ASMOV Results for OAEI 2007

Yves R. Jean-Mary[1], Mansur R. Kabuka [1,2]

[1] INFOTECH Soft, 9200 South Dadeland Blvd, Suite 620, Miami, Florida, USA 33156
[2] University of Miami, Coral Gables, Florida, USA 33124
{reggie, kabuka}@infotechsoft.com

Abstract. Numerous ontology alignment algorithms have appeared in the literature in recent years, but only a few make use of the semantics enclosed within the ontologies in order to improve the accuracy. In this paper, we present the Automated Semantic Mapping of Ontologies with Validation (ASMOV) algorithm for ontology alignment. We first provide a brief overview of the algorithm followed by an analysis of its results on the 2007 Ontology Alignment Evaluation Initiative tests. We conclude the paper by identifying the specific strengths and weaknesses of ASMOV, while pointing out the necessary improvements that need to be made.

1 Presentation of the System

In recent years, ontology alignment (or ontology mapping) has become popular in solving interoperability issues across heterogonous systems in the semantic web. Though many techniques have emerged from the literature [1] [6], the distinction between them is accentuated by the manner in which they exploit the features within an ontology.

ASMOV, an algorithm that automates the ontology alignment process while optionally accepting feedback from a user, uses automatically-adjusting weights based on four features of the ontologies; a more complete description of ASMOV has been presented in [4]. ASMOV computes similarity measures by analyzing the entities in the manner in which they are modeled in the ontology, and the iterative alignments produced by ASMOV are validated by a number of rules and a mapping validation process.

1.1 State, Purpose, General Statement

ASMOV is an automatic ontology matching tool which has been designed in order to facilitate the integration of heterogeneous systems, using their data source ontologies. The OAEI tests help us validate that ASMOV produces ontology alignments with high accuracy and that little user interaction is needed to correct these results. The current ASMOV prototype produces both class-to-class and property-to-property mappings, including mappings from object properties to datatype properties and vice versa.

1.2 Specific Techniques Used

The ASMOV algorithm iteratively calculates the similarity between concepts for a pair of ontologies by analyzing four features: textual description (id, label, and comment), external structure (parents and children), internal structure (property restrictions for classes; types, domains, and ranges for properties), and individual similarity. The measures obtained by comparing these four features are combined into a single confidence value using a weighted sum in a similar manner to [2]. In the case of ASMOV, the initial weights were chosen arbitrarily, and have been optimized based on the benchmark test results. During an automated pre-processing phase, ASMOV contains a mechanism that automatically adjusts the weights based on the information contained in the ontologies. For example, when analyzing the textual information in the pre-processing phase, if ASMOV cannot find meaningful words, it decreases the textual similarity weight based on predetermined rules. These rules are static and have not been adjusted for any of the OAEI 2007 tests.

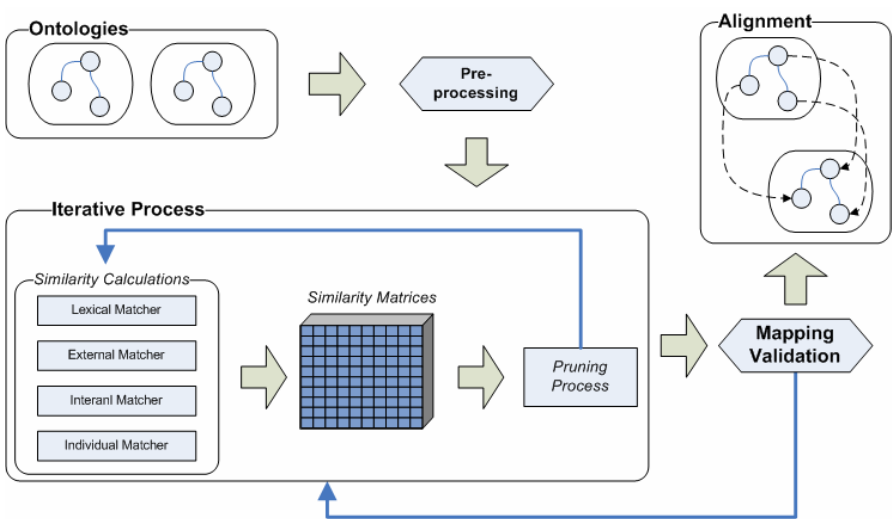

Fig. 1. The Mapping process of ASMOV

Fig. 1 illustrates the Mapping process of ASMOV. The whole process is fully automated.

