled by the mapping framework. Section 5 presents a working example. Section 6 presents implantation details. Sections 7 discuss the related work and Section 8 gives conclusions as well as the future research directions.

2 Architectural overview of the mapping framework

The high-level system architecture figure 1 shows how the functional parts of the system are related with each other. In the mediator layer the agents are organized in different levels. Agents on the broker level responsible for decomposing the query into sub queries, based on the global descriptor. The decomposed query parts are sent into the mapping agents located in the mapping layer. Mapping agents obtain the relevant information from the sources through the source agents. When only one source corresponds to the query the scenario is pretty straightforward and there is no need for any mapping between the sources, the query can be answered from the source. In a real case scenario this possibility is not so likely and this is why the mapping between local ontologies is a justified scenario in our case.

The idea that has been investigated in our research is that mapping agents can build up mappings simultaneously, utilizing different similarity measures Based on their belief agents need to harmonize their beliefs based on trust that is formed during the mapping process.

This is a two-step process:

1. Mapping agent based on evidences that is available to them built up belief about the mapping.
2. Group of mapping agents need to harmonize their beliefs over the solution space.

The key components of the prototype are grouped by the different functional levels and from bottom to top as follows.

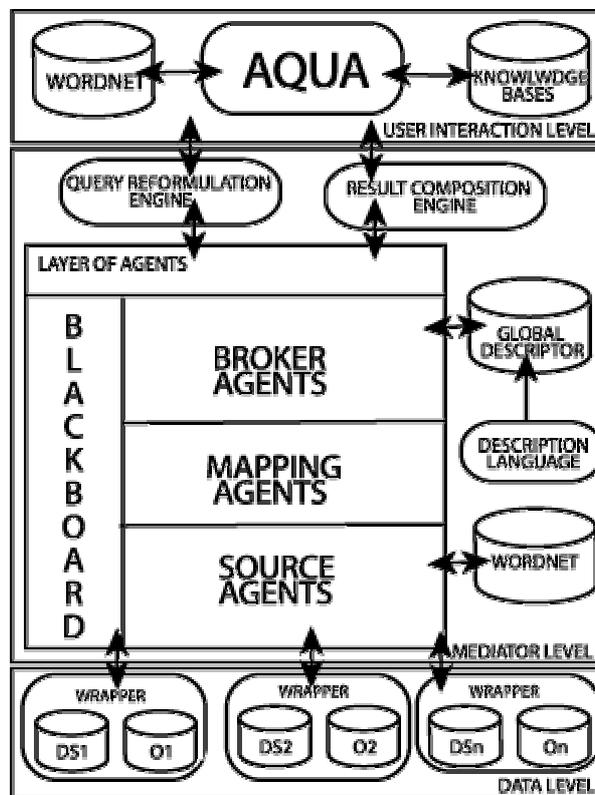


Figure 1. Architecture of the Multi-ontology mapping framework.

Data Level

On the data level the heterogeneous data sources are represented by their ontologies. The format of these sources varies from relational databases to simple files.

- Data source (DS): actual data represented in the database, file etc.
- Ontology (O) Semantic metadata that describes the particular data source.
- Wrapper creates a unified XML representation of the source that is queried by the particular resource agents.

Mediator level

Layer of agents: Typically three kind of agents: broker that receives a FOL (First Order Logic) query and decomposes it into sub queries based on the global descriptor, mapping that has knowledge of a particular domain specific area and cooperatively map up the source concept with the concepts contained by the query string, source that access a particular data source and it's ontology and passes it to the mapping agents on a request basis.

Global descriptor and description language: Key component of the system that describes what kind of information can be found in the different sources, and which agent is able to answer the query posed by the user based on the entities in the query. Practically FOL knowledge base that contains information about the agents and entities as well as the resources

Query reformulation and result composition engine: Query that is raised by the user needs to be reformulated and decomposed before entered into the system, which is the purpose of the query reformulation engine. Information flow stems from the mapping process needs to be composed into a single coherent answer, which is done by result composition engine. These subsystems are out of the scope of our research.

