

# DSSim Results for OAEI 2008

Miklos Nagy<sup>1</sup>, Maria Vargas-Vera<sup>1</sup>, and Piotr Stolarski<sup>2</sup>

<sup>1</sup> The Open University

Walton Hall, Milton Keynes, MK7 6AA, United Kingdom

`mn2336@student.open.ac.uk`; `m.vargas-vera@open.ac.uk`

<sup>2</sup> Poznan University of Economics

al. Niepodleglosci 10, 60-967 Poznan, Poland

`P.Stolarski@kie.ae.poznan.pl`

**Abstract.** The growing importance of ontology mapping on the Semantic Web has highlighted the need to manage the uncertain nature of interpreting semantic meta data represented by heterogeneous ontologies. Considering this uncertainty one can potentially improve the ontology mapping precision, which can lead to better acceptance of systems that operate in this environment. Further the application of different techniques like computational linguistics or belief conflict resolution that can contribute the development of better mapping algorithms are required in order to process the incomplete and inconsistent information used and produced during any mapping algorithm. In this paper we introduce our algorithm called “DSSim” and describe the improvements that we have made compared to OAEI 2006 and OAEI 2007.

## 1 Presentation of the system

### 1.1 State, purpose, general statement

Ontology mapping systems need to interpret heterogeneous data in order to simulate “machine intelligence”, which is a driving force behind the Semantic Web. 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. In practice there are several roadblocks[1] that hamper the development of mapping solutions that perform equally well for different domains. Additionally the different combination of these challenges needs to be addressed in order to design systems that provides good quality results. DSSim has been designed to address the combination of the 3 following challenges:

- Representation and interpretation problems: Ontology designers have a wide variety of languages and language variants to choose from in order to represent their domain knowledge. The most widely used for small and medium sized ontologies are RDF(S) and OWL as Web ontology language however OWL has three increasingly-expressive sublanguages(OWL Lite, OWL DL, OWL Full) with different expressiveness and language constructs. Other languages like SKOS, which is a standard to support the use of knowledge organization systems (KOS) such

as thesauri, classification schemes, subject heading systems and large scale taxonomies within the framework of the Semantic Web. From the logical representation point of view each representations are valid separately and no logical reasoner would find inconsistency in them individually. However the problem occurs once we need to compare ontologies with different representations in order to determine the similarities between classes and individuals. Consider for example one ontology where the labels are described with standard class *rdfs:label* tag and an another ontology where the same is described as *hasNameScientific* data property. 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.

- Quality of the Semantic Web data: 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. 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.
- Efficient mapping with large scale ontologies: Ontologies can get quite complex and very large, causing difficulties in using them for any application. 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 became 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. This year the OAEI tracks have also included a task very large cross lingual ontologies, which includes establishing mappings between Wordnet, DBPedia an GTAA(Dutch acronym for Common Thesaurus for Audiovisual Archives), which is a domain specific thesaurus with approximately 160.000 terms. A lot of researcher 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.

As a result from the mapping point of view ontologies will always contain inconsistencies, missing or overlapping elements and different conceptualisation of the same terms, which introduces a considerable amount of uncertainty into the mapping process. In order to represent and reason with this uncertainty authors (Vargas-Vera and Nagy) have proposed a multi agent ontology mapping framework [2], which uses the Dempster-Shafer [3] theory in the context of Question Answering. Since our first proposition[4] of such solution in 2005 we have gradually developed and investigated multiple components of such system and participated in the OAEI in order to validate the feasibility of our proposed solution. Fortunately during the recent years our original concept has received attention from other researchers [5, 6], which helps to broaden the general knowledge on this area. We have investigated different aspects of our original idea namely the feasibility of belief combination[7] and the resolution of conflicting beliefs [8] over the belief in the correctness of similarities using the fuzzy voting model. A comprehensive description of the Fuzzy voting model can be found [8]. For this contest (OAEI 2008) the benchmarks, anatomy, fao, directory, mldirectory, library and vlr tracks had been tested with this new version of DSSim (v0.3). Therefore, we had improved our precision and recall measures. Furthermore, experiments(based on the benchmarks) reported in [8] showed that average recall can be improved up to 12% and average recall up to 16%. These new improvements have been included into our DSSim v0.3 system and been tested through OM-2008.

