7

Towards an Automatic Semantic Data Integration: Multi-agent Framework Approach

Miklos Nagy and Maria Vargas-Vera
The Open University
United Kingdom

1. Introduction

The continuously increasing semantic meta data on the Web will soon make it possible to develop mature Semantic Web applications that have the potential to attract commercial players to contribute to the Semantic Web vision. This is especially important because it will assure that more and more people will start using Semantic Web based applications, which is a pre condition for commercial viability. 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 kind of "machine intelligence", which can support a variety of user tasks like improved search and question answering. To develop such applications researchers have developed a wide variety of building blocks that needs to be utilised together in order to achieve wider public acceptance. This is especially true for ontology mapping (Euzenat & Shvaiko, 2007), which makes it possible to interpret and align heterogeneous and distributed ontologies on the Semantic Web. However in order to simulate "machine intelligence" for ontology mapping different challenges have to be tackled.

Consider for example the difficulty of evaluating ontologies with large number of concepts. Due 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 their assessments are discussed and they produce a final assessment, which reflects their 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 experts who independently make assessments. This is because these experts combine the knowledge and experience to create a solution rather than relying on a single person's perspective. We focus our attention how this collective intelligence can be achieved by using software agents and what problems need to be addressed before one can achieve such machine intelligence for ontology mapping. Our work DSSim (Nagy et al., 2007) tries to tackle the different ontology representation, quality and size problems with providing a multi-agent ontology mapping framework, which tries to mimic the collective human actions for creating ontology mappings.

This chapter is organised as follows. In section 2 we describe the challenges and roadblock that we intend to address in our system. In section 3 the related work is presented. Section 4

describes our core multi agent ontology mapping framework. In section 5 we introduce how uncertainty is represented and in our system. Section 6 details how contradictions in belief similarity are modelled with fuzzy trust. Section 7 explains our experimental results and section 8 outlines the strengths and weaknesses of our system. In section 9 we draw our conclusions and discuss possible future research directions.

2. Challenges and roadblocks

Despite the fact that a number of ontology matching solutions have been proposed (Euzenat & Shvaiko, 2007) in the recent years none of them have proved to be an integrated solution, which can be used by different user communities. Several challenges have been identified by Shvaiko and Euzenat (Euzenat & Shvaiko, 2007), which are considered as major roadblocks for successful future implementations. To overcome a combination of these challenges was the main motivation of our work. It is easy to foresee that the combination of these challenges might differ depending on the different requirements of a particular ontology mapping solution. In the context of Question Answering, we have identified some critical and interrelated challenges, which can be considered as roadblocks for future successful implementations (Nagy et al., 2008). The main motivation of our work, which is presented in this chapter, was to overcome the combination of these challenges.

2.1 Representation problems and uncertainty

The vision of the Semantic Web is to achieve machine-processable interoperability through the annotation of the content. This implies that computer programs can achieve a certain degree of understanding of such data and use it to reason about a user specific task like question answering or data integration.

Data on the semantic web is represented by ontologies, which typically consist of a number of classes, relations, instances and axioms. These elements are expressed using a logical language. The W3C has proposed RDF(S) (Beckett, 2004) and OWL (McGuinness & Harmelen, 2004) as Web ontology language however OWL has three increasingly-expressive sublanguages(OWL Lite, OWL DL, OWL Full) with different expressiveness and language constructs. In addition to the existing Web ontology languages W3C has proposed other languages like SKOS (Miles & Bechofer, 2008), which is a standard to support the use of knowledge organization systems (KOS) such as thesauri, classification schemes, subject heading systems and taxonomies within the framework of the Semantic Web. SKOS are based on the Resource Description Framework (RDF) and it allows information to be passed between computer applications in an interoperable way. Ontology designers can choose between these language variants depending on the intended purpose of the ontologies. The problem of interpreting semantic web data however stems not only from the different language representations (Lenzerini et al., 2004) but the fact that ontologies especially OWL Full has been designed as a general framework to represent domain knowledge, which in turn can differ from designer to designer. Consider the following excerpts **Fig. 1** from different FAO (Food and Agricultural Organization of the United Nations) ontologies. Assume we need to assess similarity between classes and individuals between the two ontologies. In fragment one a class *c_8375* is modelled as named OWL individuals. In the class description only the ID is indicated therefore to determine the properties of the class

one needs to extract the necessary information from the actual named individual. In **Fig. 1** (right) the classes are represented as RDF individuals where the individual properties are defines as OWL data properties. One can note the difference how the class labels are represented on **Fig. 1** (left) through *rdfs:label* and **Fig. 1** (right) through through *hasNameScientific* and *hasNameLongEN* tags.

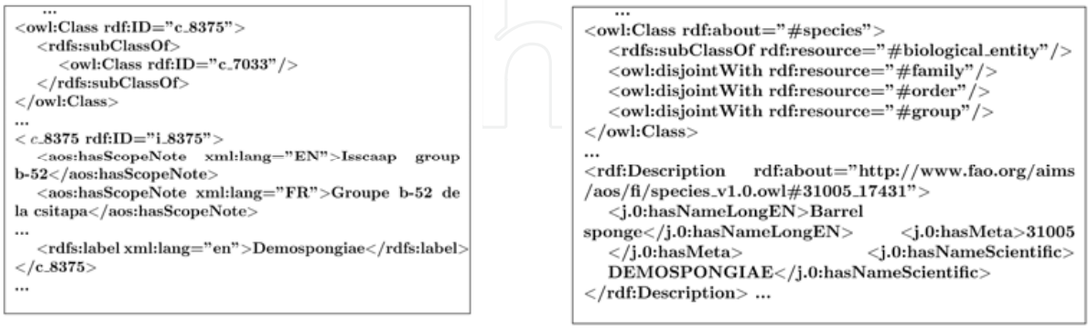

Fig. 1. Ontology fragments from the AGROVOC and ASFA ontology

From the logical representation point of view both ontologies are valid separately and no logical reasoner would find inconsistency in them individually. However the problem occurs once we need to compare them in order to determine the similarities between classes and individuals. It is easy to see that once we need to compare the two ontologies a considerable amount of uncertainty arises over the classes and its properties and in a way they can be compared. This uncertainty can be contributed to the fact that due to the different representation certain elements will be missing for the comparison e.g. we have label in fragment **Fig. 1** (left) but is missing from fragment **Fig. 1** (right) but there is *hasNameLongEN* tag in fragment **Fig. 1** (right) but missing in fragment **Fig. 1** (left). As a result of these representation differences ontology mapping systems will always need to consider the uncertain aspects of how the semantic web data can be interpreted.

2.2 Quality of Semantic Web Data

Data quality problems (Wang et al., 1993) (Wand & Wang, 1996) in the context of database integration (Batini et al., 1986) have emerged long before the Semantic Web concept has been proposed. The major reason for this is the increase in interconnectivity among data producers and data consumers, mainly spurred through the development of the Internet and various Web-based technologies. For every organisation or individual the context of the data, which is published can be slightly different depending on how they want to use their data. Therefore from the exchange point of view incompleteness of a particular data is quite common. The problem is that fragmented data environments like the Semantic Web inevitably lead to data and information quality problems causing the applications that process this data deal with ill-defined inaccurate or inconsistent information on the domain. The incomplete data can mean different things to data consumer and data producer in a given application scenario. In traditional integration scenarios resolving these data quality issues represents a vast amount of time and resources for human experts before any integration can take place. Data quality has two aspects

- Data syntax covers the way data is formatted and gets represented
- Data semantics addresses the meaning of data

Data syntax is not the main reason of concern as it can be resolved independently from the context because it can be defined what changes must occur to make the data consistent and standardized for the application e.g. defining a separation rule of compound terms like "MScThesis", "MSc_Thesis". The main problem what Semantic Web applications need to solve is how to resolve semantic data quality problems i.e. what is useful and meaningful because it would require more direct input from the users or creators of the ontologies. Clearly considering any kind of designer support in the Semantic Web environment is unrealistic therefore applications itself need to have built in mechanisms to decide and reason about whether the data is accurate, usable and useful in essence, whether it will deliver good information and function well for the required purpose. Consider the following example **Fig. 2** from the directory ontologies.

```
...
<owl:Class rdf:about="http://matching.com/source
/3887.owl#Windows_Vista">
<rdfs:label xml:lang="en"> Windows Vista Home
Edition </rdfs:label>
<j:hasSerialNumber>
<rdfs:label >00043-683-036-658</rdfs:label>
</j:hasSeriamNumber>
<rdfs:subClassOf>
<owl:Class rdf:about="http://matching.com
/source/3887.owl#Operating_Systems">
</owl:Class>
</rdfs:subClassOf>
</owl:Class>
...
```

Fig. 2. Ontology fragments from the Web directories ontology

As figure **Fig. 2** shows we can interpret Windows Vista as the subclass of the Operating systems however the designed has indicated that it has a specific serial number therefore it can be considered as an individual. At any case the semantic data quality is considered as low as the information is dubious therefore the Semantic Web application has to create its own hypothesises over the meaning of this data.

