LogMap and LogMapLt results for OAEI 2013

Ernesto Jiménez-Ruiz, Bernardo Cuenca Grau, Ian Horrocks

Department of Computer Science, University of Oxford, Oxford, UK

Abstract. We present the results obtained in the OAEI 2013 campaign by our ontology matching system LogMap and its 'lightweight" variant called LogMapLt. The LogMap project started in January 2011 with the objective of developing a scalable and logic-based ontology matching system. This is our fourth participation in the OAEI and the experience has so far been very positive.

1 Presentation of the system

LogMap [11, 12] is a highly scalable ontology matching system with built-in reasoning and inconsistency repair capabilities. LogMap also supports (real-time) user interaction during the matching process, which is essential for use cases requiring very accurate mappings. LogMap is one of the few ontology matching system that (1) can efficiently match semantically rich ontologies containing tens (and even hundreds) of thousands of classes, (2) incorporates sophisticated reasoning and repair techniques to minimise the number of logical inconsistencies, and (3) provides support for user intervention during the matching process. LogMap is also available as a "lightweight" variant called LogMapLt, which essentially only applies (efficient) string matching techniques.

LogMap relies on the following elements, which are keys to its favourable scalability behaviour (see [11, 12] for details).

Lexical indexation. An inverted index is used to store the lexical information contained in the input ontologies. This index is the key to efficiently computing an initial set of mappings of manageable size. Similar indexes have been successfully used in information retrieval and search engine technologies [4].

Logic-based module extraction. The practical feasibility of unsatisfiability detection and repair critically depends on the size of the input ontologies. To reduce the size of the problem, we exploit ontology modularisation techniques. Ontology modules with well-understood semantic properties can be efficiently computed and are typically much smaller than the input ontology (e.g. [7]).

Propositional Horn reasoning. The relevant modules in the input ontologies together with (a subset of) the candidate mappings are encoded in LogMap using a Horn propositional representation. Furthermore, LogMap implements the classic Dowling-Gallier algorithm for propositional Horn satisfiability [8, 10]. Such encoding, although incomplete, allows LogMap to detect unsatisfiable classes soundly and efficiently.

Axiom tracking and greedy repair. LogMap extends Dowling-Gallier's algorithm to track all mappings that may be involved in the unsatisfiability of a class. This extension is key to implementing a highly scalable repair algorithm.

Semantic indexation. The Horn propositional representation of the ontology modules and the mappings are efficiently indexed using an interval labelling schema [1] — an optimised data structure for storing directed acyclic graphs (DAGs) that significantly reduces the cost of answering taxonomic queries [6, 17]. In particular, this semantic index allows us to answer many entailment queries over the input ontologies and the mappings computed thus far as an index lookup operation, and hence without the need for reasoning. The semantic index complements the use of the propositional encoding to detect and repair unsatisfiable classes.

1.1 Adaptations made for the 2013 evaluation

The new version of LogMap also integrates MORe [2, 3] as OWL 2 reasoner. MORe is a modular reasoner which combines a fully-fledged (and slower) reasoner with a profile specific (and more efficient) reasoner.

LogMap's algorithm described in [11–13] has also been adapted to meet the requirements of the new interactive matching track which uses an *Oracle* as expert user.

LogMap aims at making a reduced number of calls to the Oracle, i.e.: only those borderline mappings that cannot be clearly included or excluded with automatic heuristics. For each call to the Oracle, LogMap applies conflict and ambiguity based heuristics (see [12] for details) to reduce the remaining number of calls (i.e. mappings).

Additionally, the interactive algorithm described in [12] has been slightly extended to include object and data properties in the process.

1.2 Link to the system and parameters file

LogMap is open-source and released under GNU Lesser General Public License 3.0.[1] Latest components and source code are available from the LogMap's Google code page: `http://code.google.com/p/logmap-matcher/`.

LogMap distributions can be easily customized through a configuration file containing the matching parameters.

LogMap, including support for interactive ontology matching, can also be used directly through an AJAX-based Web interface: `http://csu6325.cs.ox.ac.uk/`. This interface has been very well received by the community, with more than 900 requests processed so far coming from a broad range of users.

1.3 Modular support for mapping repair

Only very few systems participating in the OAEI 2013 competition implement repair techniques. As a result, existing matching systems (even those that typically achieve very high precision scores) compute mappings that lead in many cases to a large number of unsatisfiable classes.

We believe that these systems could significantly improve their output if they were to implement repair techniques similar to those available in LogMap. Therefore, with

[1] `http://www.gnu.org/licenses/`

Table 1: Results for Benchmark track.

System	biblio 2012			biblioc		
	P	R	F	P	R	F
LogMap	1.00	0.47	0.64	0.73	0.42	0.53
LogMapLt	0.95	0.50	0.66	0.43	0.50	0.46

Table 2: Results for Anatomy track.

