Results of the Ontology Alignment Evaluation Initiative 2016*

Manel Achichi[1], Michelle Cheatham[2], Zlatan Dragisic[3], Jérôme Euzenat[4], Daniel Faria[5], Alfio Ferrara[6], Giorgos Flouris[7], Irini Fundulaki[7], Ian Harrow[8], Valentina Ivanova[3], Ernesto Jiménez-Ruiz[9,10], Elena Kuss[11], Patrick Lambrix[3], Henrik Leopold[12], Huanyu Li[3], Christian Meilicke[11], Stefano Montanelli[6], Catia Pesquita[13], Tzanina Saveta[7], Pavel Shvaiko[14], Andrea Splendiani[15], Heiner Stuckenschmidt[11], Konstantin Todorov[1], Cássia Trojahn[16], and Ondřej Zamazal[17]

[1] LIRMM/University of Montpellier, France
lastname@lirmm.fr
[2] Data Semantics (DaSe) Laboratory, Wright State University, USA
michelle.cheatham@wright.edu
[3] Linköping University & Swedish e-Science Research Center, Linköping, Sweden
{zlatan.dragisic,valentina.ivanova,patrick.lambrix}@liu.se
[4] INRIA & Univ. Grenoble Alpes, Grenoble, France
Jerome.Euzenat@inria.fr
[5] Instituto Gulbenkian de Ciência, Lisbon, Portugal
dfaria@igc.gulbenkian.pt
[6] Università degli studi di Milano, Italy
{alfio.ferrara,stefano.montanelli}@unimi.it
[7] Institute of Computer Science-FORTH, Heraklion, Greece
{jsaveta,fgeo,fundul}@ics.forth.gr
[8] Pistoia Alliance Inc., USA
ian.harrow@pistoiaalliance.org
[9] Department of Informatics, University of Oslo, Norway
ernestoj@ifi.uio.no
[10] Department of Computer Science, University of Oxford, UK
[11] University of Mannheim, Germany
{christian,elena,heiner}@informatik.uni-mannheim.de
[12] Vrije Universiteit Amsterdam, The Netherlands
h.leopold@vu.nl
[13] LASIGE, Faculdade de Ciências, Universidade de Lisboa, Portugal
cpesquita@di.fc.ul.pt
[14] TasLab, Informatica Trentina, Trento, Italy
pavel.shvaiko@infotn.it
[15] Novartis Institutes for Biomedical Research, Basel, Switzerland
andrea.splendiani@novartis.com
[16] IRIT & Université Toulouse II, Toulouse, France
{cassia.trojahn}@irit.fr
[17] University of Economics, Prague, Czech Republic
ondrej.zamazal@vse.cz

Abstract. Ontology matching consists of finding correspondences between semantically related entities of two ontologies. OAEI campaigns aim at comparing

* The official results of the campaign are on the OAEI web site.

ontology matching systems on precisely defined test cases. These test cases can use ontologies of different nature (from simple thesauri to expressive OWL ontologies) and use different modalities, e.g., blind evaluation, open evaluation, or consensus. OAEI 2016 offered 9 tracks with 22 test cases, and was attended by 21 participants. This paper is an overall presentation of the OAEI 2016 campaign.

1 Introduction

The Ontology Alignment Evaluation Initiative[1] (OAEI) is a coordinated international initiative, which organises the evaluation of an increasing number of ontology matching systems [18,21]. Its main goal is to compare systems and algorithms openly and on the same basis, in order to allow anyone to draw conclusions about the best matching strategies. Furthermore, our ambition is that, from such evaluations, tool developers can improve their systems.

Two first events were organised in 2004: *(i)* the Information Interpretation and Integration Conference (I3CON) held at the NIST Performance Metrics for Intelligent Systems (PerMIS) workshop and *(ii)* the Ontology Alignment Contest held at the Evaluation of Ontology-based Tools (EON) workshop of the annual International Semantic Web Conference (ISWC) [41]. Then, a unique OAEI campaign occurred in 2005 at the workshop on Integrating Ontologies held in conjunction with the International Conference on Knowledge Capture (K-Cap) [4]. From 2006 until now, the OAEI campaigns were held at the Ontology Matching workshop, collocated with ISWC [19,17,6,14,15,16,2,9,12,8], which this year took place in Kobe, JP[2].