In the pre-processing phase, the ontologies are loaded into memory using Jena [5]. Each class and property is wrapped and tagged with the meaning of its id and label(s). The meaning of these texts is retrieved using UMLS Metathesaurus [7] for the anatomy test and WordNet [8] for the other tests including the benchmark tests. Through a configuration parameter, a user can force the ASMOV system to use either one, neither, or both of the lexical systems. During the pre-processing phase, a quick analysis of the ontologies being mapped is performed. This analysis entails checking for the presence of properties and meaningful words within the textual description of the classes and properties; the weights are adjusted depending on the result of this analysis.

The second phase of the algorithm is the iterative process. During this phase, pairs of entities (classes and properties) are compared using the four features described earlier, with the resulting overall similarity measure (or confidence value) being stored in a 2-dimensional matrix. At the end of each iteration, a pruning process eliminates the invalid mappings by analyzing two semantic inconsistencies: crisscross mappings and many-to-one mappings. A crisscross mapping occurs whenever a source entity ($\mathcal{SE}p$) and its child ($\mathcal{SE}c$) are mapped to a target entity ($\mathcal{TE}c$) and its parent ($\mathcal{TE}p$) respectively. Many-to-one mappings are inconsistent if it cannot be asserted through the ontology that all the classes in a many-to-one mapping are either equivalent or if each of the classes is subsumed into another. The iterative process stops when the difference in confidence values for two subsequent interations is below a given threshold and no inconsistencies are found by the pruning process, or until a cyclic situation is detected.

After the iterative process is completed, a mapping validation starts. This validation process performs a structural analysis using graphs built from the alignment and information from the ontologies. The validation is performed in three phases: class validation, property validation, and concept-property validation. If any inconsistency is found by this process, the iterative process restarts at the end of the validation process. The inconsistent mappings discovered by the mapping validation process and the pruning process are retained so that ASMOV does not try to align those same entities in subsequent mappings.

1.3 Adaptations Made for the Evaluation

No special adaptations have been made to the ASMOV system in order to run the 2007 OAEI tests; however, four Java classes have been added in order to respectively run the benchmark series of tests, the anatomy tests, the director test and the conference tests, and output the results in the OAEI alignment format. Although the rules stated clearly that all alignments should be run from the same set of parameters, ASMOV was unable to run the anatomy tests under the same conditions due to its iterative nature and the large size of the anatomy ontologies. Two changes had to be made: UMLS was used instead of WordNet as the lexical reference system and the iterative process was stopped when 90% (instead of 100%) of the mappings in the similarity matrix were unchanged in two subsequent iterations. These changes were supplied to the system via properties of a parameters file and required no changes in the coding implementation of ASMOV.

1.4 Link to the ASMOV System

The ASMOV system (including the parameters file) can be downloaded from http://support.infotechsoft.com/integration/ASMOV. A document detailing our approach can also be found there.

1.5 Link to the Set of Alignments Produced by ASMOV

The results of the 2007 OAEI campaign for the ASMOV system can be found at http://support.infotechsoft.com/integration/ASMOV.

2 Results

The 2007 Ontology Alignment Evaluation Initiative campaign consists of four tracks which include: (a) a comparison track, (b) expressive ontologies, (c) directories and thesauri, and (d) a consensus workshop track. Although ASMOV was able to participate in all four tracks, only one out of four tests of the directories and thesauri track was able to be performed due to the large size of the ontologies in the other tests. ASMOV is a Java implementation which uses Jena to parse the RDF and OWL files. All tests were carried out on a PC running Windows XP Professional with a dual-core Intel Pentium processor (2.8 GHz) and 3 gigabytes of memory.

2.1 Benchmark

Because ASMOV's overall similarity calculation (or confidence value) is based on a weighted sum and the weights are automatically adjusted based on the structure of the ontologies being aligned, all tests were ran under the same conditions, the system's default configuration. For the analysis of the results, the benchmark tests are divided into three groups: tests 101-247, tests 248-266, and tests 301-304. The precision, the recall, and the time cost for the individual tests are listed in the Appendix.