## 1.2 Specific techniques used

This year we introduced also two types of improvements. Those enhancements are mainly connected to multiword ontology entity labels and include: compound nouns comparisons with the use of semantic relations technique as well as extensive production of abbreviations based on defined language rules. The realization of the first improvement comes from the inspiration of researches on computational linguistics, whereas the second advancement is produced on the basis of pragmatic observations of exemplary unmatched alignments from the conference track.

A fundamental case which has led us to the idea of introducing the abbreviations factory - a component responsible for production of expected possible shortenings of words or phrases - came from the Conference track. The available linguistic resources (i.e. Wordnet) provide indeed a very extensive aid in dealing with different sorts of language processing tasks. Nevertheless, those resources are not ideal. As a result the mentioned Wordnet, for instance, does not offer any service for obtaining any list of shortened forms for a word or phrase. Though it may seem less important in the task of ontology matching we may consider a straight-ahead example invalidating such a view.

In some conference-track ontologies there were entities (mostly classes) denoting the concept of "Program Committee" (or "Program Committee Member"). Of course any human being with a little acquaintance of the domain would know that the phrase is commonly abbreviated to "PC" (often encountered in the ontologies). Unfortunately, such knowledge comes rather from the experience and cannot be expected to be ad-hoc part of a computer system. Another important observation is that some specific abbreviations are typical only for those specific domains and can reflect even completely other

phrases in a common language. For instance the “PC” phrase would rather be interpreted as a shortening for “Personal Computer”. In fact only few on-line abbreviation dictionaries return the sense of “Program Committee”, which hindered us from using the external resource on the favor of trying a (simpler for implementation) rule-based shortenings generator.

The compound nouns comparison method is an interesting example of algorithm dealing with interpretation of compound nouns based on earlier works done in such fields as language understanding as well as question-answering and machine translation. The problem of establishing the semantic relations between items of compound nouns has awaited many different approaches [9] [10] [11]. Yet, all of them should be regarded as partial solutions rather than a definite one. Most of the cases uses either manually created rules [9] or machine learning techniques [10] in order to automatically build classification rules that will enable to rate any given compound noun phrase into one of a set of pre-selected semantic relations which best reflects the sense and nature of that phrase. As mentioned, most approaches are not comprehensive and their authors limit their resolutions to some specific restrictions. For instance the most often case that is being scrutinized is the binary type of compound nouns, where the compound phrase is made up of only two nouns (a head and modifier).

In the context of ontology matching, the class of compound nouns semantic relation detection algorithms may be used in order to determine such relations within ontology entities’ identifiers and labels. After the relation  $r^{1,n}$  has been classified independently for entities in the first of aligned ontologies  $O^1$  and  $r^{2,m}$  separately for entities from the other ontology  $O^2$ , the alignments may be produced between the entities from  $O^1$  and  $O^2$  on the basis of similarity between the relations  $r^{1,n}$  and  $r^{2,m}$  itself. Such approach has its disadvantages but those can be in large part eliminated by introducing the algorithm into more general matching framework. For instance it fits especially well into the aligning system implemented by DSSim (described in details in [2]). As the number of elements in the set of isolated semantic relations is usually limited only to very general ones, the probability of detecting the same or similar relations is subjectively high, therefore the method itself is rather sensitive to the size of the set. Yet even if that number is relatively small the method may still be helpful if the outcomes are combined with other ways of similarity assessment. In the case of DSSim it means that the method can be treated as one of the experts.

Our implementation is, so far, a vastly simplified one. In the research we initially propose a small set of manually created rules, which employ different entries describing the ontology entity (comments are most favourable). This means that for our purposes we adopted some parts of method given in [9] but the way the rules are created rather moves our approach next to [10]. The method delimits us to processing of binary compounds, but it is potentially possible to change this. We started the experiments with a general set of semantic relations but recognized the need to switch to domain specific - depending on the type and field of representation of ontologies. Such creation of relations’ set is another challenge. We also use only two-state logic in expressing similarity function between the classified semantic relations.