Since 2011, we have been using an environment for automatically processing evaluations (§2.2), which has been developed within the SEALS (Semantic Evaluation At Large Scale) project[3]. SEALS provided a software infrastructure, for automatically executing evaluations, and evaluation campaigns for typical semantic web tools, including ontology matching. In the OAEI 2016, all systems were executed under the SEALS client in all tracks, and evaluated with the SEALS client in all tracks. This year we welcomed two new tracks: the Disease and Phenotype track, sponsored by the Pistoia Alliance Ontologies Mapping project, and the Process Model Matching track. Additionally, the Instance Matching track featured a total of 7 matching tasks based on all new data sets. On the other hand, the OA4QA track was discontinued this year.

This paper synthesises the 2016 evaluation campaign. The remainder of the paper is organised as follows: in Section 2, we present the overall evaluation methodology that has been used; Sections 3-11 discuss the settings and the results of each of the test cases; Section 12 overviews lessons learned from the campaign; and finally, Section 13 concludes the paper.

2 General methodology

We first present the test cases proposed this year to the OAEI participants (§2.1). Then, we discuss the resources used by participants to test their systems and the execution

[1] http://oaei.ontologymatching.org
[2] http://om2016.ontologymatching.org
[3] http://www.development.seals-project.eu

environment used for running the tools (§2.2). Finally, we describe the steps of the OAEI campaign (§2.3-2.5) and report on the general execution of the campaign (§2.6).

2.1 Tracks and test cases

This year's OAEI campaign consisted of 9 tracks gathering 22 test cases, and different evaluation modalities:

The benchmark track (§3): Like in previous campaigns, a systematic benchmark series has been proposed. The goal of this benchmark series is to identify the areas in which each matching algorithm is strong or weak by systematically altering an ontology. This year, we generated a new benchmark based on the original bibliographic ontology and another benchmark using a film ontology.

The expressive ontology track offers alignments between real world ontologies expressed in OWL:

- **Anatomy (§4):** The anatomy test case is about matching the Adult Mouse Anatomy (2744 classes) and a small fragment of the NCI Thesaurus (3304 classes) describing the human anatomy.
- **Conference (§5):** The goal of the conference test case is to find all correct correspondences within a collection of ontologies describing the domain of organising conferences. Results were evaluated automatically against reference alignments and by using logical reasoning techniques.
- **Large biomedical ontologies (§6):** The largebio test case aims at finding alignments between large and semantically rich biomedical ontologies such as FMA, SNOMED-CT, and NCI. The UMLS Metathesaurus has been used as the basis for reference alignments.
- **Disease & Phenotype (§7):** The disease & phenotype test case aims at finding alignments between two disease ontologies (DOID and ORDO) as well as between human (HPO) and mammalian (MP) phenotype ontologies. The evaluation was semi-automatic: consensus alignments were generated based on those produced by the participating systems, and the unique mappings found by each system were evaluated manually.

Multilingual

- **Multifarm (§8):** This test case is based on a subset of the Conference data set, translated into ten different languages (Arabic, Chinese, Czech, Dutch, French, German, Italian, Portuguese, Russian, and Spanish) and the corresponding alignments between these ontologies. Results are evaluated against these alignments.

Interactive matching

- **Interactive (§9):** This test case offers the possibility to compare different matching tools which can benefit from user interaction. Its goal is to show if user interaction can improve matching results, which methods are most promising and how many interactions are necessary. Participating systems are evaluated on the conference data set using an oracle based on the reference alignment, which can generate erroneous responses to simulate user errors.