2.3 Efficient ontology mapping with large scale ontologies

Ontologies can get quite complex and very large, causing difficulties in using them for any application (Carlo et al., 2005) (Flahive et al., 2006). This is especially true for ontology mapping where overcoming scalability issues becomes one of the decisive factors for determining the usefulness of a system. Nowadays with the rapid development of ontology applications, domain ontologies can become very large in scale. This can partly be contributed to the fact that a number of general knowledge bases or lexical databases have been and will be transformed into ontologies in order to support more applications on the

Semantic Web. Consider for example WordNet. Since the project started in 1985 WordNet[1] has been used for a number of different purposes in information systems. It is popular general background knowledge for ontology mapping systems because it contains around 150.000 synsets and their semantic relations. Other efforts to represent common sense knowledge as ontology is the Cyc project[2] , which consists of more than 300.000 concepts and nearly 3.000.000 assertions or the Suggested Upper Merged Ontology (SUMO)[3] with its 20.000 terms and 70.000 axioms when all domain ontologies are combined. However the far largest ontology so far (according to our knowledge) in terms of concept number is the DBPedia[4] , which contains over 2.18 million resources or "things", each tied to an article in the English language Wikipedia. Discovering correspondences between these large-scale ontologies is an ongoing effort however only partial mappings have been established i.e. SUMO-Wordnet due to the vast amount of human and computational effort involved in these tasks. The Ontology Alignment Initiative 2008 (Caracciolo et al., 2008) has also included a mapping track for very large cross lingual ontologies, which includes establishing mappings between Wordnet, DBPedia and GTAA (Dutch acronym for Common Thesaurus for Audiovisual Archives) (Brugman et al., 2006), which is a domain specific thesaurus with approximately 160.000 terms. A good number of researchers might argue that the Semantic Web is not just about large ontologies created by the large organisations but more about individuals or domain experts who can create their own relatively small ontologies and publish it on the Web. Indeed might be true however from the scalability point of view it does not change anything if thousands of small ontologies or a number of huge ontologies need to be processed. Consider that in 2007 Swoogle (Ding et al., 2004) has already indexed more than 10.000 ontologies, which were available on the Web. The large number of concepts and properties that is implied by the scale or number of these ontologies poses several scalability problems from the reasoning point of view. Any Semantic Web application not only from ontology mapping domain has to be designed to cope with these difficulties otherwise it is deemed to be a failure from the usability point of view.

3. Related Work

Several ontology-mapping systems have been proposed to address the semantic data integration problem of different domains independently. In this paper we consider only those systems, which have participated in the OAEI (Ontology Alignment Evaluation Initiative) competitions and has been participated more than two tracks. There are other proposed systems as well however as the experimental comparison cannot be achieved we do not include them in the scope of our analysis. Lily (Wang & Xu, 2008) is an ontology mapping system with different purpose ranging from generic ontology matching to mapping debugging. It uses different syntactic and semantic similarity measures and combines them with the experiential weights. Further it applies similarity propagation

[1] http://wordnet.princeton.edu/
[2] http://www.cyc.com/
[3] http://www.ontologyportal.org/
[4] http://dbpedia.org/About

matcher with strong propagation condition and the matching algorithm utilises the results of literal matching to produce more alignments. In order to assess when to use similarity propagation Lily uses different strategies, which prevents the algorithm from producing more incorrect alignments.

ASMOV (Jean-Mary & Kabuka, 2007) has been proposed as a general mapping tool in order to facilitate the integration of heterogeneous systems, using their data source ontologies. It uses different matchers and generates similarity matrices between concepts, properties, and individuals, including mappings from object properties to datatype properties. It does not combine the similarities but uses the best values to create a pre alignment, which are then being semantically validated. Mappings, which pass the semantic validation will be added to the final alignment. ASMOV can use different background knowledge e.g. Wordnet or UMLS Metathesaurus (medical background knowledge) for the assessment of the similarity measures.

RiMOM (Tang et al., 2006) is an automatic ontology mapping system, which models the ontology mapping problem as making decisions over entities with minimal risk. It uses the Bayesian theory to model decision-making under uncertainty where observations are all entities in the two ontologies.

Further it implements different matching strategies where each defined strategy is based on one kind of ontological information. RiMOM includes different methods for choosing appropriate strategies (or strategy combination) according to the available information in the ontologies. The strategy combination is conducted by a linear-interpolation method. In addition to the different strategies RiMOM uses similarity propagation process to refine the existing alignments and to find new alignments that cannot be found using other strategies. RiMOM is the only system other than DSSim in the OAEI contest that considers the uncertain nature of the mapping process however it models uncertainty differently from DSSim. RiMOM appeared for first time in the OAEI-2007 whilst DSSim appeared in the OAEI-2006.

MapPSO (Bock & Hettenhausen, 2008) is a research prototype, which has been designed to address the need for highly scalable, massively parallel tool for both large scale and numerous ontology alignments. MapPSO method models the ontology alignment problem as an optimisation problem. It employs a population based optimisation paradigm based on social interaction between swarming animals, which provides the best answer being available at that time. Therefore it is especially suitable for providing answers under time constraint like the ontology mapping. MapPSO employs different syntactic and semantic similarity measures and combines the available base distances by applying the Ordered Weighted Average(OWA) (Ji et al., 2008) aggregation of the base distances. It aggregates the components by ordering the base distances and applying a fixed weight vector. The motivation of the MapPSO system is identical with one of the motivations of the DSSim namely to address the need of scalable mapping solutions for large-scale ontologies. Surprisingly MapPSO did not participate in the Very Large Cross Lingual Resources track (especially designed for large scale thesauri) therefore experimental comparison cannot be achieved from this point of view.

TaxoMap (Hamdi at al., 2008) is an alignment tool, which aims is to discover rich correspondences between concepts with performing oriented alignment from a source to a target ontology taking into account labels and sub-class descriptions. It uses a part-of-speech

(Schmid 1994) and lemma information, which enables to take into account the language, lemma and use word categories in an efficient way. TaxoMap performs a linguistic similarity measure between labels and description of concepts and it has been designed to process large scale ontologies by using partitioning techniques. TaxoMap however does not process instances, which can be a drawback in several situations.
SAMBO and SAMBOdtf (Lambrix & Tan, 2006) is a general framework for ontology matching. The methods and techniques used in the framework are general and applicable to different areas nevertheless SAMBO has been designed to align biomedical ontologies. Their algorithm includes one or several matchers, which calculate similarity values between the terms from the different source ontologies. These similarities are then filtered and combined as a weighted sum of the similarity values computed by different matchers.

4. Multi agent ontology mapping framework

For ontology mapping in the context of Question Answering over heterogeneous sources we propose a multi agent architecture (Nagy et al., 2005) because as a particular domain becomes larger and more complex, open and distributed, a set of cooperating agents are necessary in order to address the ontology mapping task effectively. In real scenarios, ontology mapping can be carried out on domains with large number of classes and properties. Without the multi agent architecture the response time of the system can increase exponentially when the number of concepts to map increases. The main objective of DSSim architecture is to be able to use it in different domains for creating ontology mappings. These domains include Question Answering, Web services or any application that need 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 our implementation we have used the AQUA (Vargas-Vera & Motta, 2004) Question Answering system, 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 component are valid FOL formulas.

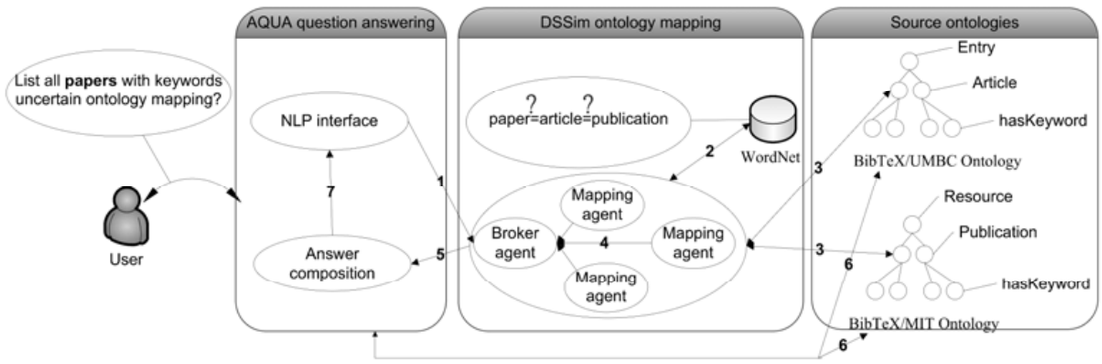

Fig. 3. Overview of the mapping system

An overview of our system is depicted on **Fig. 3** The two real word ontologies[5][6] describe BibTeX publications from the University of Maryland, Baltimore County (UMBC) and from the Massachusetts Institute of Technology (MIT).
The AQUA (Vargas-Vera & Motta, 2004) system and the answer composition component are described just to provide the context of our work (our overall framework) but these are not our major target in this paper. The user poses a natural language query to the AQUA system, which converts it into FOL (First Order Logic) terms.
The main components and its functions of the system are as follows:

1. Broker agent receives FOL term, decomposes it(in case more than one concepts are in the query) and distributes the sub queries to the mapping agents.
2. Mapping agents retrieve sub query class and property hypernyms from WordNet.
3. 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 in order to enhance their beliefs on the possible meaning of the concepts or properties in the particular context.
4. 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 build their beliefs over the correctness of the mapping.
5. Broker agent passes the possible mappings into the answer composition component for particular sub-query ontology fragment mapping in which the belief function has the highest value.
6. Answer composition component retrieves the concrete instances from the external ontologies or data sources, which is included into the answer.
7. Answer composition component creates an answer to the user's question.