System	P	R	F	Time (s)
LogMap	0.918	0.846	0.881	13
LogMapLt	0.962	0.728	0.829	7

the goal of providing a useful service to the community, we have made LogMap's ontology repair module (LogMap-Repair) available as a self-contained software component that can be seamlessly integrated in most existing ontology matching systems [14].

2 Results

In this section, we present a summary of the results obtained by LogMap and LogMapLt in the OAEI 2013 campaign. Please refer to `http://oaei.ontologymatching.org/2013/results/index.html` for complete results.

2.1 Benchmark track

Ontologies in this track have been synthetically generated. The goal of this track is to evaluate the matching systems in scenarios where the input ontologies lack important information (e.g., classes contain no meaningful URIs or labels) [9].

Table 1 summarises the average results obtained by LogMap and LogMapLt. Note that the computation of candidate mappings in LogMap and LogMapLt heavily relies on the similarities between the vocabularies of the input ontologies; hence, there is a direct negative impact in the cases where the labels are replaced by random strings.

2.2 Anatomy track

This track involves the matching of the Adult Mouse Anatomy ontology (2,744 classes) and a fragment of the NCI ontology describing human anatomy (3,304 classes). The reference alignment has been manually curated [19], and it contains a significant number of non-trivial mappings.

Table 2 summarises the results obtained by LogMap and LogMapLt. LogMap ranked 3rd among the systems not using specialised background knowledge. Regarding mapping coherence, only two tools (including LogMap) generated coherent alignments. The evaluation was run on a server with 3.46 GHz (6 cores) and 8GB RAM.

Table 3: Results for Conference track.

System	RA1 reference			RA2 reference			Time (s)
	P	R	F	P	R	F	
LogMap	0.80	0.59	0.68	0.76	0.54	0.63	24
LogMapLt	0.73	0.50	0.59	0.68	0.45	0.54	21

Table 4: Results for Library track.

System	P	R	F	Time (s)
LogMap	0.777	0.645	0.705	99
LogMapLt	0.646	0.771	0.703	20

2.3 Conference track

The Conference track uses a collection of 16 ontologies from the domain of academic conferences [18]. These ontologies have been created manually by different people and are of very small size (between 14 and 140 entities). The track uses two reference alignments RA1 and RA2. RA1 contains manually curated mappings between 21 ontology pairs, while RA2 also contains composed mappings based on the alignments in RA1.

Table 3 summarises the average results obtained by LogMap and LogMapLt. The last column represents the total runtime on generating all 21 alignments. Tests were run on a laptop with Intel Core i5 2.67GHz and 8GB RAM. LogMap ranked 3rd and produced coherent alignments.

2.4 Multifarm track

This track is based on the translation of the OntoFarm collection of ontologies into 9 different languages [16]. Both LogMap and LogMapLt, as expected, obtained poor results since they do not implement specific multilingual techniques.

2.5 Library track

The library track involves the matching of the STW thesaurus (6,575 classes) and the TheSoz thesaurus (8,376 classes). Both of these thesauri provide vocabulary for economic and social sciences. Table 4 summarises the results obtained by LogMap and LogMapLt. The track was run on a computer with one 2.4GHz core with 7GB RAM and 2 cores. LogMap ranked 5th in this track.

2.6 Interactive matching track

The interactive track is based on the conference track and it uses the RA1 reference alignment as Oracle. Table 5 summarizes the obtained results by LogMap with and without the interactive mode activated. LogMap with interactivity (LogMap-Int) improved both the average Precision and Recall wrt LogMap with the interactive mode

Table 5: Results for Interactive track.

System	RA1 reference			Calls	Time (s)
	P	R	F		
LogMap	0.80	0.59	0.68	0	24
LogMap-Int	0.90	0.64	0.73	91	27

Table 6: Summary results for the Large BioMed track

System	Total Time (s)	P	R	F	Inc. Degree.
LogMap-BK	2,391	0.904	0.700	0.785	0.013%
LogMap	2,485	0.910	0.689	0.780	0.015%
LogMapLt	371	0.874	0.517	0.598	34.1%

deactivated, and it only performed 91 calls to the Oracle along the 21 matching tasks (i.e. less than 5 questions per ontology pair).

Not that, although LogMap-Int ranked 1st in the interactive matching track, it could not outperform the best tool in the conference track, which obtained a F-measure of 0.74 (wrt the RA1 reference alignment). Nevertheless, there is still room for improvement and we aim at implementing more sophisticated matching and interactive techniques.