User Interaction level

The AQUA query answering system itself, which provides precise answers to specific questions raised by the user. It integrates Natural Language Processing (NLP), Logic, Ontologies and Information retrieval techniques

3. Similarity algorithms

The similarity-mapping algorithm takes one entity from O1 and tries to find similar entity in O2 . The similarity mapping process has different levels as follows:

- Concept-name similarity with Character-based Jaccard measure [5].

$$sim_{x_a x_b} = \frac{x_a * x_b}{\|x_a\| \|x_b\| - x_a * x_b} \quad (1)$$

where $x_a * x_b$ is the inner product of x_a, x_b and $\|x\|$ is the Euclidean norm for the vectors.

- Property set similarity with token based Jaccard distance: As first approach the property names are flattened into a bag of words per each node so similarity algorithms from the information retrieval field can be considered when two graph like structure are compared.
- Instance values similarity based on string similarity
- Concept-property similarity graph assessment

In order to increase the correctness of our similarity measures the obtained similarity coefficients need to be combined. Establishing this combination method is the primary objective that needs to be delivered with the with our outlined system. Further once the combined similarity has been calculated we need to develop a methodology to derive a belief mass function that is the fundamental property of Dempster-Shafer evidence theory.

In our prototype it is necessary to assess not only the syntactic but also the semantic similarity between concept, relations and the properties. The main reason why semantic heterogeneity occurs in the different ontology structures is the fact that different institutions developed their data sets individually, which contains mainly overlapping concepts. Assessing the above-mentioned similarities in our multi agent framework we adapted and extended the SimilarityBase and SimilarityTop algorithms [6,7] used in the current AQUA system for multiply ontologies. The goal of our approach is that the specialized agents simulate the way in which a human designer would describe its own domain based on a well-established dictionary. What also needs to be considered when the two graph structures obtained from both the user query fragment and the representation of the subset of the source ontology is that there can be a generalization or specialization of a specific concepts present in the graph which was obtained from the local source and this needs to be handled correctly. In our multi agent framework the extended and combined SimilarityBase and SimilarityTop algorithms can be described as follows:

1. Based on the WordNet reflexive lexical morphosemantic relation a directed graph is constructed from the FOL query fragment where there are bi-directional edges between the nodes representing the concepts and there are directed edges from the concepts to the property nodes. In this step the specialized agents try determine all possible alternatives for the meaning of the query fragment that it can be aware of. Figure 2 depicts the graph representation of the hasName(material, 10 CrMo 9 10) FOL query fragment.

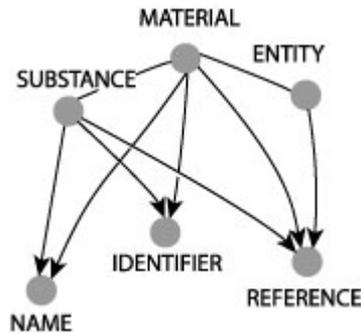


Figure 2. G0 query fragment graph

- Based on the before mentioned character and token based Jaccard distance similarity measure the specialized agent builds up a directed graphs from the local ontology structures that supposedly answers the query fragment. Figure 3 depicts two graphs obtained from two different sources.

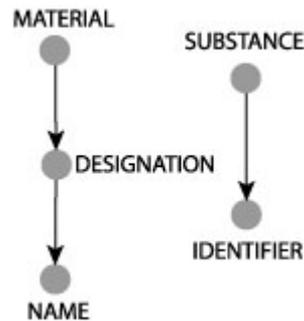


Figure 3. G1 and G2 graph representation of the local ontology fragment.

- Top-down sub-graph (isomorphism) similarity assessment[8] is applied on the graph G0 in order to find the subgraph G1 and G2 respectively. The aim is to find identical subgraphs to G1 and G2 in order to assess the similarity of the concepts and properties that can answer the query fragment. We call this method a top-down assessment because the search for the sub graphs starts from the concept nodes towards property nodes through the directed edges. Once we reached the property node the search stops. If along the path we walked through the graph we found a sub graph identical (isomorph) to G1 and G2 that agent can deduce that the query fragment can be answered from the sources that belong to the particular ontology and the concepts or proper-

ties identified in the different sources are similar to both each other and to the query fragment and a basic mass function can be calculated that express the extent of belief in the existence of the similarity mapping between them. In case G1 or G2 contains nodes that could not be found in the G0, because of the nature of the top down assessment the agent can deduce that the particular concept node is a specialization of the concept that was identified by the algorithm.