The main novelty in our solution is that we propose solving 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.

4.1 Example scenario

Based on the architecture depicted on **Fig. 3** we present the following simplified example, which will be used in the following sections of the paper in order to demonstrate our algorithm. We consider the following user query and its FOL representation as an input to our mapping component framework:
List all papers with keywords uncertain ontology mapping?

$$(\exists x)paper(x)\ and\ hasKeywords(x,[uncertain,ontology\ mapping]) \quad (1)$$

[5] http://ebiquity.umbc.edu/ontology/publication.owl
[6] http://visus.mit.edu/bibtex/0.01/bibtex.owl

- Step 1: Broker agent distributes (no decomposition is necessary in this case) the FOL query to the mapping agents.
- Step 2: Mapping agents 1 and 2 consult WordNet in order to extend the concepts and properties with their inherited hypernym in the query. These hypernyms serve as variables in the hypothesis. For the concepts "paper" e.g. we have found that "article" and "communication" or "publication" are possible concepts that can appear in any of the external ontologies.
- Step 3: Mapping agents iterate through all concepts and properties from the ontologies and create several hypothesises that must be verified with finding evidences e.g.

$$Agent\,1 : H_n(mapping) = Query\{paper, article, communication, publication\} \Leftrightarrow Onto\log y_{MIT}\{Article\} \quad (2)$$

and

$$Agent\,2 : H_n(mapping) = Query\{paper, article, communication, publication\} \Leftrightarrow Onto\log y_{UMBC}\{Publication\} \quad (3)$$

where H is the hypothesis for the mapping.

Further, we find supporting evidences for hypothesis. In this phase different syntactic and semantic similarity measures are used (see subsection 5.1, 5.2). These similarity measures are considered as different experts determining belief functions for the hypothesis. The last phase of this step is to combine the belief mass functions using Dempster's combination rule in order to form a coherent belief of the different experts on the hypothesises.

- Step 4: Mapping agents select the hypothesis in which they believe in most and sent it back to the broker agent. In our example the following mappings have been established:

$$Mapping_{Query, MIT\ onto\log y}(paper \leftrightarrow article) \quad (4)$$
$$Mapping_{Query, UMBC\ onto\log y}(paper \leftrightarrow publication)$$

- Step 5-6: The answer is composed for the user's query, which includes the relevant instances from the ontologies.

5. Uncertain reasoning and agent belief

Our proposed method works with two ontologies, which contain arbitrary number of concepts and their properties.

$$O_1 = \{C_1,..,C_n;P_1,..,P_n;I_1,..,I_n\} \quad (5)$$
$$O_2 = \{C_1,..,C_m;P_1,..,P_m;I_1,..,I_m\}$$

where *O* represents a particular ontology, *C, P* and *I* the set of concepts, properties and instances in the ontology.

In order to assess similarity we need to compare all concepts and properties from O_1 to all concepts and properties in O_2. Our similarity assessments, both syntactic and semantic produce a sparse similarity matrix where the similarity between C_n from O_1 and C_m in O_2 is represented by a particular similarity measure between the i and j elements of the matrix as follows:

$$SIM := (S_{i,j})_{n \times m} \qquad (6)$$
$$1 \le i \le n \quad and \quad 1 \le j \le m$$

where *SIM* represents a particular similarity assessment matrix, *s* is a degree of similarity that has been determined by a particular similarity e.g. Jaccard or semantic similarity measure. We consider each measure as an "expert", which assess mapping precision based on its knowledge. Therefore we assume that each similarity matrix is a subjective assessment of the mapping what needs to be combined into a coherent view. If combined appropriately this combined view provides a more reliable and precise mapping that each separate mapping alone. However one similarity measure or some technique can perform particularly well for one pair of concepts or properties and particularly badly for another pair of concepts or properties, which has to be considered in any mapping algorithm. In our ontology mapping framework each agent carries only partial knowledge of the domain and can observe it from its own perspective where available prior knowledge is generally uncertain. Our main argument is that knowledge cannot be viewed as a simple conceptualization of the world, but it has to represent some degree of interpretation. Such interpretation depends on the context of the entities involved in the process. This idea is rooted in the fact the different entities' interpretations are always subjective, since they occur according to an individual schema, which is than communicated to other individuals by a particular language. In order to represent these subjective probabilities in our system we use the Dempster-Shafer theory of evidence (Shafer, 1976), which provides a mechanism for modelling and reasoning uncertain information in a numerical way, particularly when it is not possible to assign belief to a single element of a set of variables. Consequently the theory allows the user to represent uncertainty for knowledge representation, because the interval between support and plausibility can be easily assessed for a set of hypothesizes. Missing data (ignorance) can also be modelled by Dempster-Shafer approach and additionally evidences from two or more sources can be combined using Dempster's rule of combination. The combined support, disbelief and uncertainty can each be separately evaluated. The main advantage of the Dempster-Shafer theory is that it provides a method for combining the effect of different learned evidences to establish a new belief by using Dempster's combination rule. The following elements have been used in our system in order to model uncertainty:

Frame of Discernment(Θ): finite set representing the space of hypothesizes. It contains all possible mutually exclusive context events of the same kind.

$$\Theta = \{H_1, .., H_n, .., H_N\} \qquad (7)$$

In our method Θ contains all possible mappings that have been assessed by the particular expert.

Evidence: available certain fact and is usually a result of observation. Used during the reasoning process to choose the best hypothesis in Θ.

We observe evidence for the mapping if the expert detects that there is a similarity between C_n from O_1 and C_m in O_2.

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

$$\sum_{A \subseteq \Theta} m_i(A) = 1 \qquad (8)$$

where $m_i(A)$ is our exact belief in a proposition represented by A that belongs to expert *i*. The similarity algorithms itself produce these assignment based on different similarity measures. As an example consider that O_1 contains the concept "paper", which needs to be mapped to a concept "*hasArticle*" in O_2. Based on the WordNet we identify that the concept "article" is one of the inherited hypernyms of "paper", which according to both JaroWinkler(0.91) and Jaccard(0.85) measure (Cohen et al., 2003) is highly similarity to "has Article" in O_2. Therefore after similarity assessment our variables will have the following belief mass value:

$$m_{expert1}\left(O_1\{paper, article, communication, publication\}, O_2\{hasArticle\}\right) = 0.85 \quad (9)$$
$$m_{expert2}\left(O_1\{paper, article, communication, publication\}, O_2\{hasArticle\}\right) = 0.91$$

In practice we assess up to 8 inherited hypernyms similarities with different algorithms (considered as experts), which can be combined based on the combination rule in order to create a more reliable mapping. Once the combined 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.

$$belief_i(A) = \sum_{E_k \subseteq A} m_i(E_k) \quad (10)$$

An important aspect of the mapping is how one can make a decision over how different similarity measures can be combined and which nodes should be retained as best possible candidates for the match. To combine the qualitative similarity measures that have been converted into belief mass functions we use the Dempster's rule of combination and we retain the node where the belief function has the highest value.

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

$$m_{ij}(A) = m_i \oplus m_j = \sum_{E_k E_{k'}} m_i(E_k) * m_j(E_{k'}) \quad (11)$$

where *i* and *j* represent two different agents.

The belief combination process is computationally very expensive and from an engineering point of view, this means that it not always convenient or possible to build systems in which the belief revision process is performed globally by a single unit. Therefore, applying multi agent architecture is an alternative and distributed approach to the single one, where the belief revision process is no longer assigned to a single agent but to a group of agents, in which each single agent is able to perform belief revision and communicate with the others. Our algorithm takes all the concepts and its properties from the different external ontologies and assesses similarity with all the concepts and properties in the query graph.

5.1 Syntactic Similarity

To assess syntactic similarity between ontology entities we use different string-based techniques to match names and name descriptions. These distance functions map a pair of strings to a real number, which indicates a qualitative similarity between the strings. To achieve more reliable assessment and to maximize our system's accuracy we combine different string matching techniques such as edit distance like functions e.g. Monger-Elkan (Monge & Elkan, 1996) to the token based distance functions e.g. Jaccard (Cohen et al., 2003)

similarity. To combine different similarity measures we use Dempster's rule of combination. Several reasonable similarity measures exist however, each being appropriate to certain situations. At this stage of the similarity mapping our algorithm takes one entity from Ontology 1 and tries to find similar entity in the extended query. The similarity mapping process is carried out on the following entities:

- Concept name similarity
- Property name and set similarity

The use of string distances described here is the first step towards identifying matching entities between query and the external ontology or between ontologies with little prior knowledge. However, string similarity alone is not sufficient to capture the subtle differences between classes with similar names but different meanings. Therefore we work with WordNet in order to exploit hypernymy at the lexical level. Once our query sting is extended with lexically hypernym entities we calculate the string similarity measures between the query and the ontologies. In order to increase the correctness of our similarity measures the obtained similarity coefficients need to be combined. Establishing this combination method was our primary objective that had been included into the system. Further once the combined similarities have been calculated we have developed a simple methodology to derive the belief mass function that is the fundamental property of Demster-Shafer framework.