2.1.1 Test 101-247

ASMOV performs very well in this set of tests, producing an overall precision and recall of 99%. The less accurate results were produced by the tests 202, 209 and 210. In test 202, although the identifiers of the entities were replaced by random strings and their labels and comments suppressed, ASMOV was still able to leverage other semantic information of the ontologies (namely the hierarchical information, the internal structure of the entities, and the similarity between individuals) in order to generate an alignment of 88% accuracy in both precision and recall. By our analysis, nine out of the eleven incorrectly mapped properties are as accurate as the ones provided within the gold standard, since these properties can only be differentiated by their lexical information (id, label, and comment) and the target ontology (202) has this information suppressed or replaced by random strings for its entities. In test 209, the identifiers and labels were replaced by synonyms and the comments suppressed; the obtained precision and recall were respectively 92% and 90%. In this test, ASMOV suffers mostly because of the measure produced by the similarity calculation. For example, the property '`abstract`' has been mapped to the property '`rights`' instead of the property '`summary`'; the latter was due to the fact that the lexical similarity measure between '`abstract`' and '`rights`' is 0.94 whereas the measure between ‘abstract’ and ‘summary’ is 0.92. In test 210, ASMOV found four incorrect mappings, producing a 97% precision and 95% recall. These errors were due to the fact that the lexical information was in French, which is not supported by WordNet.

2.1.2 Test 248-266

ASMOV’s accuracy decreased in these tests. Both the lexical information and the structure of the target ontologies have been heavily changed. As stated in [3], these tests are the most challenging ones, and it was extremely difficult to recognize the correct alignments. For these tests, the precision ranges from 0.77 to 0.91 and the recall was between 0.24 and 0.89.

2.1.3 Test 301-304

These tests represent four real-world ontologies of bibliographic references. Although there is a high lexical and structure similarity between these tests and the reference ontology, ASMOV encountered some difficulties in the mapping of datatype properties to object properties and vice versa. The overall precision and recall were respectively 85% and 82%.

The following table shows the average performance of ASMOV in terms of the groups of tests described above. The total time cost is also included.

Table 1. Overall Performance on the Benchmark Tests

	101-247	*248-266*	*301-304*	*H-mean*	*Time (sec)*
Precision	0.99	0.85	0.85	0.95	2654.001
Recall	0.99	0.68	0.82	0.90	

2.2 Anatomy

ASMOV’s implementation relies on Jena [5] in order to parse the ontologies to be aligned. We have encountered a few memory issues and found out that Jena does not scale well with large ontologies due to its reasoner. We have thus implemented solutions in ASMOV so that queries that involve the Jena reasoner are bypassed. For example, in order to answer a query for sub-classes or sub-properties, Jena needs to run its reasoner, which is not efficient when dealing with ontologies with large hierarchical structures; the solution to this issue was to maintain a map of parent-child relationships and query this map in order to retrieve the sub-classes and sub-properties. Also, in order to improve the accuracy of the alignment, we have implemented an adapter interface to the UMLS Metathesaurus [9]. With this, the semantic distance between the lexical information within entities (classes and properties) was calculated more accurately, ultimately improving the alignment produced by the system. An alignment was created for the anatomy ontologies using three different configurations: standard configuration, optimal precision configuration, and optimal recall configuration.

- Due to the large size of the anatomy ontologies and the iterative nature of the ASMOV algorithm, three parameters of the s*tandard configuration* had to be changed in order to generate an alignment in an acceptable time frame. The iterative threshold has been changed form 1.0 to 0.9, which means that the iterative process of ASMOV converges once 90% of the mappings do not change in two subsequent iterations. Also the '`ignoreIdInLexicalSim`' parameter was set to false; this parameter setting indicates that the lexical matcher will ignore the local name of the entities. Since the anatomy ontologies deal with the biomedical domain, the UMLS Metathesaurus is more suitable than WordNet. Moreover, since querying the UMLS Metathesaurus for each of the thousands of labels is time-consuming, we have pre-processed the ontologies and stored the semantic information retrieved from the UMLS Metathesaurus into two separate database tables: the first storing the indexed words retrieved (An indexed word is an object that represents a word tied to its semantic meaning or UMLS concept), the second containing the hierarchy of the hypernym relationships from the indexed word in question to a fixed root chosen a priori. Only a subset of the UMLS was used, containing concepts from the NCI Thesaurus and the required dependencies.
- In order to obtain an *optimal overall precision*, the threshold of valid mappings was adjusted, from 0.5 in the system standard configuration, to 0.7. This threshold indicates the acceptable confidences for valid mappings. A value of 70% means that a mapping is deemed acceptable if its confidence value ranges from the best confidence value to 70 % of that value.
- A similar approach was used to obtain the *optimal overall recall*. However, in this case, the threshold of valid mappings was set to 0.0.

Due to the lack of a gold standard in this case, our evaluation was performed by textual analysis of the mappings within the resulting alignments. In this analysis, a correct mapping is a mapping where the entities are equivalent or synonyms, according to their labels and the UMLS Metathesuarus. The results of this analysis are illustrated by the table below in terms of precision and time cost.