Instance matching (§10). The track aims at evaluating the performance of matching tools when the goal is to detect the degree of similarity between pairs of

test	formalism	relations	confidence	modalities	language	SEALS
benchmark	OWL	=	[0 1]	blind	EN	√
anatomy	OWL	=	[0 1]	open	EN	√
conference	OWL	=, <=	[0 1]	open+blind	EN	√
largebio	OWL	=	[0 1]	open	EN	√
phenotype	OWL	=	[0 1]	blind	EN	√
multifarm	OWL	=	[0 1]	open+blind	AR, CZ, CN, DE, EN, ES, FR, IT, NL, RU, PT	√
interactive	OWL	=, <=	[0 1]	open	EN	√
instance	OWL	=	[0 1]	open(+blind)	EN(+IT)	√
process model	OWL	<=	[0 1]	open+blind	EN	√

Table 1. Characteristics of the test cases (open evaluation is made with already published reference alignments and blind evaluation is made by organisers from reference alignments unknown to the participants).

items/instances expressed in the form of OWL Aboxes. Three independent tasks are defined:

SABINE: The task is articulated in two sub-tasks called *inter-lingual mapping* and *data linking*. Both sub-tasks are based on OWL ontologies containing topics as instances of the class "Topic". In inter-lingual mapping, two ontologies are given, one containing topics in the English language and one containing topics in the Italian language. The goal is to discover mappings between English and Italian topics. In data linking, the goal is to discover the DBpedia entity which better corresponds to each topic belonging to a source ontology.

SYNTHETIC: The task is articulated in two sub-tasks called *UOBM* and *SPIMBENCH*. In UOBM, the goal is to recognize when two OWL instances belonging to different data sets, i.e., ontologies, describe the same individual. In SPIMBENCH, the goal is to determine when two OWL instances describe the same Creative Work. Data Sets are produced by altering a set of original data.

DOREMUS: The DOREMUS task contains real world data coming from the French National Library (BnF) and the Philharmonie de Paris (PP). Data are about classical music work and follow the DOREMUS model (one single vocabulary for both datasets). Three sub-tasks are defined called *nine heterogeneities*, *four heterogeneities*, and *false-positive trap* characterized by different degrees of heterogeneity in work descriptions.

Process Model Matching (§11): The track is concerned with the application of ontology matching techniques to the problem of matching process models. It is based on a data set used in the Process Model Matching Campaign 2015 [3], which has been converted to an ontological representation. The data set contains nine process models which represent the application process for a master program of German universities as well as reference alignments between all pairs of models.

Table 1 summarises the variation in the proposed test cases.

2.2 The SEALS client

Since 2011, tool developers had to implement a simple interface and to wrap their tools in a predefined way including all required libraries and resources. A tutorial for tool

wrapping was provided to the participants, describing how to wrap a tool and how to use the SEALS client to run a full evaluation locally. This client is then executed by the track organisers to run the evaluation. This approach ensures the reproducibility and comparability of the results of all systems.

2.3 Preparatory phase

Ontologies to be matched and (where applicable) reference alignments have been provided in advance during the period between June 1^{st} and June 30^{th}, 2016. This gave potential participants the occasion to send observations, bug corrections, remarks and other test cases to the organisers. The goal of this preparatory period is to ensure that the delivered tests make sense to the participants. The final test base was released on July 15^{th}, 2016. The (open) data sets did not evolve after that.

2.4 Execution phase

During the execution phase, participants used their systems to automatically match the test case ontologies. In most cases, ontologies are described in OWL-DL and serialised in the RDF/XML format [11]. Participants can self-evaluate their results either by comparing their output with reference alignments or by using the SEALS client to compute precision and recall. They can tune their systems with respect to the non blind evaluation as long as the rules published on the OAEI web site are satisfied. This phase has been conducted between July 15^{th} and August 31^{st}, 2016. Unlike previous years, we requested a mandatory registration of systems and a preliminary evaluation of wrapped systems by July 31st. This reduced the cost of debugging systems with respect to issues with the SEALS client during the Evaluation phase as it happened in the past.

2.5 Evaluation phase

Participants were required to submit their wrapped tools by August 31^{st}, 2016. Tools were then tested by the organisers and minor problems were reported to some tool developers, who were given the opportunity to fix their tools and resubmit them.

Initial results were provided directly to the participants between September 23^{rd} and October 15^{th}, 2016. The final results for most tracks were published on the respective pages of the OAEI website by October 15^{th}, although some tracks were delayed.