4. Uncertainty handling

In our framework we use the Dempster-Shafer theory of evidence, which provides a mechanism for modeling and reasoning with uncertain information in a numerical way especially when it is not possible to assign a belief to a single element of a set of values. The main advantage of the Dempster-Shafer (D-S) theory over the classical probabilistic theories that the evidence of different levels of abstraction can be represented in a way that clear discrimination can be made between uncertainty and ignorance. Further advantage is that the theory provides a method for combining the effect of different learned evidences to a new belief by the means of the Dempster's combination rule. Let's first describe the basic concepts of the Dempster-Shafer theory and how it corresponds to our system.

Frame of Discernment (Θ): finite set representing the space of hypothesizes. It contains all possible mutually exclusive context events of the same kind. In our system this corresponds to the possible properties, those of the base entities that describes the concepts of the domain e.g. Material Name, Test Control, Specimen Identifier etc.

Evidence: available certain fact and is usually a result of observation. Used during the reasoning process to choose the best hypothesis in Θ . In our system this can be a certain observation that e.g. in the case of material entity the production details has been observed or not.

Belief mass function (m): is a finite amount of support assigned to the subset of Θ . It represents a strength of some evidence and

$$\sum_{A \subseteq \Theta} m(A) = 1 \quad (2)$$

where $m(A)$ is our exact belief in a proposition represented by A . The similarity algorithms itself produce these assignment based on the before mentioned (Section 3) similarities e.g. between name and identifier property the assigned value is 0.7.

Once the belief mass functions have been assigned the following additional measures can be derived from the available information.

Belief: amount of justified support to A that is the lower probability function of Dempster, which accounts for all evidence E_k that supports the given proposition A .

$$belief_i(A) = \sum_{E_k \subseteq A} m_i(E_k) \quad (3)$$

Plausibility: amount of potential support on A that is the upper probability function of Dempster, which accounts for all the observations that do not rule out the given proposition.

$$plausibility_i(A) = 1 - \sum_{E_k \cap A = \emptyset} m_i(E_k) \quad (4)$$

Ignorance: the lack of information.

$$ignorance(A) = plausibility(A) - belief(A) \quad (5)$$

Once all the necessary variables have been assigned to a qualitative value we need to combine the belief mass functions that was created by the different agents for the particular query fragment.

Dempster's rule of combination:

Suppose we have two mass functions $m_i(E_k)$ and $m_j(E_{k'})$ and we want to combine them into a global $m_{ij}(A)$. Following Dempster's combination rule

$$m_{ij}(A) = m_i \oplus m_j = \sum_{E_k \cap E_{k'}} m_i(E_k) * m_j(E_{k'}) \quad (6)$$

However when $E_k \cap E_{k'} = \emptyset$ the mass $m_i(E_k) * m_j(E_{k'})$ would go to \emptyset , it is necessary to normalize the mass function with the lost mass so

$$m_{ij}(A) = \frac{\sum_{E_k \cap E_{k'}} m_i(E_k) * m_j(E_{k'})}{1 - \sum_{E_k \cap E_{k'} = \emptyset} m_i(E_k) * m_j(E_{k'})} \quad (7)$$

An important part of the system is how the similarity measures are applied in the concrete scenario and how the particular agent assesses the belief mass functions and belief functions. In our experimental system we consider basic probability assessment over the following entities:

1. Class: The most basic concepts in the domain that correspond to classes that are the root of the various taxonomies
2. Object properties: Relation between the instances of two classes
3. Data type properties: Relation between instances of classes and RDF literals and XML Schema data types therefore it describes that the particular class e.g. material has a data type property called name that which is a string.

5. Working example

In this chapter we describe the main functionality of our system with a rather simple example. This example serves as a first test bed of this complex problem. The global descriptor describes what kind of information can be found in the different local ontologies/sources.

$$GD = DO_1 \cup DO_n \cup DO_{n+1} \quad (8)$$

where GD is the global descriptor and DO_n is one of the particular domain ontology and

$$DO_i = \{R_{i1} \dots R_{ij}\} \quad (9)$$

where R_{ij} means the relation j in the ontology i .