We will consider some simple examples describing in more details the practical aspects of our implementation. Table 1 represent exemplary general relations that we

Relation type	Rule
CAUSE	“make”
EFFECT	“result effect”
LOCATION_FROM	“originates at starts from”
AGENT	“perform performing”
INSTRUMENT	“use employ”
POSSESSOR	“has have possess own owns”
PRODUCT	“produce create”
EQUATIVE	“is also equal”
PROPERTY	“is”

**Table 1.** A model way of defining compound nouns semantic relations and classification rules.

defined (on the basis of our analysis and cited paper [9]). The created rules are simple using the formalism of regular expressions and can mainly be based on the recognition of keywords in the additional descriptions (comments) of the compound phrases. Some example keywords are also presented in the table.

Now let us take into account the case of earlier mentioned “Program Committee Member”. Assuming that the ontology entity of this name is also accompanied with the comment: “[...] person that is performing the tasks on the account of the Program Committee [...]” will result in triggering the conclusion that this is the instance of the AGENT type of semantic relation once the rules are run against the comment<sup>3</sup>.

If during processing of the other ontology in the matching task another entity of the same meta-class (i.e. a concept, relation, etc.) will be detected and it will also be assigned the same semantic relation category (AGENT) then the binary nature of the comparison will deem those entities to be equal. It should be noted that so far any other relation within our implementation cannot be considered “similar”. Thus, entities with other relations like EFFECT or POSSESSOR are simply regarded as not the same. Nevertheless as stressed above, it is vital to take into account that having two compound nouns entities recognized with the same category does not mean that those entities really refer to the same meaning. So such fact should be viewed rather as a premise, at least in the case of general semantic relations categories<sup>4</sup>.

Summing up the introduced improvements, the compound nouns comparisons with the use of semantic relations technique is a very promising method that should in the future seriously improve the outcome of our algorithm. Nevertheless, as it is still in a rather premature phase the results are not according to expectations, yet. The method is however under severe development and thus we expect major impact on results in the future versions. It would also be interesting to introduce machine-learning techniques for rules acquisition instead of use only manually implemented ones - for instance Artificial Neural Networks can be used for this task.

<sup>3</sup> Because of the existence of the “performing” keyword.

<sup>4</sup> The level of certainty is higher when the ontologies are connected to a narrower domain or if the more specific categories are introduced.

### 1.3 Adaptations made for the evaluation

Our ontology mapping system is based on a multi agent architecture where each agent built up a belief for the correctness of a particular mapping hypothesis. Their beliefs are then combined into a more coherent view in order to provide better mappings. Although for the previous OAEI contests we have re-implemented our similarity algorithm as a standalone mapping process which integrates with the alignment api, we have recognised the need for possible parallel processing for tracks which contain large ontologies e.g. very large cross-lingual resources track. This need is indeed coincide with our original idea of using distributed multi-agent architecture, which is required for scalability purposes once the size of the ontology is increasing. Our modified mapping process can utilise multi core processors by splitting up the large ontologies into smaller fragments. Both the fragment size and the number of cores that should be used for processing can be set in the “param.xml” file.

Based on the previous implementation we have modified our process for the OAEI 2008 which works as follows:

1. Based on the initial parameters divide the large ontologies into  $n*m$  fragments.
2. Parse the ontology fragments and submit them into the alignment job queue.
3. Run the job scheduler as long as we have jobs in the queue and assign jobs into idle processor cores.
  - 3.1 We take a concept or property from ontology 1 and consider (refer to it from now) it as the query fragment that would normally be posed by a user. Our algorithm consults WordNet in order to augment the query concepts and properties with their hypernyms.
  - 3.2 We take syntactically similar concepts and properties to the query graph from ontology 2 and build a local ontology graph that contains both concepts and properties together with the close context of the local ontology fragments.
  - 3.3 Different similarity and semantic similarity algorithms (considered as different experts in evidence theory) are used to assess quantitative similarity values (converted into belief mass function) between the nodes of the query and ontology fragment which is considered as an uncertain and subjective assessment.
  - 3.4 Then the similarity matrixes are used to determine belief mass functions which are combined using the Dempster’s rule of combination. Based on the combined evidences we select those mappings in which we calculate the highest belief function.
4. The selected mappings are added into the alignment.