5.2 Semantic Similarity

For semantic similarity between concept, relations and the properties we use graph-based techniques. We take the extended query and the ontology input as labelled graphs. The semantic matching is viewed as graph like structures containing terms and their inter-relationships. The similarity comparison between a pair of nodes from two ontologies is based on the analysis of their positions within the graphs. Our assumption is that if two nodes from two ontologies are similar, their neighbours might also be somehow similar. We consider 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, which as a result contain many overlapping concepts. Assessing the above mentioned similarities in our system we adapted and extended the SimilarityBase and SimilarityTop algorithms (Vargas-Vera & Motta, 2004) used in the current AQUA system for multiple ontologies. Our aim is that the similarity algorithms (experts in terms of evidence theory) would mimic the way a human designer would describe a domain based on a well-established dictionary. What also needs to be considered when the two graph structures are 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 external source and this needs to be handled correctly. In our system we adapted and extended the before mentioned SimilarityBase and SimilarityTop algorithms, which has been proved effective in the current AQUA system for multiple ontologies.

6. Conflict resolution with fuzzy voting model

The idea of individual voting in order to resolve conflict and choose the best option available is not rooted in computer but political science. Democratic systems are based on voting as Condorcet jury theorem (Austen-Smith & Banks, 1996) (Young, 1988) postulates that a group of voters using majority rule is more likely to choose the right action than an arbitrary single voter is. In these situations voters have a common goal, but do not know how to obtain this goal. Voters are informed differently about the performance of alternative ways of reaching it. If each member of a jury has only partial information, the majority decision is more likely to be correct than a decision arrived at by an individual juror. Moreover, the probability of a correct decision increases with the size of the jury. But things become more complicated when information is shared before a vote is taken. People then have to evaluate the information before making a collective decision. The same ideas apply for software agents especially if they need to reach a consensus on a particular issue. In case of ontology mapping where each agent can built up beliefs over the correctness of the mappings based on partial information we believe that it is voting can find the socially optimal choice. Software agents can use voting to determine the best decision for agent society but in case voters make mistakes in their judgments, then the majority alternative (if it exists) is statistically most likely to be the best choice. The application of voting for software agents is a possible way to make systems more intelligent i.e. mimic the decision making how humans reach consensus decision on a problematic issue.

6.1 Fuzzy voting model

In ontology mapping the conflicting results of the different beliefs in similarity can be resolved if the mapping algorithm can produce an agreed solution, even though the individual opinions about the available alternatives may vary. We propose a solution for reaching this agreement by evaluating trust between established beliefs through voting, which is a general method of reconciling differences. Voting is a mechanism where the opinions from a set of votes are evaluated in order to select the alternatives that best represent the collective preferences. Unfortunately deriving binary trust like trustful or not trustful from the difference of belief functions is not so straightforward since the different voters express their opinion as subjective probability over the similarities. For a particular mapping this always involves a certain degree of vagueness hence the threshold between the trust and distrust cannot be set definitely for all cases that can occur during the process. Additionally there is no clear transition between characterising a particular belief highly or less trustful. Therefore our argument is that the trust membership or belief difference values, which are expressed by different voters can be modelled properly by using fuzzy representation as depicted on **Fig. 4**. Before each agent evaluates the trust in other agent's belief over the correctness of the mapping it calculates the difference between its own and the other agent's belief. Depending on the difference it can choose the available trust levels e.g. if the difference in beliefs is 0.2 then the available trust level can be high and medium. We model these trust levels as fuzzy membership functions. In fuzzy logic the membership function $\mu(x)$ is defined on the universe of discourse U and represents a particular input value as a member of the fuzzy set i.e. $\mu(x)$ is a curve that defines how each point in the U is mapped to a membership value (or degree of membership) between 0 and 1. Our ontology

mapping system models the conflict resolution as a fuzzy system where the system components are described in the following subsections.

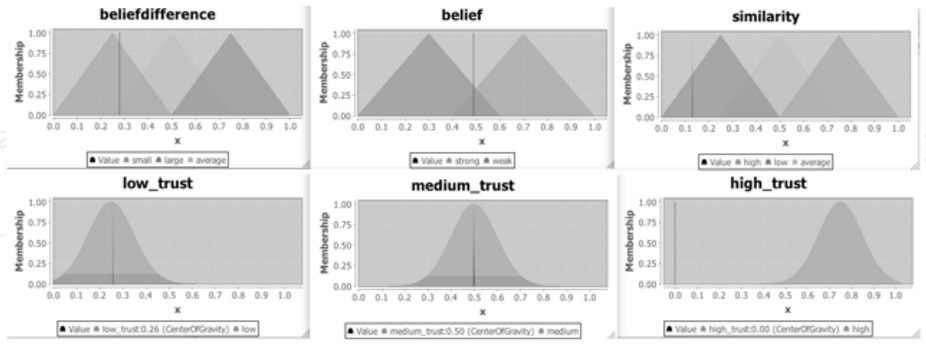

Fig. 4. Example fuzzy membership functions

6.1.1 Fuzzification of input and output variables

Fuzzification is the process of decomposing a system input and/or output into one or more fuzzy sets. We have experimented different types of curves namely the triangular, trapezoidal and gauss shaped membership functions. **Fig. 4** shows a system of fuzzy sets for an input with triangular and gauss membership functions. Each fuzzy set spans a region of input (or output) value graphed with the membership. Our selected membership functions overlap to allow smooth mapping of the system. The process of fuzzification allows the system inputs and outputs to be expressed in linguistic terms so that rules can be applied in a simple manner to express a complex system.

Belief difference is an input variable, which represents the agents own belief over the correctness of a mapping in order to establish mappings between concepts and properties in the ontology. During conflict resolution we need to be able to determine the level of difference. We propose three values for the fuzzy membership value $\mu(x)$*={small, average, large}*

Belief is an input variable, which described the 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.

$$belief_i(A) = \sum_{E_k \subseteq A} m_i(E_k) \tag{12}$$

where *m* Demster's belief mass function represents the strength of some evidence i.e. *m(A)* is our exact belief in a proposition represented by A. The similarity algorithms itself produce these assignment based on different similarity measures. We propose three values for the fuzzy membership value $\nu(x)$*={weak, strong}*

Similarity is an input variable and is the result of some syntactic or semantic similarity measure. We propose three values for the fuzzy membership value $\zeta(x)$*={low, average, high}*

Low, medium and high trusts are output variables and represent the level of trust we can assign to the combination of our input variables. We propose three values for the fuzzy membership value $\tau(x)$*={low, medium, high}*

6.1.2 Rule set

Fuzzy sets are used to quantify the information in the rule-base, and the inference mechanism operates on fuzzy sets to produce fuzzy sets. Fuzzy systems map the inputs to the outputs by a set of *condition* $\Rightarrow$ *action* rules i.e. rules that can be expressed in *If-Then* form. For our conflict resolution problem we have defined four simple rules Fig. 5. that ensure that each combination of the input variables produce output on more than one output i.e. there is always more than one initial trust level is assigned to any input variables.

RULE 1 : IF (beliefdifference IS large OR beliefdifference IS average) AND belief IS weak AND (similarity IS low OR similarity IS average) THEN low_trust IS low;
RULE 2 : IF (beliefdifference IS large OR beliefdifference IS average) AND belief IS weak AND (similarity IS low OR similarity IS average) THEN medium_trust IS medium;
RULE 3 : IF (beliefdifference IS small OR beliefdifference IS average) AND belief IS strong AND (similarity IS high OR similarity IS average) THEN high_trust IS high;
RULE 4 : IF (beliefdifference IS small OR beliefdifference IS average) AND belief IS strong AND (similarity IS high OR similarity IS average) THEN medium_trust IS medium;

Fig. 5. Fuzzy rules for trust assessment

6.1.3 Defuzzification method

After fuzzy reasoning we have the linguistic output variables, which need to be translated into a crisp (i.e. real numbers, not fuzzy sets) value. The objective is to derive a single crisp numeric value that best represents the inferred fuzzy values of the linguistic output variable. Defuzzification is such inverse transformation, which maps the output from the fuzzy domain back into the crisp domain.

In our ontology mapping system we have selected the Center-of-Area (C-o-A) defuzzification method. The C-o-A method is often referred to as the Center-of-Gravity method because it computes the centroid of the composite area representing the output fuzzy term. In our system the trust levels are proportional with the area of the membership functions therefore other defuzzification methods like Center-of-Maximum (C-o-M) or Mean-of-Maximum (M-o-M) does not correspond well to our requirements. For representing trust in beliefs over similarities we have defined three membership functions, $\tau(x)$=*{low, average, high}* in the beliefs over concept and property similarities in our ontology mapping system. Our main objective is to be able to resolve conflict between two beliefs in Dempster-Shafer theory, which can be interpreted qualitatively as one source strongly supports one hypothesis and the other strongly supports another hypothesis, where the two hypotheses are not compatible. Consider for example a situation where three agents have used WordNet as background knowledge and build their beliefs considering different concepts context, which was derived from the background knowledge e.g. agent 1 used the direct hypernyms, agent 2 the sister terms and agent 3 the inherited hypernyms. Based on string similarity measures a numerical belief value is calculated, which represent a strength of the confidence that the two terms are related to each other. The scenario is depicted in Table 1.