As discussed the global descriptor can be best represented by FOL since the AQUA system also creates the query in FOL.

The global descriptor contains information about:

- Agents: MaterialAgent, SpecimenAgent, SourceAgent, TestConditionAgent and TestAgent as constant symbols
- Query and property information:
(canAnswer(x,Test),hasInformation(x,MaximumStress)) as predicate symbols.

In the following example the system uses two ontologies O_1 and O_2 and creates similarity mapping between the query fragment and the concepts in the ontologies respectively. Both O_1 and O_2 ontology describes mechanical material test information from different institutes. Extracts from the two ontologies can be found in section 6.1.

To illustrate the mapping process in our system the following steps are taken before the query can be answered:

1. At system startup the Global Descriptor contains only the pre defined concept-mapping agent pairs that describe which agent knows the particular concept:
 $\forall x$ Materialagent(x) and canAnswer(x,Material)
 $\forall x$ Specimenagent(x) and canAnswer(x,Specimen)
 $\forall x$ Testagent(x) and canAnswer(x,Test)
 $\forall x$ Sourceagent(x) and canAnswer(x,Source)
 $\forall x$ TestConditionagent(x) and canAnswer(x, TestCondition)
2. FOL Query passed to the broker agent:
Which test has been carried out on a bar shaped specimen?

$(\forall x, \exists y) (\text{Test}(x) \text{ and Specimen}(y) \text{ and form}(y, \text{bar}) \text{ and carriedOutOn}(x, y))$

3. Broker agent decomposes the query based on the information present in the Global Descriptor and forwards it to the particular agents:
 - TestAgent \rightarrow Test(x) and carriedOutOn(x,y)
 - SpecimenAgent \rightarrow Specimen(y) and form(y,bar) and carriedOutOn(x,y)
 Both agent received part of the query that corresponds to multiply entities. Since this is a relation between the two concepts, agents need to share the meaning of this expression. Agents place this into a blackboard, which is visible for all agents.
 - Blackboard \rightarrow carriedOutOn(x,y)
4. Test and Specimen agents retrieve fragments of two ontologies. Test Agent identifies two similar concepts:
 - $O_1 \rightarrow \text{TestResult}$ and $O_2 \rightarrow \text{Test}$
 Specimen Agent identifies two similar properties:
 - $O_1 \rightarrow \text{Form}$ and $O_2 \rightarrow \text{SpecimenForm}$
- a) Dempster-Shafer belief mass function is evaluated based on the node name similarities

TestAgent	SpecimenAgent
Test-TestResult=0.1	Specimen-Specimen=1.0
Control-TestControl=0.3	Form-SpecimenForm=0.3
Temperature-TestTemperature=0.4	Name-SpecimenName=0.25
Standard-TestStandard=0.2	Characterisation- SpecimenCharacterisation=.25

Table 1. Assigned belief function for the different entities.

- b) Dempster-Shafer belief mass function is evaluated (Table 1) based on the node structure similarities
 Test(Control, Temperature, Standard)- TestResult(TestControl, TestTemperature, TestStandard)= 0.5
 Specimen(Name, Form, Characterization) and Geometry(SpecimenForm, SpecimenName, SpecimenChar)=0.6
- c) Combined similarity, belief function can be calculated cooperatively by the two agents.
 TestResult in O_1 is similar concept to Test in O_2 with belief function 0.8
 Geometry in O_1 is similar concept to Specimen in O_2 and Form in O_1 is similar property in SpecimenForm in O_2
5. New findings can be added to the global descriptor:
 - $\forall x \text{ Testagent}(x) \text{ and canAnswer}(x, \text{TestResult})$
 - $\forall x \text{ Specimenagent}(x) \text{ and canAnswer}(x, \text{Geometry})$

6. Implementation

Our framework is implemented with JADE [9] agents using SWI prolog [10] engine to achieve reasoning capabilities. Because of the original ACL (Agent Communication language) implemented by JADE assumes that every used ontology is a subset of the domain ontology or there exists a map between it and the domain ontology; we defined our own agent

```
<acp>
  <Query>
    <QueryFragment>hasIdentifier(Material,Cr Mo 10)</QueryFragment>
  </Query>
</acp>

<acp>
  <Answer>
    <Similarity>
      <Class ID="Material">
        <Source ID="Ontology 1" BMF="1">Material</Source>
        <Source ID="Ontology 2" BMF="0.4">Subject</Source>
      </Class>
    </Similarity>
  </Answer>
</acp>
```