Table 2. Performance of ASMOV in the Anatomy test

	Precision	***Time (sec)***
Standard configuration	0.82	54943.656
Optimal Precision	0.89	145382.953
Optimal Recall	0.75	87339.437

2.3 Directory

The standard configuration of ASMOV was used in order to run the directory tests. It took 44 minutes and 27 seconds to run and produced alignments that seem accurate for the most part. As a gold standard for these tests is not available, we are not yet able to report accuracy measures such as precision, recall, and F_1-measure.

2.4 Food

The food ontologies were too large to be run using the current prototype implementation of ASMOV. It took over an hour for Jena to parse the ontologies, and since ASMOV calculates similarity calculations for every pair of entities (class-to-class and property-to-property), the time cost for the alignment is prohibitive. We therefore opt-out of this track; we are currently working on mechanisms to improve the performance of ASMOV.

2.5 Conference

This collection of tests dealing with conference organization contains 14 ontologies. ASMOV was able to create 91 alignments from the ontologies. These ontologies were not analyzed in terms of precision and recall since no gold standard alignments were available.

3 General Comments

3.1 Comments on the Results

ASMOV performed well in the 2007 OAEI tests: the precision and recall of the benchmark tests are higher than those obtained by all entrants in OAEI 2006. This has been achieved by the use of multiple different ontology features and the ability of ASMOV to auto-adjust its weights to the characteristics of the ontologies, which enabled ASMOV to recognize correct alignments even when some information such as lexical similarity was absent. In addition, the use of a semantic validation process enables the algorithm to reject invalid mappings, and improves the overall precision and recall by 5% and 4% respectively.

The main weakness of the algorithm, in its current implementation, is its inability or inefficiency when processing large ontologies such as the anatomy and the food ontologies. We are currently working in mechanisms to improve the performance of the algorithm itself and of its implementation.

3.2 Discussions on the Way to Improve ASMOV

The mapping validation is source dependent, making the alignment process a directional one. Let's consider two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$; what the alignment produces when $\mathcal{O}_1$ is the source and $\mathcal{O}_2$ is the target may be different than the one obtained when the reverse occurs. As our future work, we intend to improve the mapping validation process so that it does not favor the source ontology. Also, the use of Jena as a parser seems not to be ideal, especially when the ontologies are large. For our implementation, we had to bypass some of the methods of Jena that forced calls to its reasoner and caused performance issues. We are currently investigating the use of other parsers such as the OWL-API or more powerful ones, as well as the use of RDF data stores. Although ASMOV will always converge in linear time, the amount of time needed for convergence may be too great when dealing with large ontologies. Also, the use of a checksum to stop the iterative process may cause the algorithm to converge prematurely. Thus, the convergence aspect of ASMOV needs also to be revisited. As stated earlier, ASMOV is to be used as an integration tool; consequently, the confidence values need to be accurate. This accuracy is dictated by the weights which need to be optimum. Therefore, extensive testing of the weighted calculations need to be done to improve the accuracy of ASMOV. In its current state, the user interaction component of ASMOV has not been implemented yet; ASMOV will be extended to be able to present the user with a graphical interface, facilitating system-user interaction.

3.3 Comments on the OAEI 2006 Test Cases

The testing phase of ASMOV was done using the benchmark tests, which were crucial in identifying coding issues and wrong assumptions made in the design phase. In future campaigns, we would like see a benchmarking of larger ontologies so that systems can address scalability issues. Also a benchmark test in different domains such as the biomedical domain (anatomy track) would be useful for systems targeting such domains.

4 Conclusion

In this report, we provided a brief description of an automated alignment tool named ASMOV and analyzed its performance at the 2007 Ontology Alignment Evaluation Initiative campaign. The test results show that ASMOV is effective in the ontology alignment realm, and because of its flexibility, it performs well in multiple ontology domains such as bibliographic references (benchmark tests) and the biomedical domain (anatomy test). We concluded the paper by indicating the strengths and weaknesses of ASMOV, and by stating the direction of our future work.

Acknowledgments. This work is funded by the National Institutes of Health (NIH) under grant R43RR018667. The authors also wish to acknowledge the contribution of Mr. Thomas Taylor and Mr. Patrick Shironoshita of INFOTECH Soft, Inc.