CONFLICT DETECTION	BELIEF 1	BELIEF 2	BELIEF 3
Obvious	0.85	0.80	0.1
Difficult	0.85	0.65	0.45

Table 1. Belief conflict detection

The values given in **Table 1** are demonstrative numbers just for the purpose of providing an example. In our ontology mapping framework DSSim, the similarities are considered as subjective beliefs, which is represented by belief mass functions that can be combined using the Dempster's combination rule. This subjective belief is the outcome of a similarity algorithm, which is applied by a software agent for creating mapping between two concepts in different ontologies. In our ontology mapping framework different agents assess similarities and their beliefs on the similarities need to be combined into a more coherent result. However these individual beliefs in practice are often conflicting. In this scenario applying Dempster's combination rule to conflicting beliefs can lead to an almost impossible choice because the combination rule strongly emphasizes the agreement between multiple sources and ignores all the conflicting evidence through a normalization factor. The counter-intuitive results that can occur with Dempster's rule of combination are well known and have generated a great deal of debate within the uncertainty reasoning community. Different variants of the combination rule (Sentz & Ferson, 2002) have been proposed to achieve more realistic combined belief. Instead of proposing an additional combination rule we turned our attention to the root cause of the conflict itself namely how the uncertain information was produced in our model.

The fuzzy voting model was developed by Baldwin (Baldwin, 1999) and has been used in Fuzzy logic applications. However, to our knowledge it has not been introduced in the context of trust management on the Semantic Web. In this section, we will briefly introduce the fuzzy voting model theory using a simple example of 10 voters voting against or in favour of the trustfulness of an another agent's belief over the correctness of mapping. In our ontology mapping framework each mapping agent can request a number of voting agents to help assessing how trustful the other mapping agent's belief is. According to Baldwin (Baldwin, 1999) a linguistic variable is a quintuple $(L,T(L),U,G,\mu)$ in which L is the name of the variable, $T(L)$ is the term set of labels or words (i.e. the linguistic values), U is a universe of discourse, G is a syntactic rule and μ is a semantic rule or membership function. We also assume for this work that G corresponds to a null syntactic rule so that $T(L)$ consists of a finite set of words. A formalization of the fuzzy voting model can be found in (Lawry, 1998). Consider the set of words *{Low_trust (L_t), Medium_trust (M_t) and High_trust (H_t) }* as labels of a linguistic variable trust with values in $U=[0,1]$. Given a set "m" of voters where each voter is asked to provide the subset of words from the finite set $T(L)$, which are appropriate as labels for the value u. The membership value $\chi_{\mu(w)(u)}$ is taking the proportion of voters who include u in their set of labels, which is represented by w. The main objective when resolving conflict is to have sufficient number of independent opinions that can be consolidated. To achieve our objective we need to introduce more opinions into the system i.e. we need to add the opinion of the other agents in order to vote for the best possible outcome. Therefore we assume for the purpose of our example that we have 10 voters (agents). Formally, let us define

$$V = A_1, A_2, A_3, A_4, A_5, A_6, A_7, A_8, A_9, A_{10} \quad (13)$$
$$T(L) = \{L_t, M_t, H_t\}$$

The number of voters can differ however assuming 10 voters can ensure that

1. The overlap between the membership functions can proportionally be distributed on the possible scale of the belief difference [0..1]
2. The work load of the voters does not slow the mapping process down

Let us start illustrating the previous ideas with a small example. By definition consider three linguistic output variables *L* representing trust levels and *T(L)* the set of linguistic values as *T(L)={Low_trust, Medium_trust, High_trust}*. The universe of discourse is *U*, which is defined as *U=[0,1]*. Then, we define the fuzzy sets per output variables μ (Low_trust), μ(Medium_trust) and μ(High_trust) for the voters where each voter has different overlapping trapezoidal, triangular or gauss membership functions as depicted on **Fig. 4**. The difference in the membership functions represented by the different vertices of the membership functions in **Fig. 1**4 ensures that voters can introduce different opinions as they pick the possible trust levels for the same difference in belief. The possible set of trust levels *L=TRUST* is defined by the **Table 2**. Note that in the table we use a short notation *L_t* means *Low_trust*, *M_t* means *Medium_trust* and *H_t* means *High_trust*. Once the input fuzzy sets (membership functions) have been defined the system is ready to assess the output trust memberships for the input values. Both input and output variables are real numbers on the range between [0..1]. Based on the difference of beliefs, own belief and similarity of the different voters the system evaluates the scenario. The evaluation includes the fuzzification which converts the crisp inputs to fuzzy sets, the inference mechanism which uses the fuzzy rules in the rule-base to produce fuzzy conclusions (e.g., the implied fuzzy sets), and the defuzzification block which converts these fuzzy conclusions into the crisp outputs. Therefore each input (belief difference, belief and similarity) produces a possible defuzzified output (low, medium or high trust) for the possible output variables. Each defuzzified value can be interpreted as a possible trust level where the linguistic variable with the highest defuzzified value is retained in case more than one output variable is selected. As an example consider a case where the defuzzified output has resulted in the situation described in **Table 2**. Note that each voter has its own membership function where the level of overlap is different for each voter. Based on a concrete input voting agent nr 1 could map the defuzzified variables into high, medium and low trust whereas voting agent 10 to only low trust.

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
L_t	L_t	L_t	L_t	L_t	L_t	L_t	L_t	L_t	L_t
M_t	M_t	M_t	M_t	M_t	M_t				
H_t	H_t	H_t							

Table 2. Possible values for voting

Note that behind each trust lever there is a real number, which represents the defuzzified value. These values are used to reduce the number of possible linguistic variables in order to obtain the vote for each voting agent. Each agent retains the linguistic variable that represents the highest value and is depicted in **Table 3**.

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10
H_t	M_t	L_t	L_t	M_t	M_t	L_t	L_t	L_t	L_t

Table 3. Voting

Taken as a function of x these probabilities form probability functions. They should therefore satisfy:

$$\sum_{w \in T(L)} P_r(L = w \mid x) = 1 \tag{14}$$

which gives a probability distribution on words:

$$\sum P_r(L = Low\_trust \mid x) = 0.6 \tag{15}$$
$$\sum P_r(L = Medium\_trust \mid x) = 0.3$$
$$\sum P_r(L = High\_trust \mid x) = 0.1$$

As a result of voting we can conclude that given the difference in belief $x=0.67$ the combination should not consider this belief in the similarity function since based on its difference compared to another beliefs it turns out to be a distrustful assessment. The before mentioned process is then repeated as many times as many different beliefs we have for the similarity i.e. as many as different similarity measures exist in the ontology mapping system.

7. Experimental analysis

Experimental comparison of ontology mapping systems is not a straightforward task as each system is usually designed to address a particular need from a specific domain. Authors have the freedom to hand pick some specific set of ontologies and demonstrate the strengths and weaknesses of their system carrying out some experiments with these ontologies. The problem is however that it is difficult to run the same experiments with another system and compare the two results. This problem has been acknowledged by the Ontology Mapping community and as a response to this need the Ontology Alignment Evaluation Initiative[7] has been set up in 2004. The evaluation was measured with recall, precision and F-Measure, which are useful measures that have a fixed range and meaningful from the mapping point of view.

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 follows:

$$\mathrm{Pr}ecision = \frac{\left|\{relevant\ items\} \cap \{retrieved\ items\}\right|}{\left|\{retrieved\ items\}\right|} \tag{16}$$

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 follows:

$$\mathrm{Re}call = \frac{\left|\{relevant\ items\} \cap \{retrieved\ items\}\right|}{\left|\{relevant\ items\}\right|} \tag{17}$$

[7] http://oaei.ontologymatching.org/

F-Measure: The weighted harmonic mean of precision and recall and is calculated as follows:

$$F - measure = \frac{2 * (\Pr ecision * \Re call)}{(\Pr ecision + \Re call)} \quad (18)$$

Recall is 100% when every relevant entity is retrieved. However it is possible to achieve 100% by simply returning every entity in the collection for every query. Therefore, recall by itself is not a good measure of the quality of a search engine. Precision is a measure of how well the engine performs in not returning non-relevant documents. Precision is 100% when every entity returned to the user is relevant to the query. There is no easy way to achieve 100% precision other than in the trivial case where no document is ever returned for any query. Both precision and recall has a fixed range: 0.0 to 1.0 (or 0% to 100%). A good mapping algorithm must have a high recall to be acceptable for most applications. The most important factor in building better mapping algorithms is to increase precision without worsening the recall. In order to compare our system with other solutions we have participated in the OAEI competitions since 2006. Each year we have been involved in more tracks than the previous year. This gave us the possibility to test our mapping system on different domains including medical, agriculture, scientific publications, web directories, food and agricultural products and multimedia descriptions. The experiments were carried out to assess the efficiency of the mapping algorithms themselves. The experiments of the question answering (AQUA) using our mappings algorithms are out of the scope of this paper. Our main objective was to compare our system and algorithms to existing approaches on the same basis and to allow drawing constructive conclusions.