Figure 4. Agent Communication Protocol

communication protocol(Figure 4) that sits atop of the standard ACL messages and describes not only the similarity information but the quantitative measure of the uncertainty inherent to the mapping process. This protocol is a simple XML based communication protocol called ACP (Agent Communication Protocol) that is tightly integrated with the AQUA FOL formula representation and the specific nature of the question answering. The two main entities are the query and the answer. The sub elements in each node depend on which agent communicates with whom e.g. the query and answer structure between the broker and the mapping agents is depicted before.

6.1 Source ontologies

Our ontology O is defined by its set of concepts C (instances of “owl:Class”) with a corresponding relations R (instances of “owl:ObjectProperty or owl:”DataTypeProperty”) exist between single concepts. Ontologies that describe the entities in the different databases cover the main domain specific concepts like test result, source, material, specimen, test condition, etc. We assume that different institutions create their own domain specific ontology and since these domains describe

the same information in different domains their designers have a different conceptualization, which leads to a different definitions of concepts and relationships for same objects even if it is expressed in the same ontology language. The following example ontology fragments describe two data source where in ontology 1 there is a relation explicitly described between the TEST and the SPECIMEN whereas in the second example it is expressed through one unique property of the SPECIMEN namely the identifier.

Our examples are represented in OWL ontology language(Figure 5,6)

```

                                ONTOLOGY 1
<owl:Class rdf:ID="Test"/>
<owl:Class rdf:ID="Specimen"/>
<owl:DatatypeProperty rdf:ID="Control">
  <rdfs:domain rdf:resource="# Test"/>
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:ObjectProperty rdf:ID="hasSpecimen">
  <rdfs:range rdf:resource="#Test"/>
  <rdfs:domain rdf:resource="#Specimen"/>
</owl:ObjectProperty>

```

Figure 5. Sample ontology fragment

```

                                ONTOLOGY 2
<owl:Class rdf:ID="TestResult"/>
<owl:Class rdf:ID="Specimen"/>
<owl:DatatypeProperty rdf:ID="Control">
  <rdfs:domain rdf:resource="# Test"/>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>
<owl:DatatypeProperty rdf:ID="SpecimenIdentifier">
  <rdfs:domain rdf:resource="#TestResult"/>
  <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

```

Figure 6. Sample ontology fragment

7. Related work

Ontology mapping is widely investigated area and a numerous approaches led to different solutions.

Derived from the data engineering community several solutions have been proposed that based on a mediator architecture where logical database schemas are used as shared mediated views over the queried schemas. A number of systems have been proposed e.g TSIMMIS[11], Information Manifold [12], InfoSleuth [13], MOMIS [14] that shows the flexibility and the scalability of these approaches.

Derived from the knowledge engineering community solutions the use of ontologies (conceptual domain knowledge schemas) is the main approach for resolving semantic differences in heterogeneous data sources.

To date uncertainty handling during the mapping process was not in the focus of the research community since initially only different logic(FOL, Description Logics) based approaches has been utilized. As practical application of ontologies emerged on the web it has been acknowledged that considering the dynamic nature of the Web the problem of inconsistencies, controversies and lack of information needs to be handled. First systems that used probabilistic information like LSD, GLUE [15] proved that combining different similarity measures based on their probability could significantly improve the accuracy of the mapping process. It is worth to note that the Bayesian networks and different variants dominate current research addressing the qualitative reasoning and decision-making problem under uncertainty. Although these approaches successfully lead to numerous real world applications there are several situations where the problem cannot be represented properly within the classical probability framework. The most related research for ontology mapping framework under uncertainty using Bayesian networks [16] to tackle this problem.

8. Conclusion and future research

In our prototype we successfully addressed the problem of a single agent or application that is limited by its knowledge, perspective, and its computational resources. It is clear that if we try to move towards a fully automated ontology mapping in order to provide a better integration of the heterogeneous sources we need to investigate the limitations of multi agent systems such as our prototype. In this complex environment different scientific disciplines need to be utilized together to achieve better results to the users' query within an acceptable time frame. We think that in our implementation we have made a encouraging step towards a theoretical solution but the different key system components such as similarity measure or the uncertainty handling part needs to be investigated further. In our future research we are planning to establish a qualitative comparison of the similarity algorithms that fulfill all the requirements of our examined domain and our tasks.