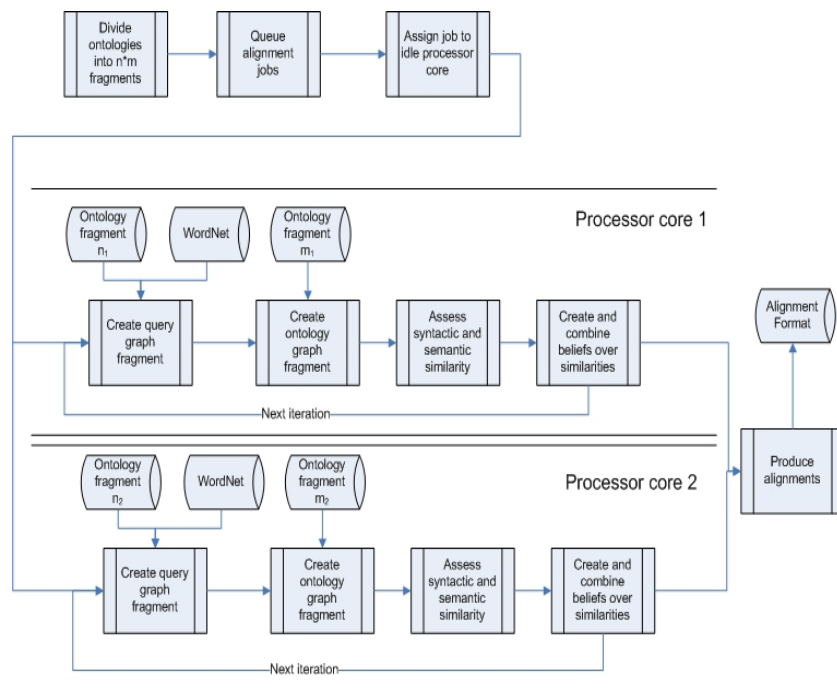
The overview of the mapping process is depicted on figure 1.

### 1.4 Link to the system and parameters file

<http://kmi.open.ac.uk/people/miklos/OAEI2008/tools/DSSim.zip>

### 1.5 Link to the set of provided alignments (in align format)

<http://kmi.open.ac.uk/people/miklos/OAEI2008/results/DSSim.zip>



**Fig. 1.** The mapping process on a dual-core processor

## **2 Results**

### **2.1 benchmark**

The benchmarks have been extended with new tests this year, which allows a more fine-grained evaluation of the results. It is definitely more difficult than last contest (2007). However our algorithm has produced the same results as last year. If we do not consider the new tests we have improved the recall with keeping the same precision. 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 years 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. The overall conclusion is that our system produces stable quality mappings, which is good however we still see room for improvements.

### **2.2 anatomy**

The anatomy track contains two reasonable sized real world ontologies. Both the Adult Mouse Anatomy (2.744 classes) and the NCI Thesaurus (3.304 classes) describes anatomical concepts. The classes are represented with standard owl:Class tags with proper rdfs:label 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. We could not make use of the owl:Restriction and oboInOwl:has-RelatedSynonym tags as this would require ontology specific additions. 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 well as we have expected in this test compared to the best system (SAMBO) because we do not use any domain specific background knowledge or heuristics but the standard WordNet dictionary. The run time per test was around 30 min, which is an improvement compared to last year.

### **2.3 fao**

The fao track contains one reasonable sized and two large ontologies. 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` or `aos:hasTranslation` but we do not make use of it as it describes domain specific information. 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. 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. The FAO track 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. The total execution time was around 10 hours.