2.7 Large BioMed track

This track consists of finding alignments between the Foundational Model of Anatomy (FMA), SNOMED CT, and the National Cancer Institute Thesaurus (NCI). These ontologies are semantically rich and contain tens of thousands of classes. UMLS Metathesaurus [5] has been selected as the basis for the track reference alignments.

In this track LogMap has been evaluated with two variants: LogMap and LogMap-BK. LogMap-BK uses normalisations and spelling variants from the general (biomedical) purpose UMLS Lexicon,[2] while LogMap has this feature deactivated.

Table 6 summarises the results obtained by LogMap and LogMapLt. The table shows the total time in seconds to complete all tasks in the track and averages for Precision, Recall, F-measure and Incoherence degree. The track was run on a server with 16 CPUs and allocating 15GB RAM.

Regarding mapping coherence, only two tools (including LogMap and its variant LogMap-BK) generated almost coherent alignments. LogMap-BK ranked 3rd among the systems not using specialised background knowledge and 1st among the systems computing almost coherent alignments. LogMapLt was the fastest to complete all tasks.

[2] `http://www.nlm.nih.gov/pubs/factsheets/umlslex.html`

Table 7: Results for Instance matching track.

System	RDFT		
	P	**R**	**F**
LogMap	0.922	0.746	0.812

2.8 Instance matching

This year only LogMap participated in the Instance Matching track. The dataset was based on dbpedia ontology[3] and included controlled transformations in the data (i.e. value and structure transformations).

Table 7 summarises the average results obtained by LogMap. The results are quite promising considering that LogMap does not implement sophisticated instance matching techniques. Furthermore, LogMap outperformed one of the participating tools specialised in instance matching.

Adaptations to the original dataset The original provided dataset was preprocessed in order to be properly interpreted by the OWL API and to avoid inconsistencies when reasoning. Next we summarise the performed changes:

– *Added import of dbpedia:* The dataset (ABOX) is based on dbpedia, however, the dbpedia ontology was not included as TBOX. Hence the OWL API was interpreting the instance entities of the dataset as "annotations" and not as "OWL named individuals". Furthermore, by adding dbpedia TBOX to the datasets, an OWL 2 reasoner could be used to infer the corresponding class type for each instance.
– *Minor changes to dbpedia:* The integration of the provided dataset (ABOX) and dbpedia (TBOX) resulted in an inconsistent knowledge base. The inconsistencies were due to some data property assertion axioms pointing to the incorrect datatype and a functional datatype property which was used in two or more data property assertion axioms with the same subject. To avoid these inconsistencies dbpedia was slightly modified by removing the range and the functionality of the corresponding data properties.
– *Added additional object properties:* The dataset also references the object properties "curriculum", "places" and "label" which are not included in the dbpedia ontology. Hence, these properties has been explicitly declared as OWL object properties.
– *Removal of invalid characters:* the dataset also included some characters that could not be processed by the OWL API and Protégé (e.g. \u).

3 General comments and conclusions

3.1 Comments on the results

LogMap, apart from Benchmark and Multifarm tracks for which does not implement specific techniques, has been one of the top systems in the OAEI 2013. Furthermore,

[3] `http://dbpedia.org/`

it has also been one of the few systems implementing repair techniques and providing (almost) coherent mappings in all tracks.

LogMap's main weakness relies on the fact that the computation of candidate mappings is based on the similarities between the vocabularies of the input ontologies; hence, there is a direct negative impact in the cases where the ontologies are lexically disparate or do not provide enough lexical information (e.g. Benchmark and Multifarm).

3.2 Discussions on the way to improve the proposed system

LogMap is now a stable and mature system that has been made available to the community. There are, however, many exciting possibilities for future work. For example we aim at exploiting background knowledge to be competitive in the Multifarm track and to improve the performance in the other tracks.

3.3 Comments on the OAEI test cases

The number and quality of the OAEI tracks is growing year by year. However, there is always room for improvement:

Comments on the OAEI instance matching track. I consider the 2012 IIMB Instance Matching track more challenging, from the logical point of view, than the current task. The IIMB dataset included a TBOX and the controlled transformations also involved changes on the instance class types. Thus the application of logic based techniques had an important impact since lexically similar instances belonging to two disjoint class types should not be matched.

Comments on the OAEI interactive matching track. The new interactive track has been a very important step forward in the OAEI, however, larger and more challengings tasks should be included. For example, matching tasks (e.g. anatomy and largebio) where the number of questions to the expert user or Oracle may be critical. Furthermore, it is quite unlikely that the expert user will be perfect, thus, the interactive matching track should also consider the evaluation of several Oracles with different error rates such as the evaluation performed in [12].

Comments on the OAEI largebio track. One of the objectives of the largebio track is the creation of a "silver standard" reference alignment by harmonising the output of the different participating systems. In the next OAEI campaign it would be very interesting to actively use this "silver standard" in the construction of the track's reference alignment.

3.4 Comments on the OAEI 2013 measures

Although the *mapping coherence* is a measure already used in the OAEI we consider that is not given yet the required weight in the evaluation. Thus, developers focus on creating matching systems that maximize the F-measure but they disregard the impact of the generated output in terms of logical errors. As a result, even highly precise mappings lead to a large number of unsatisfiable classes.

Thus, we encourage ontology matching system developers to develop their own repair techniques or to use state-of-the-art techniques such as Alcomo [15] and LogMap-Repair (see Section 1.3), which have shown to work well in practice [14].

Acknowledgements

This work was supported by the Seventh Framework Program (FP7) of the European Commission under Grant Agreement 318338, ”Optique”, the Royal Society, and the EPSRC projects Score!, ExODA and MaSI[3].

References

1. Agrawal, R., Borgida, A., Jagadish, H.V.: Efficient management of transitive relationships in large data and knowledge bases. In: ACM SIGMOD Conf. on Management of Data. pp. 253–262 (1989)
2. Armas Romero, A., Cuenca Grau, B., Horrocks, I.: MORe: Modular Combination of OWL Reasoners for Ontology Classification. In: Int'l Sem. Web Conf. (ISWC). pp. 1–16 (2012)
3. Armas Romero, A., Cuenca Grau, B., Horrocks, I., Jiménez-Ruiz, E.: MORe: a Modular OWL Reasoner for Ontology Classification. In: OWL Reasoning Evaluation (ORE) (2013)
4. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval. ACM Press / Addison-Wesley (1999)
5. Bodenreider, O.: The unified medical language system (UMLS): integrating biomedical terminology. Nucleic Acids Research 32, 267–270 (2004)
6. Christophides, V., Plexousakis, D., Scholl, M., Tourtounis, S.: On labeling schemes for the Semantic Web. In: Int'l World Wide Web (WWW) Conf. pp. 544–555 (2003)
7. Cuenca Grau, B., Horrocks, I., Kazakov, Y., Sattler, U.: Modular reuse of ontologies: Theory and practice. J. Artif. Intell. Res. 31, 273–318 (2008)
8. Dowling, W.F., Gallier, J.H.: Linear-time algorithms for testing the satisfiability of propositional Horn formulae. J. Log. Prog. 1(3), 267–284 (1984)
9. Euzenat, J., Rosoiu, M.E., dos Santos, C.T.: Ontology matching benchmarks: Generation, stability, and discriminability. J. Web Sem. 21, 30–48 (2013)
10. Gallo, G., Urbani, G.: Algorithms for testing the satisfiability of propositional formulae. J. Log. Prog. 7(1), 45–61 (1989)
11. Jiménez-Ruiz, E., Cuenca Grau, B.: LogMap: Logic-based and Scalable Ontology Matching. In: Int'l Sem. Web Conf. (ISWC). pp. 273–288 (2011)
12. Jiménez-Ruiz, E., Cuenca Grau, B., Zhou, Y., Horrocks, I.: Large-scale interactive ontology matching: Algorithms and implementation. In: European Conf. on Artif. Intell. (ECAI). pp. 444–449 (2012)
13. Jiménez-Ruiz, E., Grau, B.C., Horrocks, I.: LogMap and LogMapLt results for OAEI 2012. In: Proceedings of the 7th International Workshop on Ontology Matching (2012)
14. Jimenez-Ruiz, E., Meilicke, C., Cuenca Grau, B., Horrocks, I.: Evaluating mapping repair systems with large biomedical ontologies. In: 26th Description Logics Workshop (2013)
15. Meilicke, C.: Alignment Incoherence in Ontology Matching. Ph.D. thesis, University of Mannheim (2011)
16. Meilicke, C., Castro, R.G., Freitas, F., van Hage, W.R., Montiel-Ponsoda, E., de Azevedo, R.R., Stuckenschmidt, H., Šváb-Zamazal, O., Svátek, V., Tamilin, A., Trojahn, C., Wang, S.: MultiFarm: a benchmark for multilingual ontology matching. J. Web Sem. (2012)
17. Nebot, V., Berlanga, R.: Efficient retrieval of ontology fragments using an interval labeling scheme. Inf. Sci. 179(24), 4151–4173 (2009)
18. Šváb, O., Svátek, V., Berka, P., Rak, D., Tomášek, P.: OntoFarm: towards an experimental collection of parallel ontologies. In: Int'l Sem. Web Conf. (ISWC). Poster Session (2005)
19. Zhang, S., Mork, P., Bodenreider, O.: Lessons learned from aligning two representations of anatomy. In: Conf. on Principles of Knowledge Representation and Reasoning (KR) (2004)