7.1 Benchmarks

The OAEI benchmark contains tests, which were systematically generated starting from some reference ontology and discarding a number of information in order to evaluate how the algorithm behave when this information is lacking. The bibliographic reference ontology (different classifications of publications) contained 33 named classes, 24 object properties, 40 data properties. Further each generated ontology was aligned with the reference ontology. The benchmark tests were created and grouped by the following criteria:

- Group 1xx: simple tests such as comparing the reference ontology with itself, with another irrelevant ontology or the same ontology in its restriction to OWL-Lite
- Group 2xx: systematic tests that were obtained by discarding some features from some reference ontology e.g. name of entities replaced by random strings, synonyms, name with different conventions, strings in another language than English, comments that can be suppressed or translated in another language, hierarchy that can be suppressed, expanded or flattened. Further properties that can be suppressed or having the restrictions on classes discarded, and classes that can be expanded, i.e. replaced by several classes or flattened
- Group 3xx: four real-life ontologies of bibliographic references that were found on the web e.g. BibTeX/MIT, BibTeX/UMBC

Fig 6 shows the 6 best performing systems out of 13 participants. We have ordered the systems based on the their the F-Value of the H-means because the H-mean unifies all results for the test and F-Value represents both precision and recall.

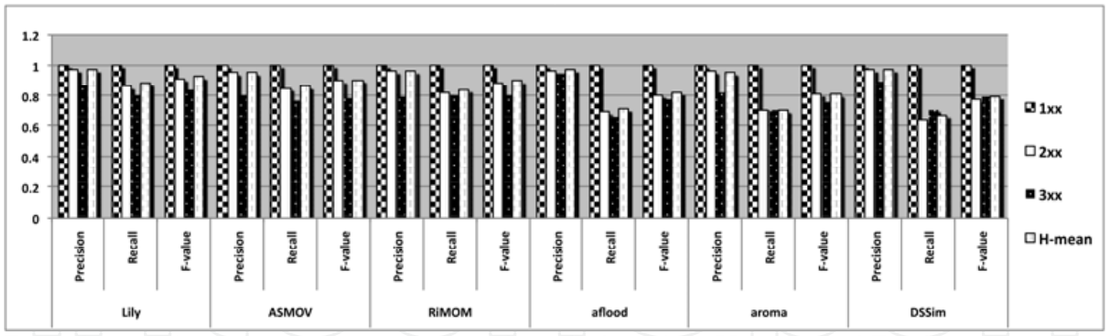

Fig 6. Best performing systems in the benchmarks based on H-mean and F-value

In the benchmark test we have performed in the upper mid range compared to other systems. Depending on the group of tests our system compares differently to other solutions:

- Group 1xx: Our results are nearly identical to the other systems.
- Group 2xx: For the tests where syntactic similarity can determine the mapping outcome our system is comparable to other systems. However where semantic similarity is the only way to provide mappings our systems provides less mappings compared to the other systems in the best six.
- Group 3xx: Considering the F-value for this group only 3 systems SAMBO, RIMOM and Lily are ahead.

The weakness of our system to provide good mappings when only semantic similarity can be exploited is the direct consequence of our mapping architecture. At the moment we are using four mapping agents where 3 carries our syntactic similarity comparisons and only 1 is specialised in semantics. However it is worth to note that our approach seems to be stable compared to our last year's performance, as our precision recall values were similar in spite of the fact that more and more difficult tests have been introduced in this year. As our architecture is easily expandable with adding more mapping agents it is possible to enhance our semantic mapping performance in the future.

7.2 Anatomy

The anatomy track (**Fig 7**) contains two reasonable sized real world ontologies. Both the Adult Mouse Anatomy (2.744 classes) and the NCI Thesaurus for Human Anatomy (3.304 classes) describes anatomical concepts. The classes are represented with standard *owl:Class* tags with proper rdfs:label tags. Besides their large size and a conceptualization that is only to a limited degree based on the use of natural language, they also differ from other ontologies with respect to the use of specific annotations and roles, e.g. the extensive use of the *partOf* relations, *owl:Restriction* and *oboInOwl: hasRelatedSynonym* tags. Our mapping algorithm has used the labels to establish syntactic similarity and has used the *rdfs:subClassOf* tags to establish semantic similarities between class hierarchies. For this track we did not use any medical background knowledge but the standard WordNet dictionary. Three systems SAMBO, SAMBOdtf and ASMOV have used domain specific background knowledge for this track.

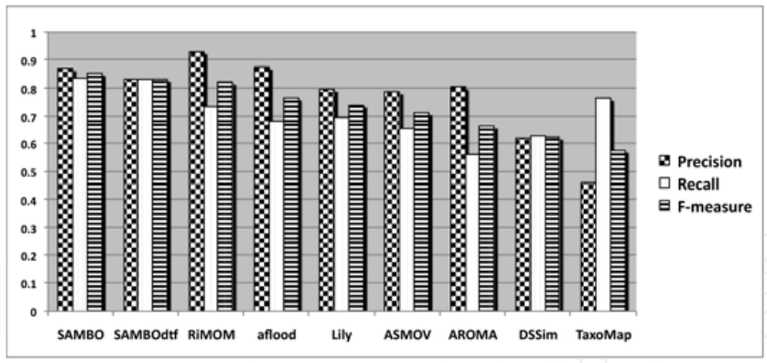

Fig 7. All participating systems in the anatomy track ordered by F-value

The anatomy track represented a number of challenges for our system. Firstly the real word medical ontologies contain classes like "outer renal medulla peritubular capillary", which cannot be easily interpreted without domain specific background knowledge. Secondly one ontology describes humans and the second describes mice. To find semantically correct mappings between them requires deep understanding of the domain. According to the results our system DSSim did not perform as we expected in this test compared to the other systems, as we do not use any domain specific background knowledge or heuristics. The best performing system was SAMBO, which has been designed specifically for the biomedical domain. In order to improve our performance we consider to experiment with medical background knowledge in the future.

7.3 Fao

The Food and Agricultural Organization of the United Nations (FAO) track contains one reasonable sized and two large real world ontologies.

1. The AGROVOC describes the terminology of all subject fields in agriculture, forestry, fisheries, food and related domains (e.g. environment). It contains around 2.500 classes. The classes itself are described with a numerical identifier through *rdf:ID* attributes. Each class has an instance, which holds labels in multiple languages describing the class. For establishing syntactic similarity we substitute the class label with its instance labels. Each instance contains a number of additional information like *aos:hasLexicalization* of *aos:hasTranslation* but we do not make use of it as it describes domain specific information.
2. ASFA contains 10.000 classes and it covers the world's literature on the science, technology, management, and conservation of marine, brackish water, and freshwater resources and environments, including their socio-economic and legal aspects. It contains only classes and its labels described by the standard *owl:Class* formalism.
3. The fisheries ontology covers the fishery domain and it contains a small number of classes and properties with around 12.000 instances. Its conceptual structure is different from the other two ontologies. These differences represented the major challenge for creating the alignments.

For the OAEI contest three sub tracks were defined as follows:

- agrafsa: create class to class mapping between the AGROVOC and ASFA ontologies
- agrorgbio: create class to class mappings between the AGROVOC organism module and fisheries biological entities where the fisheries instances are matched against classes.
- fishbio: create class to class mappings between fisheries commodities and biological entities where the instances are matched against classes

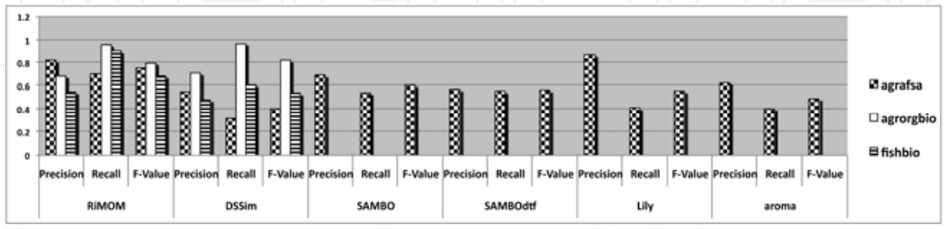

Fig. 8. All participating systems in the FAO track ordered by F-value

The FAO track (**Fig. 8**) was one of the most challenging ones as it contains three different sub tasks and large scale ontologies. As a result DSSim was one of the two systems, which could create complete mappings. The other systems have participated in only one sub task. In terms of overall F-Value RiMOM has performed better than DSSim. This can be contributed to the fact that the FAO ontologies contain all relevant information e.g. *rdfs:label*, *hasSynonym*, *hasLexicalisation* on the individual level and using them would imply implementing domain specific knowledge into our system. Our system has underperformed RiMOM because our individual mapping component is only part of our whole mapping strategy whereas RiMOM could choose the favour instance mapping over other strategies. However in the agrorgbio sub task DSSim outperformed RiMOM, which shows that our overall approach is comparable.

7.4 Directory

The purpose of this track was to evaluate performance of existing alignment tools in real world taxonomy integration scenario. Our aim is to show whether ontology alignment tools can effectively be applied to integration of "shallow ontologies". The evaluation dataset was extracted from Google, Yahoo and Looksmart web directories. The specific characteristics of the dataset are:

- More than 4500 of node matching tasks, where each node matching task is composed from the paths to root of the nodes in the web directories. Expert mappings for all the matching tasks.
- Simple relationships. Basically web directories contain only one type of relationship so called "classification relation".
- Vague terminology and modelling principles: The matching tasks incorporate the typical "real world" modelling and terminological errors.