## 2.4 directory

In the library track only 6 systems have participated this year. In terms of F-value DSSim has performed the best however the difference is marginal compared to the CIDER or Lily systems. The directory test as well has been manageable in terms of execution time. In general the large number of small-scale ontologies made it possible to verify some mappings for some cases. The tests contain only classes without any labels but in some cases different classes have been combined into one class e.g. "News\_and\_Media" that introduces certain level of complexity for determining synonyms using any background knowledge. To address these difficulties we have used a compound noun algorithms described in section 1.2. The execution time was around 15 minutes.

## 2.5 mldirectory

This track contains ontologies from five domains namely automobile, movie, outdoor, photo and software in both English and Japanese. They contain class descriptions in OWL format and RDF descriptions for the instances with labels and comments. We have produced only the English-English class alignments using the instance labels for the classes where possible. There were no reference alignments for this track and no expert evaluation were carried out for the results. The evaluators have compared the systems based on how many alignments the systems produced and what was overlap between the provided results. In the english-english alignment only 4 systems have participated from which only three (including DSSim) has run all data sets. Based on the

total alignment provided DSSim achieved the third place (around 30 % less mappings compared to RiMOM but only 5% less than Lily). The most surprising concerning the results is that there were no mappings, which were provided by all 4 systems at any of the data sets. From the performance point of view the run time for this track was around 8 hours.

## **2.6 library**

The library track contains two SKOS describing scientific collections (GTT) which is a huge vocabulary containing 35.000 general concepts and the Brinkman thesaurus, containing a large set of headings with more than 5.000 descriptions. Additionally not all labels were available in English therefore we have used the original Dutch labels. The implication is that we could not determine hypernyms from WordNet, which might impact our mapping precision negatively. We have participated in this track last year as well and although we did not add Dutch background knowledge to our algorithm we expect improvements compared to last year. The track is difficult partly because of its relative large size and because of its multilingual representation. Nevertheless in the library track DSSim has performed the best 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. This year the run time was around 12 hours.

## **2.7 vlc**

This vlc track was the most complex this year and DSSim was the only system that have participated in this track. It contains 3 large ontologies. The GTAA thesaurus is a Dutch public audiovisual broadcasts 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. 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 tests will be needed in order to identify the strengths and weaknesses of our system. The run time of the track was over 2 weeks and we could not run the complete GTAA-DBPedia combination due to the lack of time.

## **2.8 conferences**

This test set is made up of collection of 15 real-case ontologies dealing with the domain of conference organization. Although all the ontologies are well embedded in the described field, nevertheless they are heterogeneous in their nature. This heterogeneity comes mainly from: designed ontology application type, ontology expressivity in terms of formalism, and robustness. Out of given 15 ontologies the production of alignments should result in 210 possible combinations (we treat the equivalent alignment as symmetric). However, we obtained 91 non-empty alignment files in the generation. DSSim was one of two participants, which provided the maximum 105 alignments this year. The results were evaluated based on different methods e.g. sample and approximate or reference alignments. Fortunately this year a new reference alignment has been produced, which contains all possible pairs of five ontologies. Three confidence threshold was used for the evaluation(0.2, 0.5, 0.7) where the given threshold was used to filter results with the given threshold. Based on the F-measure our system performed differently considering the given threshold values. With threshold 0.2 DSSim is on the third position out of the three participating systems where the difference between systems is marginal. The position changes as the threshold increases. Using 0.5 as threshold DSSim moves to the first position while maintaining a marginal difference compared to the second place. The situation changes considerably using the 0.7 threshold. The difference between DSSim and the other systems increases considerably(DSSim 42 % compared to Lily 15% and ASMOV 11 %). From the performance point of view the alignments took about 1.5 hour on a rather slow computer<sup>5</sup>. After the reviewing stage of the results we came to the conclusions that good results generation is challenging for all ontology pairs in this track.

## **3 General comments**

### **3.1 Discussions on the way to improve the proposed system**

We have experienced that developing ontology specific functionality into the mapping system could considerably improve the quality of the mappings. For example using aos:hasTranslation tag in the fao track can provide additional information for assessing similarities. However these solutions will only work for the specific ontologies only which contradicts with our objective to provide a good mapping system independent on the domain. From the background knowledge point of view we have concluded that based on the latest results that the additional multi lingual and domains specific background knowledge could provide added value for improving both recall and precision of the system.