We believe that probability theory and distribution does not have enough expressive power to tackle certain aspects of the uncertainty e.g. total ignorance.

As a consequence we expect that evidence (Dempster-Shafer) theory is the most suitable approach and needs to be investigated in ontology mapping context though this has not been done so far. The reason is that Dempster Shafer combination rule can easily be unfeasible in case of domains with large number of variables. Different optimisations methods have been developed but to date we could not find approaches that considered distributed environment. Local computation and valuation networks uses joint tree structure to narrow down the number of focal elements and different architectures has been proposed based on message passing schemes to carry our inference and resolve the problem of the Dempster's rule of combination. In our scenario we assume a dynamic multi agent environment where different agents has partial knowledge of the domain.

References

- [1] Vargas-Vera M. and Motta E. (2004) AQUA - Ontology-based Question Answering System. Third International Mexican Conference on Artificial Intelligence (MICAI-2004), Lecture Notes in Computer Science 2972 Springer Verlag, (eds R. Monroy et al.), April 26-30, 2004.
- [2] Vargas-Vera M., Motta E. and Domingue J. (2003) AQUA: An Ontology-Driven Question Answering System. AAAI Spring Symposium, New Directions in Question Answering, Stanford University, March 24-26, 2003.
- [3] MatML Materials Markup language, <http://www.matml.org>
- [4] Shafer Glenn,(1976) A Mathematical Theory of Evidence. Princeton University Press.
- [5] Haveliwala T, Gionis A., Klein D., and Indyk P (2002). Evaluating strategies for similarity search on the web. Proceedings of WWW, Hawaii, USA, May 2002.
- [6] Vargas-Vera M. and Motta E. (2004) A Knowledge-Based Approach to Ontologies Data Integration. KMi-TR-152, The Open University, July 2004.
- [7] Vargas-Vera M. and Motta E. (2004) An Ontology-driven Similarity Algorithm. KMI-TR-151, Knowledge Media Institute, The Open University, July 2004.
- [8] Atallah Mikhail J.,(1999) Algorithms and Theory of Computation Handbook, ed., CRC Press LLC, 1999.
- [9] JADE Web Site, <http://jade.tilab.com/>
- [10] Wielemaker J . SWI-Prolog 5.1: Reference Manual. SWI, University of Amsterdam, Roetersstraat 15, 1018 WB Amsterdam, The Netherlands, 1997-2003. E-mail: jan@swi.psy.uva.nl.
- [11] Garcia-Molina, H.; Papakonstantinou, Y.; Quass, D.; Rajararnan, A.; Sagiv, Y.; Ullman, J.; Vassalos, V.; Widom, J (1997) The TSIMMIS Approach to Mediation: Data Models and Languages, Journal of Intelligent Information Systems, 8(2):117-132.
- [12] A. Halevy (1998) The Information manifold approach to data integration.

- [13] Bayardo, R.; et al.. Infosleuth (1997) Agent-based Semantic Integration of Information in Open and Dynamic Environments. In Proceedings of ACM SIGMOD Conference on Management of Data, 195-206. Tucson, Arizona.
- [14] Beneventano, D.; Bergamaschi, S.; Guerra, F.; Vincini, M..(2001) The MOMIS Approach to Information Integration. In ICEIS(1), 194-198.
- [15] Doan, A. H.; Madhavan, J.; Domingos, P.; Halevy (2002) A. Learning to Map between Ontologies on the Semantic Web. In WWW 2002.
- [16] Zhongli Ding, Yun Peng, Rong Pan. A Bayesian Approach to Uncertainty Modeling in OWL Ontology. In Proceedings of 2004 International Conference on Advances in Intelligent Systems - Theory and Applications (AISTA2004). November 15-18, 2004, Luxembourg-Kirchberg, Luxembourg. in Intelligent Systems - Theory and Applications (AISTA2004). November 15-18, 2004, Luxembourg-Kirchberg, Luxembourg.