These node matching tasks were represented by pairs of OWL ontologies, where classification relation is modelled as OWL *subClassOf* construct. Therefore all OWL

ontologies are taxonomies (i.e. they contain only classes (without Object and Data properties) connected with subclass relation.

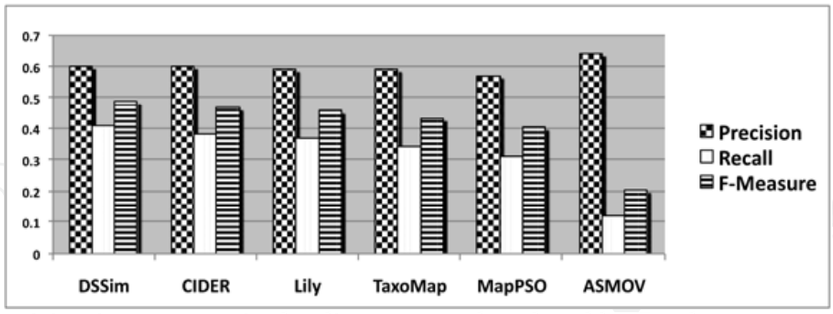

Fig 9. All participating systems in the directory track ordered by F-value

In the library track only 6 systems (**Fig 9**) have participated this year. In terms of F-value DSSim has performed the best however the difference is marginal compared to the CIDER (Gracia & Mena, 2008) or Lily systems. The concepts in the directory ontologies mostly can mostly be characterised as compound nouns e.g. "News_and_Media" and we need to process(split) them properly before consulting background knowledge in order to provide better mappings in the future.

7.5 Library

The objective of this track was to align two Dutch thesauri used to index books from two collections held by the National Library of the Netherlands.

Each collection is described according to its own indexing system and conceptual vocabulary. On the one hand, the Scientific Collection is described using the GTT, a huge vocabulary containing 35.000 general concepts ranging from "Wolkenkrabbers (Sky-scrapers)" to "Verzorging (Care)". On the other hand, the books contained in the Deposit Collection are mainly indexed against the Brinkman thesaurus, containing a large set of headings (more than 5.000) that are expected to serve as global subjects of books. Both thesauri have similar coverage (there are more than 2.000 concepts having exactly the same label) but differ in granularity. For each concept, the thesauri provide the usual lexical and semantic information: preferred labels, synonyms and notes, broader and related concepts, etc. The language of both thesauri is Dutch, but a quite substantial part of Brinkman concepts (around 60%) come with English labels. For the purpose of the alignment, the two thesauri have been represented according to the SKOS model, which provides with all these features.

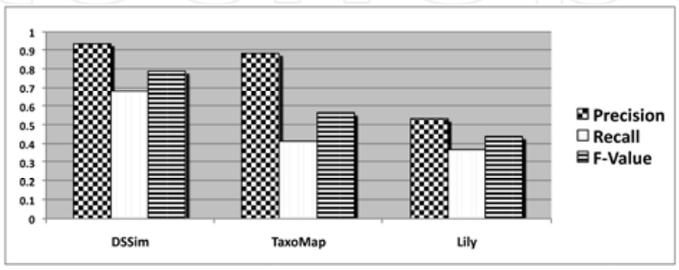

Fig. 10. All participating systems in the library track ordered by F-value

In the library track DSSim has performed the best (**Fig. 10**) out of the 3 participating systems. The track is difficult partly because of its relative large size and because of its multilingual representation. However these ontologies contain related and broader terms therefore the mapping can be carried out without consulting multi lingual background knowledge. This year the organisers have provided instances as separate ontology as well however we did not make use of it for creating our final mappings. For further improvements in recall and precision we will need to consider these additional instances in the future.

7.6 Very Large Cross-Lingual Resources

This vlcr track was the most complex this year. It contains 3 large ontologies. The GTAA thesaurus is a Dutch public audiovisual broadcast's archive, for indexing their documents, contains around 3.800 subject keywords, 97.000 persons, 27.000 names and 14.000 locations. The DBPedia is an extremely rich dataset. It contains 2.18 million resources or "things", each tied to an article in the English language Wikipedia. The "things" are described by titles and abstracts in English and often also in Dutch. We have converted the original format into standard SKOS in order to use it in our system. However we have converted only the labels in English and in Dutch whenever it was available. The third resource was the WordNet 2.0 in SKOS format where the synsets are instances rather than classes. In our system the WordNet 3.0 is included into as background knowledge therefore we have converted the original noun-synsets into a standard SKOS format and used our WordNet 3.0 as background knowledge. Unfortunately DSSim was the only system, which participated in this track therefore we cannot make qualitative comparisons. Nevertheless in this track our precision has ranged from 10% to 94% depending on the test and facet. The lowest precision 0.1 occurred on the GTAA-Wordnet mapping for the persons facet. This can be explained because the GTAA contains nearly hundred thousand persons, which does not have at all correspondence in WordNet. In fact WordNet contains very few persons. As the number of entities in these ontologies are very large only an estimation was can be calculated for the recall/coverage and for not all the facets. The estimated recall values for the evaluated samples were relatively low around 20%. For more advanced evaluation more test will be carried out in order to identify the strengths and weaknesses of our system.

8. Strengths and weaknesses of our solution

Based on the OAEI experiments, we can conclude that our solution compares and scales well to other well established ontology mapping systems. Nevertheless it is clear (OAEI seems to share our opinion) that it is not possible to clearly define a "winner" on these yearly competitions. Each system has its strengths and weaknesses and they tend to perform differently on different domains. However we can define some criteria to determine where we perform well and on which areas do we need to make further progress.

1. Domain independence: This is a definite strength of our system. Our solution does not rely on pre-defined thresholds or parameters that needs to be changed from domain to domain. Several mapping systems utilise machine learning in order to determine these parameters however these solutions are likely to be dependent on

the training set. DSSim uses WordNet as the background knowledge. This ensures that we can provide equivalent mappings on different domains. Nevertheless domain specific background knowledge can influence the results positively. The anatomy track has proved that systems that use domain specific background knowledge are far superior compared to the systems with general background knowledge. Nevertheless the drawback of these systems is that they cannot produce equally good results once the domain is changing. For example the AOAS system (Zhang & Bodenreide, 2007) performed the best on the anatomy track on the OAEI 2007 but they did not produce result in any other track as their system was fine tuned for the medical domain.

2. Conflict management: This area needs to be improved in our system. DSSim do manage conflicting beliefs over a particular mapping, which can occur when different agents have built up conflicting beliefs for the correctness of a mapping candidate. The problem occurs when we have already selected a mapping candidate and later on in the mapping process we add an another mapping that contradicts the previous one. Systems e.g. ASMOV, which try to detect conflicting mappings in the result-set can provide better overall results compared to our solution.
3. Mapping quality: DSSim does not produce always the best precision and recall for each track however our mapping quality is stable throughout different domains. We consider this as a strength of our system because we foresee different application domains where our solution can be used. In this context it is more important that we can produce equally good enough mappings.
4. Mapping performance: Due to our multi-agent architecture our solution scales well with medium and large domains alike. For example in the OAEI 2008 the largest ontologies were in the Very Large Cross-Lingual Resources track. DSSim was the only system that has participated in this track. Our solution can scale well for large domains because as the domain increases we can distribute the problem space between an increasing number of agents. Additionally our solution fits well to current hardware development trends, which predicts an increasing number of processor core in order to increase the computing power.
5. Traceability of the reasoning: Unfortunately this is a weakness of our system as we cannot guarantee that running the algorithm twice on the same domain we will always get exactly the same results. The reason is that our belief conflict resolution approach (Nagy et al., 2008) uses fuzzy voting for resolving belief conflicts which can vary from case to case. Additionally beliefs are based on similarities between a set of source and target variables. The set of variables are deducted from the background knowledge, which can differ depending on the actual context of our query. Therefore it is not feasible to trace exactly why a particular mapping has been selected as good mapping compared to another candidate mappings.

9. Conclusions

In this paper we have investigated a combination of 3 challenges that we think is crucial to address in order to provide an integrated ontology mapping solution. We have provided

our solution DSSim, which is the core ontology mapping component for our proposed architecture that integrates with question answering at the moment. However our system is easily expandable, layered with clear interfaces, which allows us to integrate our solution into different context like Semantic Web Services (Vargas-Vera et al., 2009). Further in this paper we have shown how the fuzzy voting model can be used to resolve contradictory beliefs before combining them into a more coherent state by evaluating fuzzy trust.
We have proposed new levels of trust for resolving these conflicts in the context of ontology mapping, which is a prerequisite for any systems that makes use of information available on the Semantic Web. Our system is conceived to be flexible because the membership functions for the voters can be changed dynamically in order to influence the outputs according to the different similarity measures that can be used in the mapping system. We have described initial experimental results with the benchmarks of the Ontology Alignment Initiative, which demonstrates the effectiveness of our approach through the improved recall and precision rates. There are many areas of ongoing work, with our primary focus considering the effect of the changing number of voters and the impact on precision and recall or applying our algorithm in different application areas. We continuously evaluate the performance of our system through OAEI competitions (Nagy et al., 2006) (Nagy et al., 2007) (Nagy et al., 2008) that allows us to improve, evaluate and validate our solution compared to other state of the art systems. So far our qualitative results are encouraging therefore we aim to investigate further the belief combination optimisation, compound noun processing and agent communication strategies for uncertain reasoning in the future.