References

1. Euzenat, J. State of the art on ontology alignment. http://www.inrialpes.fr/exmo/cooperation/kweb/heterogeneity/deli/. August, 2004.
2. Euzenat J. and Valtchev P. Similarity-based ontology alignment in OWL-lite. *In Proc. 15th ECAI*, Valencia (ES), 2004, 333-337.
3. Hu, W., Cheng, G., Zheng D., Zhong X., and Qu, Y: The Results of Falcon-AO in the OAEI 2006 Campaign. International Workshop on Ontology Matching (OM 2006).
4. Jean-Mary Y., Kabuka, M. ASMOV: Ontology Alignment with Semantic Validation. Joint SWDB-ODBIS Workshop on Semantics, Ontologies, Databases, September 2007, Vienna, Austria, 15-20
5. Jena from HP Labs http://www.hpl.hp.com/semweb/
6. Rahm, E. and Bernstein, P. A. A survey of approaches to automatic schema matching. The VLDB Journal, 2001, 10:334-350.
7. Unified Medical Language System (UMLS) htt://umlsks.nlm.nih.gov/
8. WordNet http://wordnet.princeton.edu/

Appendix: Raw Results

The OAEI 2007 tests were carried out on a PC running Windows XP Professional with a dual-core Intel Pentium processor (2.8 GHz) and 3 gigabytes of memory.

Matrix of Results

The following table includes the results of ASMOV in the benchmark series of tests. It illustrates the performance of the system in terms of precision (Prec.), recall (Rec.) and processing time (Time). The processing is calculated as follows: ontology parsing time + ASMOV computational time + time involved in the generation of the alignment. For the benchmark tests, the following configuration was used:

- The lexical Similarity is calculated using WordNet and Levenshtein Distance.
- The semantic distance between words was calculated using Lin’s equation.
- The threshold used to stop the iteration process was set to 1.0.
- The threshold indicating that similarity measures have not changed was set to 0.0.
- The valid mappings were the ones that had a confidence value greater or equal to 50 % of the best calculated confidence value.
- The weights associated with missing features were re-distributed proportionally so the similarity measure stayed uniformed.

Note that the same setting was used to run the directory and the consensus tests.

#	Name	Prec.	Rec.	Time (hh.mm.ss.mms)
101	Reference alignment	1.00	1.00	0.4.7.734
102	Irrelevat ontology	NaN	NaN	0.1.6.828
103	Language generalization	1.00	1.00	0.0.55.015
104	Language restriction	1.00	1.00	0.0.54.563
201	No names	1.00	1.00	0.0.52.687
202	No names, no comments	0.88	0.88	0.0.40.985
203	No comments	1.00	1.00	0.0.35.203
204	Naming conventions	1.00	1.00	0.1.0.453
205	Synonyms	1.00	1.00	0.1.15.203
206	Translation	1.00	0.99	0.0.55.937
207		1.00	0.99	0.0.53.563
208		1.00	1.00	0.0.37.093
209		0.92	0.90	0.1.4.578
210		0.97	0.95	0.0.47.532
221	No specialisation	1.00	1.00	0.0.59.687
222	Flatenned hierachy	1.00	1.00	0.0.55.766
223	Expanded hierarchy	1.00	1.00	0.1.3.719
224	No instance	1.00	1.00	0.0.39.281
225	No restrictions	1.00	1.00	0.0.53.0
228	No properties	1.00	1.00	0.0.38.906
230	Flatenned classes	0.99	1.00	0.1.1.438
231		1.00	1.00	0.0.53.844
232		1.00	1.00	0.0.39.875
233		1.00	1.00	0.0.41.672
236		1.00	1.00	0.0.23.797
237		1.00	1.00	0.0.40.75
238		1.00	1.00	0.0.54.297
239		0.97	1.00	0.0.39.11
240		0.97	1.00	0.0.43.171
241		1.00	1.00	0.0.24.766
246		0.97	1.00	0.0.24.156
247		0.94	0.97	0.0.28.312
248		0.86	0.82	0.0.50.922
249		0.89	0.89	0.0.36.328
250		0.91	0.30	0.0.22.281
251		0.83	0.77	0.1.27.407
252		0.87	0.87	0.0.40.734
253		0.85	0.81	0.0.46.203
254		0.83	0.30	0.0.22.031
257		0.91	0.30	0.0.22.36
258		0.82	0.76	0.1.25.047
259		0.87	0.87	0.0.42.828
260		0.78	0.24	0.0.22.407
261		0.91	0.30	0.0.25.578
262		0.83	0.30	0.0.22.281
265		0.77	0.34	0.0.22.734
266		0.91	0.30	0.0.25.485
301	BibTeX/MIT	0.93	0.82	0.0.50.343
302	BibTeX/UMBC	0.68	0.58	0.1.20.563
303	Karlsruhe	0.75	0.86	0.2.42.141
304	INRIA	0.95	0.96	0.0.53.406