---

<sup>5</sup> Pentium III 750 MHz, 512 MB

### **3.2 Comments on the OAEI 2008 procedure**

The OAEI procedure and the provided alignment api works very well out of the box for the benchmarks, anatomy, directory, mldirectory and conference tracks. However for the fao, vlcr and library track we had to develop an SKOS parser, which can be integrated into the alignment api. Our SKOS parser convert SKOS file to OWL, which is then processed using the alignment api. Additionally we have developed a multi threaded chunk SKOS parser which can process SKOS file iteratively in chunks avoiding memory problems. For the vlcr track we had to develop several conversion and merging utility as the original file formats were not easily processable.

### **3.3 Comments on the OAEI 2008 test cases**

We have found that most of the benchmark tests can be used effectively to test various aspects of an ontology mapping system since it provides both real word and generated/modified ontologies. The ontologies in the benchmark are conceived in a way that allows anyone to clearly identify system strengths and weaknesses which is an important advantage when future improvements have to be identified. The anatomy, library and mldirectory tests are perfect to verify the additional domain specific or multi lingual domain knowledge. Unfortunately this year we could not integrate our system with such background knowledge so the results are not as good as we expected.

## **4 Conclusion**

Based on the experiments gained during OAEI 2006, 2007 and 2008 we had a possibility to realise a measurable evolution in our ontology mapping algorithm and test it with 8 different mapping tracks. Our main objective is to improve the mapping precision with managing the inherent uncertainty of any mapping process and information in the different ontologies. The different formalisms of the ontologies suggest that on the Semantic Web there is a need to qualitatively compare and evaluate the different mapping algorithms. Participating in the Ontology Alignment Evaluation Initiative is an excellent opportunity to test and compare our system with other solutions and helped a great deal identifying the future possibilities that needs to be investigated further. Further DSSim team was invited for oral presentation to the Ontology Mapping Workshop 2008 (OM-2008). An extract from the organizers is as follows: “Based on the discussion among the OAEI organisers and taking into account the number of tracks addressed and quality of matching results, it has been resolved that only the DSSim and ASMOV teams are offered to make oral presentations concerning their evaluation results”.

## **References**

1. Shvaiko, P., Euzenat, J.: Ten challenges for ontology matching. Technical Report DISI-08-042, University of Trento (2008)
2. Nagy, M., Vargas-Vera, M., Motta, E.: Dssim - managing uncertainty on the semantic web. In: Proceedings of the 2nd International Workshop on Ontology Matching. (2007)

3. Shafer, G.: A Mathematical Theory of Evidence. (1976)
4. Nagy, M., Vargas-Vera, M., Motta, E.: Multi agent ontology mapping framework in the aqua question answering system. In: International Mexican Conference on Artificial Intelligence (MICAI-2005). (2005)
5. Besana, P.: A framework for combining ontology and schema matchers with dempster-shafer (poster). In: Proceedings of the International Workshop on Ontology Matching. (2006)
6. Yaghlane, B.B., Laamari, N.: Owl-cm: Owl combining matcher based on belief functions theory. In: Proceedings of the 2nd International Workshop on Ontology Matching. (2007)
7. Nagy, M., Vargas-Vera, M., Motta, E.: Feasible uncertain reasoning for multi agent ontology mapping. In: IADIS International Conference-Informatics 2008. (2008)
8. Nagy, M., Vargas-Vera, M., Motta, E.: Managing conflicting beliefs with fuzzy trust on the semantic web. In: The 7th Mexican International Conference on Artificial Intelligence (MICAI 2008). (2008)
9. Turney, P.D.: Similarity of semantic relations. *Computational Linguistics* **32**(3) (2006) 379–416
10. Kim, S.N., Baldwin, T.: Interpreting semantic relations in noun compounds via verb semantics. In: Proceedings of the COLING/ACL on Main conference poster sessions. (2006) 491–498
11. Banerjee, S., Pedersen, T.: Extended gloss overlaps as a measure of semantic relatedness. In: Proceedings of the Eighteenth International Joint Conference on Artificial Intelligence. (2003) 805–810