10. References

Austen-Smith, D.; Banks, J.S. (1996). Information Aggregation, Rationality, and the Condorcet Jury Theorem, pp-34-45, The American Political Science Review

Baldwin, J.F. (1999). Mass assignment Fundamentals for computing with words, pp 22-44, Springer-Verlag, vol 1566

Batini, C.; Lenzerini, M.; Navathe, S.B. (1986). A comparative analysis of methodologies for database schema integration, pp 323-364, ACM Computing Surveys

Beckett, D. (2004). RDF/XML Syntax Specification, http://www.w3.org/RDF/

Bock, J.; Hettenhausen, J. (2008). MapPSO Results for OAEI 2008, *Proceedings of the 3rd International Workshop on Ontology Matching,* Germany, Karlsruhe

Brugman, H.; Malaisé, V.; Gazendam, L. (2006). A Web Based General Thesaurus Browser to Support Indexing of Television and Radio Programs, *Proceedings of the 5th international conference on Language Resources and Evaluation (LREC 2006)* , Italy Genoa

Carlo, W.; Tharam, D.; Wenny, R.; Elizabeth, C. (2005). Large scale ontology visualisation using ontology extraction, pp 113-135, International Journal of Web and Grid Services, vol 1

Caracciolo, C.; Euzenat, J.; Hollink, L.; Ichise, R. (2008). First results of the Ontology Alignment Evaluation Initiative 2008, *Proceedings of the 3rd International Workshop on* Ontology Matching, Germany, Karlsruhe

Cohen, W.; Ravikumar, P.; Fienberg, S.E. (2003). A comparison of string distance metrics for name-matching tasks, *Proceedings of Information Integration on the Web (IIWeb 2003)*, Indonesia, Jakarta

Ding, L.; Finin, T.; Joshi, A. (2004). Swoogle: A Search and Metadata Engine for the Semantic Web, *Proceedings of the Thirteenth ACM Conference on Information and Knowledge Management,* USA, Washington DC

Euzenat, J.; Shvaiko, P. (2007). Ontology matching, Springer-Verlag, 3-540-49611-4, Heidelberg, Germany

Flahive, A.; Apduhan, B.O.; Rahayu, J.W.; Taniar, D. (2006). Large scale ontology tailoring and simulation in the Semantic Grid Environment, pp 256-281, International Journal of Metadata, Semantics and Ontologies, vol 1

Gracia, J.; Mena, E. (2008). Ontology Matching with CIDER:Evaluation Report for the OAEI 2008, *Proceedings of the 3rd International Workshop on Ontology Matching*, Germany, Karlsruhe

Hamdi, F.; Zargayouna, H.; Safar, B.; Reynaud, C. (2008). TaxoMap in the OAEI 2008 alignment contest , *Proceedings of the 3rd International Workshop on Ontology Matching*, Karlsruhe, germany

Jean-Mary Y.R; Kabuka, M.R. (2007). ASMOV: Ontology Alignment with Semantic Validation, *Proceedings of Joint SWDB-ODBIS Workshop*, Austria, Vienna

Ji, Q. ; Haase, P. ;Qi, G. (2008). Combination of Similarity Measures in Ontology Matching by OWA Operator, *Proceedings of the 12th International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'08)*, Spain, Malaga

Lambrix, P.; Tan, H. (2006). SAMBO - a system for aligning and merging biomedical ontologies, pp 196-206, Journal of Web Semantics, Special issue on Semantic Web for the Life Sciences, vol 4

Lawry, J. (1998). A voting mechanism for fuzzy logic, pp 315-333, International Journal of Approximate Reasoning, vol 19

Lenzerini, M.; Milano, D.; Poggi, A. (2004). Ontology representation and reasoning, NoE InterOp (IST-508011), WP8, subtask 8.2

McGuinness, D.L; Harmelen, F. (2004). OWL Web Ontology Language, http://www.w3.org/TR/owl-features/

Miles, A.; Bechhofer, S. (2008). SKOS Simple Knowledge Organization System, http://www.w3.org/TR/skos-reference/

Monge, A.E.; Elkan, C.P. (1996). The field matching problem: Algorithms and applications, *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, USA, Portland

Nagy, M. ; Vargas-Vera, M. ; Motta, E. (2006). DSSim-ontology Mapping with Uncertainty, *Proceedings of the 1st International Workshop on Ontology Matching*, USA, Georgia

Nagy, M. ; Vargas-Vera, M. ; Motta, E. (2005). Multi Agent Ontology Mapping Framework in the AQUA Question Answering System, pp 70-79, *Proceedings of MICAI 2005 : Advances in Artificial Intelligence, 4th Mexican International Conference on Artificial Intelligence*, Mexico, Monterrey

Nagy, M. ; Vargas-Vera, M. ; Motta, E. (2007). DSSim - managing uncertainty on the semantic web, *Proceedings of the 2nd International Workshop on Ontology Matching,* South Korea, Busan

Nagy, M. ; Vargas-Vera, M. ; Stolarski, P. (2008). DSSim results for the OAEI 2008, *Proceedings of the 3rd International Workshop on Ontology Matching,* Germany, Karlsruhe

Nagy, M. ; Vargas-Vera, M. ; Motta, E. (2008). Introducing Fuzzy Trust for Managing Belief Conflict over Semantic Web Data, *Proceedings of 4th International Workshop on Uncertain Reasoning on the Semantic Web*, Germany, Karlsruhe

Sentz, K.; Ferson, S. (2002). Combination of Evidence in Dempster-Shafer Theory, Systems Science and Industrial Engineering Department, Binghamton University

Schmid, H. (1994). Probabilistic part-of-speech tagging using decision trees,pp 25-36, *Proceedings of the International Conference on New Methods in Language Processing*, UK, Manchester

Shafer, G. (1976). A Mathematical Theory of Evidence, 978-0691081755, Princeton University Press

Tang, J.;Li, J.; Liang, B.; Li, Y. (2006). Using Bayesian decision for ontology mapping, pp 243-262, Web Semantics: Science, Services and Agents on the World Wide Web, vol 4

Vargas-Vera, M. ; Motta, E. (2004). AQUA - Ontology-based Question Answering System, *Proceedings of the 3rd International Mexican Conference on Artificial Intelligence (MICAI-2004*), Mexico, Monterrey

Vargas-Vera, M. ; Motta, E. (2004). An Ontology-driven Similarity Algorithm, KMI-TR-151, Knowledge Media Institute, The Open University, UK, Milton Keynes

Vargas-Vera, M.;Nagy, M.; Zyskowski, D.; Haniewicz, K.;Abramowicz, W.; Kaczmarek, M. (2009). Challenges on Semantic Web Services, 978-1-60566-650-1, Handbook of Research on Social Dimensions of Semantic Technologies and Web Services, Information Science Reference

Wand, Y.; Wang, R.Y. (1996). Anchoring data quality dimensions in ontological foundations, pp 86-95, Communications of the ACM, vol 39

Wang, P.; Xu, B. (2008). Lily: Ontology Alignment Results for OAEI 2008, *Proceedings of the 3rd International Workshop on Ontology Matching*, Germany, Karlsruhe

Wang, R.Y.;Kon, H.B.; Madnick, S.E. (1993). Data Quality Requirements Analysis and Modeling, *Proceedings of the 9th International Conference on Data Engineering,* Austria, Vienna

Zhang, S.; Bodenreide, O. (2007). Hybrid Alignment Strategy for Anatomical Ontologies Results of the 2007 Ontology Alignment Contest, Proceedings of the 2nd International Workshop on Ontology Matching, South Korea, Busan

Young, H.P. (1988). Condorcet's Theory of Voting, pp 1231-1244, The American Political Science Review, vol 82

Semantic Web
Edited by Gang Wu

ISBN 978-953-7619-54-1
Hard cover, 310 pages
Publisher InTech
Published online 01, January, 2010
Published in print edition January, 2010

Having lived with the World Wide Web for twenty years, surfing the Web becomes a way of our life that cannot be separated. From latest news, photo sharing, social activities, to research collaborations and even commercial activities and government affairs, almost all kinds of information are available and processible via the Web. While people are appreciating the great invention, the father of the Web, Sir Tim Berners-Lee, has started the plan for the next generation of the Web, the Semantic Web. Unlike the Web that was originally designed for reading, the Semantic Web aims at a more intelligent Web severing machines as well as people. The idea behind it is simple: machines can automatically process or “understand” the information, if explicit meanings are given to it. In this way, it facilitates sharing and reuse of data across applications, enterprises, and communities. According to the organisation of the book, the intended readers may come from two groups, i.e. those whose interests include Semantic Web and want to catch on the state-of-the-art research progress in this field; and those who urgently need or just intend to seek help from the Semantic Web. In this sense, readers are not limited to the computer science. Everyone is welcome to find their possible intersection of the Semantic Web.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Miklos Nagy and Maria Vargas-Vera (2010). Towards an Automatic Semantic Data Integration: Multi-agent Framework Approach, Semantic Web, Gang Wu (Ed.), ISBN: 978-953-7619-54-1, InTech, Available from: http://www.intechopen.com/books/semantic-web/towards-an-automatic-semantic-data-integration-multi-agent-framework-approach

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen