Evaluating Ontology Matching Systems on Large, Multilingual and Real-world Test Cases

C. Meilicke[1], O. Šváb-Zamazal[3], C. Trojahn[2], E. Jiménez-Ruiz[4], J.L. Aguirre[2], H. Stuckenschmidt[1], B. Cuenca Grau[4]

[1] University of Mannheim
[2] INRIA & LIG, Grenoble
[3] University of Economics, Prague
[4] University of Oxford, UK

Abstract. In the field of ontology matching, the most systematic evaluation of matching systems is established by the Ontology Alignment Evaluation Initiative (OAEI), which is an annual campaign for evaluating ontology matching systems organized by different groups of researchers. In this paper, we report on the results of an intermediary OAEI campaign called OAEI 2011.5. The evaluations of this campaign are divided in five tracks. Three of these tracks are new or have been improved compared to previous OAEI campaigns. Overall, we evaluated 18 matching systems. We discuss lessons learned, in terms of scalability, multilingual issues and the ability do deal with real world cases from different domains.

1 Introduction

The development in the area of semantic technologies has been enabled by the standardization of knowledge representation languages on the web, in particular RDF and OWL. Based on these languages, many tools have been developed to perform various tasks on the semantic web, such as searching, querying, integrating and reasoning about semi-structured information. However, a crucial step in their large scale adoption in real world applications is the ability to determine the quality of a system in terms of its expected performance on realistic data. Semantic technologies, even though they support a similar functionality, are often not evaluated against the same data sets or the measured results are reproducible with significant effort only. Hence, the challenges on semantic technologies evaluation involves (a) the evaluation of technologies on the basis of test cases that allow conclusions relevant for real world applications and (b) the automatism and reproducibility of the evaluation process and its results.

Regarding the first point, in the field of ontology matching, systematic evaluations are established by the Ontology Alignment Evaluation Initiative (OAEI) [7]. It is an annual evaluation campaign, carried out since 2004, that offers datasets, from different domains, organized by different groups of researchers. Recently, two new datasets have been proposed [17,11] that put a special focus on scalability and multilingual coverage. These are important aspects, due to recent initiatives such as Open Linked Data, where a large amount of multilingual

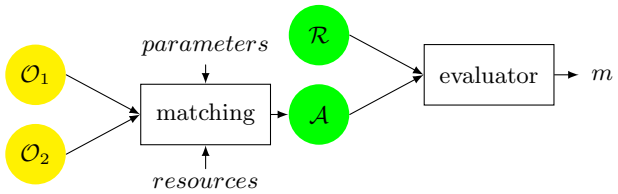

Fig. 1. Ontology matching process and evaluation (from [8]).

data has to be processed. The aim of this paper is to report on the evaluation results of an intermediary OAEI campaign, called OAEI 2011.5, that includes these two datasets. With respect to the second point, the SEALS project (`http://about.seals-project.eu/`) has focused on establishing automatic and systematic evaluation methods for semantic technologies by providing, in particular, a software infrastructure for automatically executing evaluations. This infrastructure involves a controlled execution environment where evaluation organizers can run a set of tools on the same data set. Tools, test data and results in the context of an evaluation campaign are stored in the SEALS repositories. The OAEI 2011.5 campaign is executed on top of this infrastructure. This allows to reproduce all evaluation results that are reported within this paper.

First, we describe ontology matching and the evaluation of ontology matching systems in §2. In §3 we continue with a description of the experimental setting that we applied to OAEI 2011.5. We present the results of our evaluation experiments for each dataset on its own in §4.1-§4.5. Finally, we summarize the most important lessons learned in §5.

2 Ontology matching evaluation

There have been different formalizations of the matching process [1,16]. We follow the framework presented in [8] (see Figure 1). According to this framework, ontology matching systems generate alignments that are sets of correspondences. Given two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$, an example for a correspondence is the statement that `SubjectArea` in $\mathcal{O}_1$ is the same as a `Topic` in $\mathcal{O}_2$ or that `ExternalReviewer` in $\mathcal{O}_1$ is a subclass of `Reviewer` in $\mathcal{O}_2$. In this example, one of the correspondences expresses an equivalence, while the other one expresses a subsumption relation. The core elements of a correspondence are an entity from $\mathcal{O}_1$, an entity from $\mathcal{O}_2$, and a relation that is supposed to hold between them. The matched entities can be classes, properties or instances. In our experiments we are only concerned with matching classes and properties via equivalence.

A minimal data set for evaluating ontology matching systems consists of two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ and an alignment $\mathcal{R}$ that is used as a gold standard. The quality of an alignment $\mathcal{A}$ is measured in terms of its compliance (precision and recall) against the reference alignment $\mathcal{R}$. Precision is defined as $|\mathcal{A} \cap \mathcal{R}|/|\mathcal{A}|$, while recall is defined as $|\mathcal{A} \cap \mathcal{R}|/|\mathcal{R}|$. The F-measure combines precision and

recall and is usually represented as their harmonic mean. For most of our experiments we present aggregated values for these three measures.

In addition to these compliance based measures, it is also important to measure the runtime of a matching process and to understand what factors have an impact on runtime and alignment quality (size of the ontologies, available resources in terms of computational power and additional background knowledge). We have, for example, conducted specific experiments to see whether a matching system can exploit a multi-core architecture to speed up the matching process. Another criteria is the coherence of the generated alignment as defined in [18]. The coherence of an alignment $\mathcal{A}$ is commonly measured with respect to the number of unsatisfiable classes obtained when reasoning with the input ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ together with $\mathcal{A}$. The coherence of an alignment is very important and determines whether it can be used for certain types of applications (e.g., query processing and data migration) that require coherent alignments.

3 Experimental setting

Before describing data sets and tools, we give a brief overview on the overall procedure of OAEI 2011.5. The whole campaign can be divided in three phases:

Preparatory Participants wrap their tools against a predefined interface. Thus, evaluations can be executed locally by using a provided client software. This allows to check whether their tool works correctly with the data sets.

Execution Final tool versions are uploaded by participants and the organizers run the evaluation using SEALS infrastructure with both blind and published datasets. Generated results are stored in the SEALS repository.

Evaluation Stored results are analyzed, aggregated, visualized and published. An extended report on all OAEI 2011.5 results can also be found at `http://oaei.ontologymatching.org/2011.5/results/index.html`

We have run evaluation experiments divided in five different tracks. The tracks MultiFarm and Large BioMed appear for the first time in an OAEI campaign.

Benchmarks For this track, the focus of this campaign was on scalability; to that extent, we considered four "seed ontologies" from different domains and with different sizes. Two of them are completely new (`jerm` and `provenance`), `biblio` and `finance` were already considered in OAEI 2011. All data sets were created artificially by a test generator.

Conference The Conference track uses a collection of ontologies from the domain of conference organization [25]. The ontologies have been created manually by different people and are of moderate size (between 14 and 140 concepts and properties). Reference alignments for a subset of 7 ontologies have been created manually and used since 2008 in OAEI campaigns.

Anatomy The anatomy track is about matching the Adult Mouse Anatomy (2744 classes) and parts of the NCI Thesaurus (3304 classes) describing the human anatomy. The reference alignment, which contains approximately

Table 1. Participation in OAEI 2011 and OAEI 2011.5 tracks B=Benchmarks, C=Conference, M=MultiFarm, A=Anatomy, and L=Large BioMed.

System	2011	2011.5	B	C	M	A	L	State, University
AgrMaker [5]	√			√		√		US, University of Illinois at Chicago
Aroma [6]	√		√	√		√	√	France, INRIA Grenoble Rhône-Alpes
AUTOMSv2 [15]		√	√	√	√			Finland, VTT Technical Research Centre
CIDER [9]	√			√	√			Spain, Universidad Politécnica de Madrid
CODI [21]	√	√	√	√	√	√		Germany, Universität Mannheim
CSA [24]	√			√	√	√	√	Vietnam, University of Ho Chi Minh City
GOMMA [14]		√	√	√		√	√	Germany, Universität Leipzig
Hertuda		√	√	√				Germany, TU Darmstadt
LDOA	√			√				Tunisia, Tunis-El Manar University
Lily [26]	√		√	√		√		China, Southeast University
LogMap [13]	√	√	√	√	√	√	√	UK, University of Oxford
MaasMtch [22]	√	√	√	√	√	√	√	Netherlands, Maastricht University
MapEVO [2]	√	√	√	√	√	√		Germany, Forschungszentrum Informatik
MapPSO [2]	√	√	√	√	√	√		Germany, Forschungszentrum Informatik
MapSSS [4]	√	√	√	√	√	√	√	US, Wright State University
Optima [23]	√			√				US, University of Georgia
WeSeEMtch		√	√	√	√			Germany, TU Darmstadt
YAM++ [20]	√	√	√	√				France, LIRMM

1000 correspondences has been created by domain experts [27]. Aside from some small modifications, the data set has been used for OAEI since 2007.

MultiFarm This track is based on translating the OntoFarm collection to 9 different languages (English, Chinese, Czech, Dutch, French, German, Portuguese, Russian, and Spanish). As a results of this, the track offers challenging test cases for 36 language pairs (further details can be found in [17]).

Large BioMed This track aims at finding alignments between large and semantically rich biomedical ontologies such as FMA, SNOMED CT, and NCI [11]. In the OAEI 2011.5 we have evaluated the FMA-NCI matching problem using two reference alignments based on the UMLS Metathesaurus [3].

Table 1 lists the submitted systems to the SEALS platform in the OAEI 2011 and 2011.5 campaigns. Note that we have also evaluated participants of OAEI 2011, always using the most up-to-date version. As also shown in Table 1, not all tools could be evaluated on all tracks. This is related to problems in processing a certain dataset, memory exceptions or timeouts. We refer the reader to the OAEI 2011.5 web page for details that we omit due to the lack of space.

Note that we have evaluated GOMMA with two different configurations in Anatomy and Large BioMed tracks. GOMMA_{bk} uses specialised background knowledge, while $\text{GOMMA}_{\text{nobk}}$ has this feature deactivated. Furthermore, AgrMaker is also configured to use specialised background knowledge in Anatomy (referred as $\text{AgrMaker}_{\text{bk}}$).

In addition, we implemented two simple matching algorithms. As *Baseline-E* we refer to a matcher based on string equality disregarding capitalization. *LogMapLt* is a string matcher that exploits the creation of an inverted file, a type of index that is commonly used in information retrieval, to efficiently compute correspondences.[5] In general, recall increases from Baseline-E to LogMapLt,

[5] See lexical indexation in [10].

Table 2. Results for benchmark; n/a: not able to run the test, u/r: uncompleted result.

System	biblio	jerm	provenance	finance	avg.	#
MapSSS	0.86 (0.99|0.75)	0.76 (0.98|0.63)	0.75 (0.98|0.61)	0.83 (0.99|0.71)	0.80 (0.99|0.68)	4/4
Aroma	0.76 (0.97|0.63)	0.96 (0.99|0.93)	0.6 (0.78|0.49)	0.7 (0.90|0.57)	0.76 (0.91|0.66)	3/4
WeSeE	0.67 (0.89|0.53)	0.68 (0.99|0.51)	0.64 (0.97|0.48)	0.69 (0.96|0.54)	0.67 (0.95|0.52)	3/4
LogMapLt	0.58 (0.70|0.50)	0.67 (0.98|0.51)	0.66 (0.99|0.50)	0.66 (0.90|0.52)	0.64 (0.89|0.51)	–
Hertuda	0.67 (1.00|0.50)	0.66 (0.96|0.50)	0.54 (0.59|0.50)	0.6 (0.75|0.50)	0.62 (0.83|0.50)	2/4
LogMap	0.48 (0.69|0.37)	0.66 (1.00|0.50)	0.66 (1.00|0.49)	0.6 (0.96|0.43)	0.60 (0.91|0.45)	2/4
GOMMA	0.67 (0.79|0.58)	0.67 (0.97|0.51)	0.22 (0.15|0.55)	0.66 (0.84|0.55)	0.56 (0.69|0.55)	3/4
MaasMtch	0.5 (0.49|0.52)	0.52 (0.52|0.52)	0.5 (0.50|0.50)	0.52 (0.52|0.52)	0.51 (0.51|0.52)	0/4
MapPSO	0.2 (0.58|0.12)	0.05 (0.06|0.05)	0.07 (0.08|0.05)	0.16 (0.28|0.11)	0.12 (0.25|0.08)	0/4
MapEVO	0.37 (0.43|0.33)	0.04 (0.06|0.03)	0.01 (0.02|0.01)	0.02 (0.04|0.01)	0.11 (0.14|0.10)	0/4
Lily	0.75 (0.95|0.62)	0.71 (0.93|0.58)	0.68 (0.92|0.54)	u/r	0.71 (0.93|0.58)	3/3
CODI	0.75 (0.93|0.63)	0.96 (1.00|0.93)	n/a	n/a	0.86 (0.97|0.78)	2/2
YAM++	0.83 (0.99|0.72)	0.72 (0.99|0.56)	u/r	n/a	0.78 (0.99|0.64)	2/2
AUTOMSv2	0.69 (0.97|0.54)	n/a	n/a	n/a	0.69 (0.97|0.54)	1/1

while precision decreases. Note that in many cases it is not easy to top these baselines in terms of F-measure.

4 Evaluation results and discussion

4.1 Benchmarks track

We considered four "seed ontologies" from different domains and with different sizes. For each seed ontology, 94 tests were automatically generated. Table 2 presents the average results for each benchmark, along with the overall average; values are given in the format *F-measure (precision|recall)*. Systems are first ordered according to the number of benchmarks for which an output was provided, then by the highest general average. The last column of the table shows the number of benchmarks for which the matchers generated results and performed at least as good as the LogMapLt baseline. For example, Aroma passed all benchmarks, and topped the results of LogMapLt in 3 of them.

There is no best systems for all benchmarks. However, MapSSS generates the best alignments in terms of F-measure, with Aroma, WeSeE and LogMapLt as followers. We observe a high variance in the results of several systems. Outliers are, for example, a high recall for Aroma with jerm, or a poor precision for GOMMA with provenance. This might depend on inter-dependencies between matching systems and datasets, and needs additional analysis requiring a deep knowledge of the evaluated systems. Such information is, in particular, useful for developers to detect and fix problems specific to their tool.

Regarding runtime, a data set of 15 tests was used for each seed ontology. All the experiments were done in a 3GHz Xeon 5472 (4 cores) machine running Linux Fedora 8 with 8GB RAM. Figure 2 shows a semi-log graph for runtime measurement against benchmark size in terms of classes and properties.

GOMMA, Aroma and LogMap are the fastest tools. We cannot conclude on a general correlation between runtime and quality of alignments. The fastest tools provide in many cases better compliance results than slower tools (MapEVO and

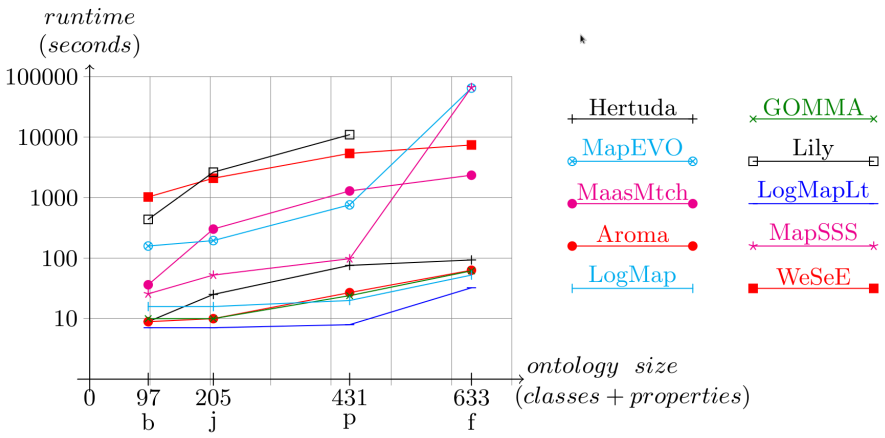

Fig. 2. Benchmark track runtimes. b=biblio, j=jerm, p=provenance, f=finance

Table 3. F-measures and groups assignment within the Conference track.

Group	System	$F_{0.5}$	F_1	F_2
1	YAM++	0.75	0.71	0.67
1	CODI	0.69	0.63	0.58
1	LogMap	0.70	0.61	0.55
1	AgrMaker	0.59	0.57	0.55
1	WeSeEMtch	0.61	0.55	0.49
1	Hertuda	0.65	0.55	0.48
LogMapLt		0.62	0.54	0.48
2	GOMMA	0.67	0.53	0.44
2	AUTOMSv2	0.64	0.52	0.44
Baseline-E		0.64	0.52	0.43

Group	System	$F_{0.5}$	F_1	F_2
3	CSA	0.49	0.51	0.54
3	MaasMatch	0.53	0.49	0.45
3	CIDER	0.55	0.49	0.44
3	MapSSS	0.47	0.46	0.46
3	Lily	0.37	0.40	0.43
3	AROMA	0.35	0.38	0.41
3	Optima	0.26	0.32	0.42
4	LDOA	0.12	0.17	0.28
4	MapPSO	0.11	0.06	0.04
4	MapEVO	0.03	0.02	0.01

MapPSO). However, Lily, which is the slowest tool, provides also alignments of high quality. Furthermore, we observe that tools are more sensitive to the number of classes and properties contained in the ontologies than to the number of axioms; the biblio and jerm ontologies have a similar number of axioms (1332 vs. 1311), but the results for these benchmarks are different for almost all tools.

4.2 Conference track

For OAEI 2011.5, the available reference alignments have been refined and harmonized. New reference alignments have been generated as a transitive closure computed on the original reference alignments. In order to obtain a coherent result, conflicting correspondences (i.e. those causing an unsatisfiability) have been manually inspected and removed. As a result the degree of correctness and completeness of the new reference alignment is probably slightly better than for the old one. However, the differences are relatively restricted.

Table 3 shows the results of all participants with regard to the new reference alignment. $F_{0.5}$-measure, F_1-measure and F_2-measure are computed for the

threshold that provides the highest average F_1-measure. F_1 is the harmonic mean of precision and recall where both are equally weighted; F_2 weights recall higher than precision and $F_{0.5}$ weights precision higher than recall. The matchers shown in the table are ordered according to their highest average F_1-measure. Baselines LogMapLt and Baseline-E divide matchers into four groups. Group 1 consists of best matchers (YAM++, CODI, LogMap, AgrMaker, WeSeEMtch and Hertuda) having better results than baseline LogMapLt in terms of average F_1-measure. Group 2 consists of matchers that perform worse than baseline LogMapLt in terms of average F_1-measure but still better than Baseline-E (GOMMA, AUTOMSv2). Group 3 (CSA, MaasMtch, CIDER, MapSSS, Lily, AROMA and Optima) contains matchers that are worse than Baseline-E but are better (or almost the same) in terms of average F_2-measure. Finally, group 4 consists of matchers (LDOA, MapPSO and MapEVO) performing worse than Baseline-E with regard to all F-measures.

For better comparison with previous years we also evaluated the matching systems with the old reference alignments. The results based on the old reference alignments are in the most of cases better by 0.03 to 0.04 points. Regarding comparison between the OAEI 2011 and OAEI 2011.5 top matchers, YAM++ improved its results by 0.09 percentage points and remained on the top. LogMap worsened by 0.03 percentage points while CODI provided the same results, hence CODI and LogMap changed their position in the order according to F_1-measure.

4.3 Multifarm track

In this dataset, we distinguished between two types of test cases:[6] (i) those test cases where two different ontologies have been translated in different languages; and (ii) those test cases where the same ontology has been translated in different languages. Significant differences between results measured for (i) and (ii) can be observed in Table 4. While the three systems that implement specific multilingual techniques (WeSeE, AUTOMSv2 and YAM++ use different translators for translating the ontologies to English) clearly generate the best results for test cases (i), only one of these systems is among the top systems for type (ii). This subset is dominated by the systems YAM++, CODI, and MapSSS.

We can observe that systems focusing on multilingual methods provide much better results than generic matching systems. However, the absolute results are still not very good, if compared to the top results of the Conference dataset (0.71 F_1-measure). From all specific multilingual methods, the techniques implemented in YAM++ generate the best alignments in terms of F-measure (followed by AUTOMSv2 and WeSeE). It is also an interesting outcome to see that CIDER can generate clearly the best results compared to all other systems with non-specific multilingual systems.

Looking for the average of all systems in test cases (i) and the different pairs of languages, the best scores are for de-en (.29) and es-pt (.26) pairs. We

[6] We used a subset of the whole MultiFarm dataset, omitting the ontologies Edas and Ekaw and suppressing test cases where Russian and Chinese are involved.

Table 4. Multifarm track: results aggregated per matcher over all languages

System	Type (i)				Type (ii)			
	Size	P	R	F	Size	P	R	F
YAM++	1,838	0.54	0.39	0.45	5,838	0.93	0.48	0.63
AUTOMSv2	746	0.63	0.25	0.36	1,379	0.92	0.16	0.27
WeSeE	4,211	0.24	0.39	0.29	5,407	0.76	0.36	0.49
CIDER	737	0.42	0.12	0.19	1,090	0.66	0.06	0.12
MapSSS	1,273	0.16	0.08	0.10	6,008	0.97	0.51	0.67
LogMap	335	0.36	0.05	0.09	400	0.61	0.02	0.04
CODI	345	0.34	0.04	0.08	7,041	0.83	0.51	0.63
MaasMtch	15,939	0.04	0.28	0.08	11,529	0.23	0.23	0.23
LogMapLt	417	0.26	0.04	0.07	387	0.56	0.02	0.04
MapPSO	7,991	0.02	0.06	0.03	6,325	0.07	0.04	0.05
CSA	8,482	0.02	0.07	0.03	8,348	0.49	0.36	0.42
MapEVO	4,731	0.01	0.01	0.01	3,560	0.05	0.01	0.02

Table 5. Anatomy track: precision, recall, recall+, F-measure, and runtimes in seconds

System	Size	Precision	Recall	Recall+	F-measure	Time (s)	Reduction
$\text{AgrMaker}_{\text{bk}}$	1,436	0.942	0.892	0.728	0.917	1037	55%
GOMMA_{bk}	1,468	0.927	0.898	0.736	0.912	37	61%
CODI	1,305	0.960	0.827	0.562	0.888	1177	98%
LogMap	1,391	0.918	0.842	0.588	0.879	35	55%
$\text{GOMMA}_{\text{nobk}}$	1,270	0.952	0.797	0.471	0.868	43	53%
MapSSS	1,213	0.934	0.747	0.337	0.830	563	101%
LogMapLt	1,155	0.956	0.728	0.290	0.827	-	-
Lily	1,370	0.811	0.733	0.510	0.770	657	80%
Aroma	1,279	0.751	0.633	0.344	0.687	59	67%
CSA	2,472	0.464	0.757	0.595	0.576	5026	99%
MaasMtch	2,738	0.430	0.777	0.435	0.554	68498	37%

cannot neglect certain language features in the matching process. The average best F-measures were indeed observed for the pairs of languages that have some degree of overlap in their vocabularies (de-en, fr-pt, es-pt). This is somehow expected, however, we could find exceptions to this behavior. In fact, MultiFarm requires systems exploiting more sophisticated matching strategies than label similarity and for many ontologies in MultiFarm it is the case. It has to be further analysed with a deep analysis of the individual pairs of ontologies. Furthermore, the way the MultiFarm ontologies have been translated by the different human expert may have an impact in the compliance of the translations according to the original ontologies.

4.4 Anatomy track

The results for the anatomy track are presented in Table 5. Top results in terms of F-measure are generated by $\text{AgrMaker}_{\text{bk}}$ and GOMMA_{bk}. These systems are closely followed by CODI, LogMap, $\text{GOMMA}_{\text{nobk}}$, and finally (with some distance) MapSSS. Some systems could not top the LogMapLt baseline in terms of F-measure. However, most of these systems have higher recall scores. Low F-measure values are caused by low precision in all of these cases. This means that those systems find a large amount of non-trivial correspondences.

For measuring runtimes, we have executed all systems on virtual machines with one, two, and four cores each with 8GB RAM. Runtime results shown in Table 5 are based on the execution of the machines with one core. The column

rightmost shows the reduction rate that was achieved when running the tools on the four core environment, i.e. the value is computed as runtime on a 4-core environment divided by runtime using 1-core. A matcher that scales perfectly well would achieve a value of 25%. We executed each system three times and report on average runtimes in seconds.

The fastest systems are LogMap, GOMMA (with and without the use of background knowledge) and AROMA. The enormous variance in measured runtimes is an interesting result. In general, there seems to be no positive correlation between the quality of the alignment and a long runtime. The rightmost column shows that some systems scale well and some systems can not at all exploit a multicore environment. AgrMaker, LogMap and GOMMA reduce their runtime on a 4-core environment up to 50%-65% compared to executing the system with one core. The top system in terms of scalability is MaasMatch; we measured a reduction up to 40%. However, we observed that running a system with 1-core vs. 4-cores has no effect on the order of systems. Differences in runtimes are too strong and thus the availability of additional cores does not change this order.

4.5 Large BioMed track

We evaluated the FMA-NCI matching problem using two reference alignments based on UMLS [11]. The first reference alignment contains 3,024 correspondences and represents the *original* UMLS-based alignment between FMA and NCI [12]. This set, however, leads to a significant number of unsatisfiable classes when integrated with FMA and NCI. The second reference alignment addresses this problem and presents a refined set which contains 2,898 correspondences [10]. Three tasks have been considered involving different fragments of FMA and NCI:

Task 1 consists of matching two (relatively small) modules of FMA and NCI. The FMA module contains 3,696 classes (5% of FMA), while the NCI module contains 6,488 classes (10% of NCI).

Task 2 consists of matching two (relatively large) modules of FMA and NCI. The FMA module contains 28,861 classes (37% of FMA) and the NCI module contains 25,591 classes (38% of NCI).

Task 3 consists of matching the whole FMA and NCI ontologies, which contains 78,989 and 66,724 classes, respectively.

We have executed all systems in a high performance server with 16 CPUs and 10 Gb. Table 6 summarizes the obtained results where systems has been ordered according to the F-measure against the refined reference alignment. Besides precision (P), recall (R), F-measure (F) and runtimes we have also evaluated the coherence of the alignments when reasoning together with the input ontologies.[7]

GOMMA (with its two configurations) and LogMap are a bit ahead in terms of F-measure with respect to Aroma, MaasMatch, CSA and MapSSS, which

[7] We have used the OWL 2 reasoner HermiT [19]

Table 6. Results for the Large BioMed track

Task 1									
System	**Size**	**Unsat.**	**Refined UMLS**			**Original UMLS**			**Time (s)**
			P	**R**	**F**	**P**	**R**	**F**	
$GOMMA_{bk}$	2,878	6,292	0.925	**0.918**	**0.921**	0.957	**0.910**	**0.933**	34
LogMap	2,739	**2**	0.935	0.884	0.909	0.952	0.863	0.905	20
$GOMMA_{nobk}$	2,628	2,130	**0.945**	0.857	0.899	**0.973**	0.846	0.905	27
LogMapLt	2,483	2,104	0.942	0.807	0.869	0.969	0.796	0.874	10
Aroma	2,575	7,558	0.802	0.713	0.755	0.824	0.702	0.758	68
MaasMatch	3,696	9,718	0.580	0.744	0.652	0.597	0.730	0.657	9,437
CSA	3,607	9.590	0.514	0.640	0.570	0.528	0.629	0.574	14,414
MapSSS	1,483	565	0.840	0.430	0.569	0.860	0.422	0.566	571

Task 2									
System	**Size**	**Unsat.**	**Refined UMLS**			**Original UMLS**			**Time (s)**
			P	**R**	**F**	**P**	**R**	**F**	
LogMap	2,664	**5**	**0.877**	0.806	**0.840**	**0.887**	0.782	**0.831**	71
$GOMMA_{bk}$	2,942	7,304	0.817	**0.830**	0.823	0.838	**0.815**	0.826	216
$GOMMA_{nobk}$	2,631	2,127	0.856	0.777	0.815	0.873	0.760	0.813	160
LogMapLt	3,219	12,682	0.726	0.807	0.764	0.748	0.796	0.771	26
CSA	3,607	49.831	0.514	0.640	0.570	0.528	0.629	0.574	14,048
Aroma	3,796	23,298	0.471	0.616	0.534	0.484	0.607	0.539	2,088
MapSSS	2,314	8,401	0.459	0.366	0.407	0.471	0.360	0.408	20,352

Task 3									
System	**Size**	**Unsat.**	**Refined UMLS**			**Original UMLS**			**Time (s)**
			P	**R**	**F**	**P**	**R**	**F**	
LogMap	2,658	**9**	**0.868**	0.796	**0.830**	**0.875**	0.769	0.819	126
$GOMMA_{bk}$	2,983	17,005	0.806	**0.830**	0.818	0.826	**0.815**	**0.820**	1,093
$GOMMA_{nobk}$	2,665	5,238	0.845	0.777	0.810	0.862	0.759	0.807	960
LogMapLt	3,466	26,429	0.675	0.807	0.735	0.695	0.796	0.742	57
CSA	3,607	$>10^5$	0.514	0.640	0.570	0.528	0.629	0.574	14,068
Aroma	4,080	$>10^5$	0.467	0.657	0.546	0.480	0.647	0.551	9,503
MapSSS	2,440	33,186	0.426	0.359	0.390	0.438	0.353	0.391	$>10^5$

could not top the results of our base-line LogMapLt. Furthermore, MaasMatch failed to complete Tasks 2 and 3. $GOMMA_{bk}$ obtained the best results in terms of recall for all three tasks and the best F-measure for Task 1, while LogMap provided the best results in terms of precision and F-measure for Tasks 2 and 3. Finally, $GOMMA_{nobk}$ provided the most precise alignments for Task 1. The use of the original UMLS-based reference alignment did not imply important variations. It is worth mentioning, however, that $GOMMA_{bk}$ improves its results when comparing with the original UMLS alignment and provides the best F-measure for Task 3.

As expected, efficiency decreases as the size of the input ontologies increases. For example, $GOMMA_{bk}$'s F-measure decreased from 0.921 (Task 1) to 0.818 (Task 3). Furthermore, $GOMMA_{bk}$'s runtime also increased from 34 seconds to more than 18 minutes. CSA is an exception since (surprisingly) maintained exactly the same results for the three tasks.

Regarding mapping coherence, only LogMap generated an 'almost' clean output in all three tasks. Although $GOMMA_{nobk}$ also provides highly precise output correspondences, they lead to a huge amount of unsatisfiable classes.

5 Lessons learned and future work

In the following we summarize the most important lessons learned and raise some conclusions related to future work.

Multilingual coverage Only 3 systems are able to deal, at a minimal level, with the multilingual labels in MultiFarm, thus there is plenty of room for improvements towards a multilingual semantic web. We could also observe a strong correlation between the ranking in Benchmark and the ranking in MultiFarm type (ii), for non-specific multilingual systems, while there is no (or very weak) correlation between results for tests of types (i) and (ii).

Precision and recall It is hard to top our baselines in terms of F-measure. This is related to the fact that it is not easy to detect non-trivial correspondences without a (significant) loss in precision. Nevertheless, comparing OAEI 2011.5 and OAEI 2011 there is an increase in a number of high quality matchers for tracks that have not changed (Anatomy and Conference).

Computational resources There is a high variance in runtimes between different matching algorithms. These differences cannot be counterbalanced by additionally computational power in number of cores. At the same time, we have also seen that some systems can cope with large ontologies only with large amount of RAM.

Scalability The Benchmark results indicate that there are two families of systems. Those that scale well with respect to ontology size, and those where we find big differences in runtimes. Moreover, we have learned that the relevant factor is not the number of axioms, but the number of classes and properties.

Coherence As shown in the Large BioMed track even highly precise alignment sets may lead to a huge number of unsatisfiable classes. Thus, the use of techniques to assess alignment coherence is critical. However, LogMap, CODI, and YAM++ are the only systems that use such techniques.[8] In future evaluations this aspect should not be neglected.

Large ontologies Efficiency significantly decreases as the size of the input ontologies increases (see Benchmark and Large BioMed tracks). In the OAEI 2012, however, we intend to evaluate even harder problems such as FMA-SNOMED and SNOMED-NCI [11]. Although these matching problems will represent another significant leap in complexity, we take our positive experiences as an indication that matching these ontologies is still feasible.

Acknowledgements

Some of the authors are partially supported by the EU FP7 project SEALS (IST-2009-238975). Ondřej Šváb-Zamazal has been partially supported by the CSF grant no. P202/10/0761. Ernesto Jiménez-Ruiz was supported by the EPSRC project LogMap. Bernardo Cuenca Grau was supported by the Royal Society.

[8] Additional results for the Conference track show that, besides LogMap, CODI and YAM generate also coherent alignments in many cases.

References

1. P. Bernstein, A. Halevy, and R. Pottinger. A vision of management of complex models. *ACM SIGMOD Record*, 29(4):55–63, 2000.
2. J. Bock, C. Dánschel, and M. Stumpp. MapPSO and MapEVO results for OAEI 2011. In *Proc. of 6th OM Workshop*, pages 142–147, 2011.
3. O. Bodenreider. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nuc. acids res.*, 32, 2004.
4. M. Cheatham. MapSSS results for OAEI 2011. In *Proc. of 6th OM Workshop*, 2011.
5. I. F. Cruz, F. P. Antonelli, and C. Stroe. Agreementmaker: Efficient matching for large real-world schemas and ontologies. *PVLDB*, 2(2):1586–1589, 2009.
6. J. David, F. Guillet, and H. Briand. Association Rule Ontology Matching Approach. *J. Sem. Web Inf. Sys.*, 3(2):27–49, 2007.
7. J. Euzenat, C. Meilicke, H. Stuckenschmidt, P. Shvaiko, and C. Trojahn. Ontology alignment evaluation initiative: Six years of experience. *J. Data Sem.*, 15, 2011.
8. J. Euzenat and P. Shvaiko. *Ontology matching*. Springer, Heidelberg (DE), 2007.
9. J. Gracia, J. Bernad, and E. Mena. Ontology matching with CIDER: evaluation report for OAEI 2011. In *Proc. of 6th OM Workshop*, pages 126–133, 2011.
10. E. Jiménez-Ruiz and B. Cuenca Grau. LogMap: Logic-based and Scalable Ontology Matching. In *10th International Semantic Web Conference (ISWC)*, 2011.
11. E. Jiménez-Ruiz, B. Cuenca Grau, and I. Horrocks. Exploiting the UMLS Metathesaurus in the Ontology Alignment Evaluation Initiative. In *Proc. of 2nd International Workshop on Exploiting Large Knowledge Repositories (E-LKR)*, 2012.
12. E. Jiménez-Ruiz, B. Cuenca Grau, I. Horrocks, and R. Berlanga. Logic-based assessment of the compatibility of UMLS ontology sources. *J. Biomed. Sem.*, 2011.
13. E. Jiménez-Ruiz, B. Cuenca Grau, Y. Zhou, and I. Horrocks. Large-scale interactive ontology matching: Algorithms and implementation. In *Proc. of ECAI*, 2012.
14. T. Kirsten, A. Gross, M. Hartung, and E. Rahm. GOMMA: a component-based infrastructure for managing and analyzing life science ontologies and their evolution. *J. Biomed. Sem.*, 2:6, 2011.
15. K. Kotis, A. Katasonov, and J. Leino. Aligning Smart and Control Entities in IoT. In *5th Conference on Internet of Things and Smart Spaces*, 2012.
16. M. Lenzerini. Data integration: A theoretical perspective. In *Proc. 21st Symposium on Principles of Database Systems (PODS)*, pages 233–246, 2002.
17. C. Meilicke, R. G. Castro, F. Freitas, W. R. van Hage, E. Montiel-Ponsoda, R. R. de Azevedo, H. Stuckenschmidt, O. Šváb-Zamazal, V. Svátek, A. Tamilin, C. Trojahn, and S. Wang. Multifarm: A benchmark for multilingual ontology matching. *Journal of Web Semantics*, 2012. Accepted for publication.
18. C. Meilicke and H. Stuckenschmidt. Incoherence as a basis for measuring the quality of ontology mappings. In *Proc. of the Ontology Matching Workshop*, 2008.
19. B. Motik, R. Shearer, and I. Horrocks. Hypertableau Reasoning for Description Logics. *Journal of Artificial Intelligence Research*, 36:165–228, 2009.
20. D. Ngo, Z. Bellahsene, and R. Coletta. YAM++ – results for OAEI 2011. In *Proc. of 6th OM Workshop*, pages 228–235, 2011.
21. M. Niepert, C. Meilicke, and H. Stuckenschmidt. A probabilistic-logical framework for ontology matching. In *Proc. of AAAI*, 2010.
22. F. C. Schadd and N. Roos. MaasMatch results for OAEI 2011. In *Proc. of 6th OM Workshop*, pages 179–183, 2011.

23. U. Thayasivam and P. Doshi. Optima results for OAEI 2011. In *Proc. of 6th OM Workshop*, pages 204–211, 2011.
24. Q.-V. Tran, R. Ichise, and B.-Q. Ho. Cluster-based similarity aggregation for ontology matching. In *Proc. of 6th OM Workshop*, pages 142–147, 2011.
25. O. Šváb, V. Svátek, P. Berka, D. Rak, and P. Tomášek. Ontofarm: Towards an experimental collection of parallel ontologies. In *Poster Track of ISWC*, 2005.
26. P. Wang. Lily results on SEALS platform for OAEI 2011. In *Proc. of 6th OM Workshop*, pages 156–162, October 2011.
27. S. Zhang, P. Mork, and O. Bodenreider. Lessons learned from aligning two representations of anatomy. In *Proc. 13th KR Conference*, pages 555–560, 2004.