The standard evaluation measures are usually precision and recall computed against the reference alignments. More details on the evaluation are given in the sections for the test cases.

2.6 Comments on the execution

Following the recent trend, the number of participating systems has remained approximately constant at slightly over 20 (see Figure 1). This year was no exception, as we counted 21 participating systems (out of 30 registered systems). Remarkably, participating systems have changed considerably between editions, and new systems keep emerging. For example, this year 10 systems had not participated in any of the previous OAEI campaigns. The list of participants is summarised in Table 2. Note that some systems were also evaluated with different versions and configurations as requested by developers (see test case sections for details).

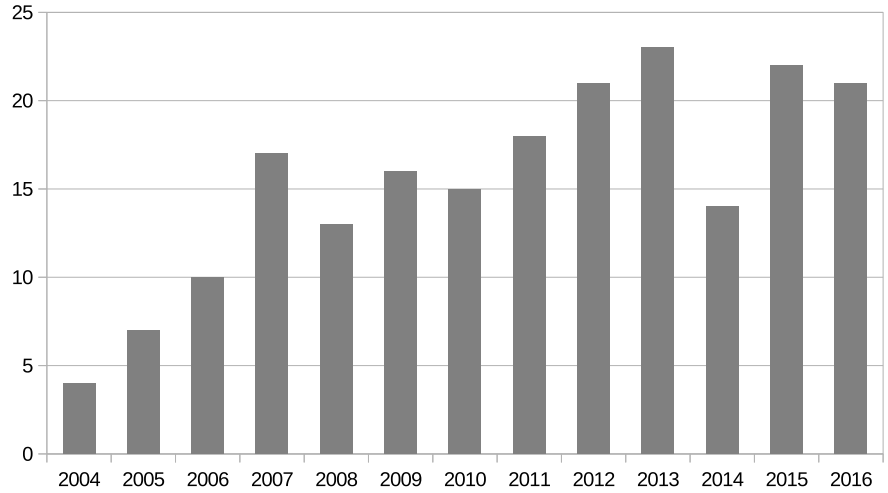


Fig. 1. Number of systems participating in OAEI per year.

3 Benchmark

The goal of the benchmark data set is to provide a stable and detailed picture of each algorithm. For that purpose, algorithms are run on systematically generated test cases.

3.1 Test data

The systematic benchmark test set is built around a seed ontology and many variations of it. Variations are artificially generated by discarding and modifying features from a seed ontology. Considered features are names of entities, comments, the specialisation hierarchy, instances, properties and classes. This test focuses on the characterisation of the behaviour of the tools rather than having them compete on real-life problems. Full description of the systematic benchmark test set can be found on the OAEI web site.

Since OAEI 2011.5, the test sets are generated automatically from different seed ontologies [20]. This year, we used two ontologies:

biblio The bibliography ontology used in the previous years which concerns bibliographic references and is inspired freely from BibTeX;

film A movie ontology developed in the MELODI team at IRIT (FilmographieV1[4]). It uses fragments in French and labels in French and English.

The characteristics of these ontologies are described in Table 3.

The film data set was not available to participants when they submitted their systems. The tests were also blind for the organisers since we did not look into them before running the systems.

The reference alignments are still restricted to named classes and properties and use the "=" relation with confidence of 1.

[4] `https://www.irit.fr/recherches/MELODI/ontologies/FilmographieV1.owl`

System	Alin	AML	CroLOM	CroMatcher	DiSMatch	DKP-AOM	DKP-AOM-Lite	FCA-Map	Lily	LogMap	LogMap-Bio	LogMapLt	LPHOM	Lyam++	NAISC	PhenoMF	PhenoMM	PhenoMP	RiMOM	SimCat	XMap	Total=21
Confidence	√	√	√	√	√					√	√		√	√	√				√	√	√	13
benchmarks		√		√					√	√		√									√	6
anatomy	√	√		√		√	√	√	√	√	√	√	√	√							√	13
conference	√	√		√		√	√	√	√	√		√	√	√	√						√	13
largebio	√	√		√		√	√	√	√	√	√	√	√	√							√	13
phenotype		√			√			√		√	√	√		√		√	√	√			√	11
multifarm		√	√							√			√	√						√	√	7
interactive	√	√								√											√	4
process model		√				√	√			√												4
instance		√								√		√							√			4
total		9	1	4	1	4	4	4	4	9	3	6	4	5	1	1	1	1	1	1	7	77

Table 2. Participants and the state of their submissions. Confidence stands for the type of results returned by a system: it is ticked when the confidence is a non-boolean value.

Test set	biblio	film
classes+prop	33+64	117+120
instances	112	47
entities	209	284
triples	1332	1717

Table 3. Characteristics of the two seed ontologies used in benchmarks.

3.2 Results

In order to avoid the discrepancy of last year, all systems were run in the most simple homogeneous setting. So, this year, we can write anew: All tests have been run entirely in the same conditions with the same strict protocol.

Evaluations were run on a Debian Linux virtual machine configured with four processors and 8GB of RAM running under a Dell PowerEdge T610 with 2*Intel Xeon Quad Core 2.26GHz E5607 processors and 32GB of RAM, under Linux ProxMox 2 (Debian). All matchers where run under the SEALS client using Java 1.8 and a maximum heap size of 8GB.

As a result, many systems were not able to properly match the benchmark. Evaluators availability is not unbounded and it was not possible to pay attention to each system as much as necessary.

Participation From the 21 systems participating to OAEI this year, only 10 systems were providing results for this track. Several of these systems encountered problems:

However we encountered problems with one very slow matcher (LogMapBio) that has been run anyway. RiMOM did not terminate, but was able to provide (empty) alignments for biblio, not for film. No timeout was explicitly set.

Reported figures are the average of 5 runs. As has already been shown in [20], there is not much variance in compliance measures across runs.

Compliance Table 4 synthesises the results obtained by matchers.

	biblio			film		
Matcher	Prec.	F-m.	Rec.	Prec.	F-m.	Rec.
edna	.35(.58)	.41(.54)	.51(.50)	.43 (.68)	.47 (.58)	.50 (.50)
AML	1.0	.38	.24	1.0	.32	.20
CroMatcher	.96 (.60)	.89 (.54)	.83 (.50)		NaN	
Lily	.97 (.45)	.89 (.40)	.83 (.36)	.97 (.39)	.81 (.31)	.70 (.26)
LogMap	.93 (.90)	.55 (.53)	.39 (.37)	.83 (.79)	.13 (.12)	.07 (.06)
LogMapLt	.43	.46	.50	.62	.51	.44
PhenoMF	.03	.01	.01	.03	.01	.01
PhenoMM	.03	.01	.01	.03	.01	.01
PhenoMP	.02	.01	.01	.03	.01	.01
XMap	.95 (.98)	.56 (.57)	.40 (.40)	.78 (.84)	.60 (.62)	.49 (.49)
LogMapBio	.48 (.48)	.32 (.30)	.24 (.22)	.59 (.58)	.07 (.06)	.03 (.03)

Table 4. Aggregated benchmark results: Harmonic means of precision, F-measure and recall, along with their confidence-weighted values.

Systems that participated previously (AML, CroMatcher, Lily, LogMap, LogMapLite, XMap) still obtain the best results with Lily and CroMatcher still achieving an impressive .89 F-measure (against .90 and .88 last year). They combine very high precision (.96 and .97) with high recall (.83). The PhenoXX suite of systems return huge but poor alignments. It is surprising that some of the systems (AML, LogMapLite) do not clearly outperform edna (our edit distance baseline).

On the film data set (which was not known from the participants when submitting their systems, and actually have been generated afterwards), the results of biblio are

fully confirmed: (1) those system able to return results were still able to do it besides CroMatcher and those unable, were still not able; (2) the order between these systems and their performances are commensurate. Point (1) shows that these are robust systems. Point (2) shows that the performances of these system are consistent across data sets, hence we are indeed measuring something. However, (2) has for exception LogMap and LogMapBio whose precision is roughly preserved but whose recall dramatically drops. A tentative explanation is that film contains many labels in French and these two systems rely too much on WordNet. Anyway, these and CroMatcher seem to show some overfit to biblio.

Polarity Besides LogMapLite, all systems have higher precision than recall as usual and usually very high precision as shown on the triangle graph for biblio (Figure 2). This can be compared with last year.

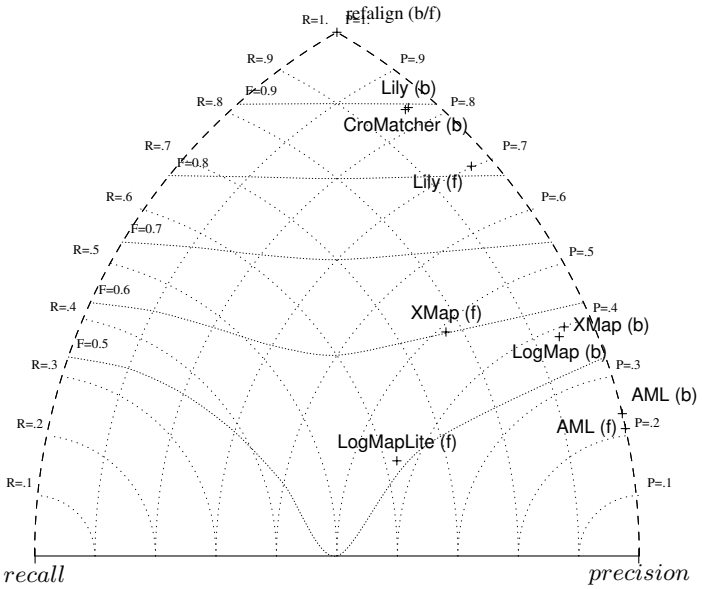


Fig. 2. Triangle view on the benchmark data sets (biblio=(b), film=(f), run 5, non present systems have too low F-measure, below .5).

The precision/recall graph (Figure 3) confirms that, as usual, there are a level of recall unreachable by any system and this is where some of them go to catch their good F-measure.

Concerning confidence-weighted measures, there are two types of systems: those (CroMatcher, Lily) which obviously threshold their results but keep low confidence values and those (LogMap, XMap, LogMapBio) which provide relatively faithful measures. The former shows a strong degradation of the measured values while the latters resist

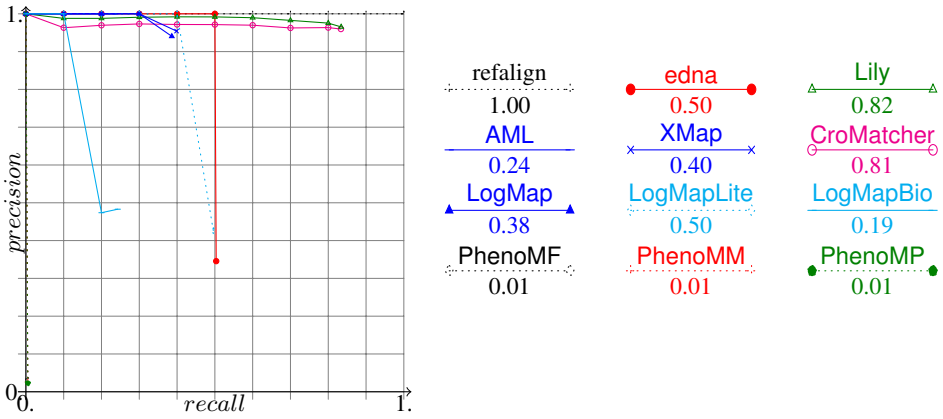


Fig. 3. Precision/recall plots on biblio.

very well with XMap even improving its score. This measure which is supposed to reward systems able to provide accurate confidence values is beneficial to these faithful systems.

Speed Beside LogMapBio which uses alignment repositories on the web to find matches, all matchers do the task in less than 40 min (for biblio and 12h for film). There is still a large discrepancy between matchers concerning the time spent from less than two minutes for LogMapLite, AML and XMap to nearly two hours for LogMapBio (on biblio).

	biblio			film		
Matcher	time	stdev	F-m./s.	time	stdev	F-m./s.
AML	120	±13%	.32	183	±1%	.17
CroMatcher	1100	±3%	.08		NaN	
Lily	2211	±1%	.04	2797	±1%	.03
LogMap	194	±5%	.28	40609	±33%	.00
LogMapLt	96	±10%	.48	116	±0%	.44
PhenoMF	1632	±8%	.00	1798	±7%	.00
PhenoMM	1743	±7%	.00	1909	±7%	.00
PhenoMP	1833	±7%	.00	1835	±7%	.00
XMap	123	±9%	.46	2981	±21%	.02
LogMapBio	54439	±6%	.00	193763419	±32%	.00

Table 5. Aggregated benchmark results: Time (in second), standard deviation on time and points of F-measure per second spent on the three data sets.

Table 5 provides the average time, time standard deviation and 1/100e F-measure point provided per second by matchers. The F-measure point provided per second shows

that efficient matchers are, like two years ago, LogMapLite and XMap followed by AML and LogMap. The correlation between time and F-measure only holds for these systems.

Time taken by systems is, for most of them, far larger on film than biblio and the deviation from average increased as well.

3.3 Conclusions

This year, there is no increase or decrease of the performance of the best matchers which are roughly the same as previous years. Precision is still preferred to recall by the best systems. It seems difficult to other matchers to catch up both in terms of robustness and performances. This confirms the trend observed last year.

4 Anatomy

The anatomy test case confronts matchers with a specific type of ontologies from the biomedical domain. We focus on two fragments of biomedical ontologies which describe the human anatomy[5] and the anatomy of the mouse[6]. This data set has been used since 2007 with some improvements over the years.

4.1 Experimental Setting

We conducted experiments by executing each system in its standard setting and we compare precision, recall, F-measure and recall+. The measure recall+ indicates the amount of detected non-trivial correspondences. The matched entities in a non-trivial correspondence do not have the same normalised label. The approach that generates only trivial correspondences is depicted as baseline StringEquiv in the following section.

We ran the systems on a server with 3.46 GHz (6 cores) and 8GB RAM allocated to each matching system. Further, we used the SEALS client to execute our evaluation. However, we slightly changed the way precision and recall are computed, i.e., the results generated by the SEALS client vary in some cases by 0.5% compared to the results presented below. In particular, we removed trivial correspondences in the oboInOwl namespace like:

```
http://...oboInOwl#Synonym = http://...oboInOwl#Synonym
```

as well as correspondences expressing relations different from equivalence. Using the Pellet reasoner we also checked whether the generated alignment is coherent, i.e., that there are no unsatisfiable classes when the ontologies are merged with the alignment.

4.2 Results

Table 6 reports all the 13 participating systems that could generate an alignment. As previous years some of the systems participated with different versions. LogMap participated with LogMap, LogMapBio and a lightweight version LogMapLite that uses only some core components. Similarly, DKP-AOM also participated with two versions, DKP-AOM and DKP-AOM-Lite. Several systems participate in the anatomy track for the first time. These are Alin, FCA_Map, DLPHOM and LYAM. There are also systems having been participant for several years in a row. LogMap is a constant participant since 2011.

[5] http://www.cancer.gov/cancertopics/cancerlibrary/terminologyresources/

[6] http://www.informatics.jax.org/searches/AMA_form.shtml

AML and XMap joined the track in 2013. DKP-AOM, Lily and CroMatcher participate for the second year in a row in this track. Lily participated in the track back in 2011. CroMatcher participated in 2013 but did not produce an alignment within the given time frame. Thus, this year we have 10 different systems (not counting different versions) which generated an alignment. For more details, we refer the reader to the papers presenting the systems.

Matcher	Runtime	Size	Precision	F-measure	Recall	Recall+	Coherent
AML	47	1493	0.95	0.943	0.936	0.832	√
CroMatcher	573	1442	0.949	0.925	0.902	0.773	-
XMap	45	1413	0.929	0.896	0.865	0.647	√
LogMapBio	758	1531	0.888	0.892	0.896	0.728	√
FCA_Map	117	1361	0.932	0.882	0.837	0.578	-
LogMap	24	1397	0.918	0.88	0.846	0.593	√
LYAM	799	1539	0.863	0.869	0.876	0.682	-
Lily	272	1382	0.87	0.83	0.794	0.515	-
LogMapLite	20	1147	0.962	0.828	0.728	0.288	-
StringEquiv	-	946	0.997	0.766	0.622	0.000	-
LPHOM	1601	1555	0.709	0.718	0.727	0.497	-
Alin	306	510	0.996	0.501	0.335	0.0	√
DKP-AOM-Lite	372	207	0.99	0.238	0.135	0.0	√
DKP-AOM	379	207	0.99	0.238	0.135	0.0	√

Table 6. Comparison, ordered by F-measure, against the reference alignment, runtime is measured in seconds, the "size" column refers to the number of correspondences in the generated alignment.

Unlike the last two editions of the track when 6 systems generated an alignment in less than 100 seconds, this year only 4 of them were able to complete the alignment task in this time frame. These are AML, XMap, LogMap and LogMapLite. Similarly to the last 4 years LogMapLite has the shortest runtime, followed by LogMap, XMap and AML. Depending on the specific version of the systems, they require between 20 and 50 seconds to match the ontologies. The table shows that there is no correlation between quality of the generated alignment in terms of precision and recall and required runtime. This result has also been observed in previous OAEI campaigns.

The table also shows the results for precision, recall and F-measure. In terms of F-measure, the top 5 ranked systems are AML, CroMatcher, XMap, LogMapBio and FCA_Map. LogMap is sixth with a F-measure very close to FCA_Map. All the long-term participants in the track showed comparable results (in term or F-measure) to their last year's results and at least as good as the results of the best systems in OAEI 2007-2010. LogMap and XMap generated the same number of correspondences in their alignment (XMap generated one correspondence more). AML and LogMapBio generated a slightly different number—16 correspondences more for AML and 18 less for LogMapBio.

The results for the DKP-AOM systems are identical this year; by contrast, last year the lite version performed significantly better in terms of the observed measures. While Lily had improved its 2015 results in comparison to 2011 (precision: from 0.814 to

0.870, recall: from 0.734 to 0.793, and F-measure: from 0.772 to 0.830), this year it performed similarly to last year. CroMatcher improved its results in comparison to last year. Out of all systems participating in the anatomy track CroMatcher showed the largest improvement in the observed measures in comparison to its values from the previous edition of the track.

Comparing the F-measures of the new systems, FCA_Map (0.882) scored very close to one of the tracks' long-term participants LogMap. Another of the new systems—LYAM—also achieved a good F-measure (0.869) which ranked sixth. As for the other two systems, LPHOM achieved a slightly lower F-measure than the baseline (StringEquiv) whereas Alin was considerably below the baseline.

This year, 9 out of 13 systems achieved an F-measure higher than the baseline which is based on (normalised) string equivalence (StringEquiv in the table). This is a slightly better result (percentage-wise) than last year's (9 out of 15) and similar to 2014's (7 out of 10). Two of the new participants in the track and the two DKP-AOM systems achieved an F-measure lower than the baseline. LPHOM scored under the StringEquiv baseline but at the same time it is the system that produced the highest number of correspondences. Its precision is significantly lower than the other three systems which scored under the baseline and generated only trivial correspondences.

This year seven systems produced coherent alignments which is comparable to the last two years, when 7 out of 15 and 5 out of 10 systems achieved this. From the five best systems only FCA_Map produced an incoherent alignment.

4.3 Conclusions

Like for OAEI in general, the number of participating systems in the anatomy track this year was lower than in 2015 and 2013 but higher than in 2014, and there was a combination of newly-joined systems and long-term participants.

The systems that participated in the previous edition scored similarly to their previous results, indicating that no substantial developments were made with regard to this track. Of the newly-joined systems, (FCA_Map and LYAM) ranked 4th and 6th with respect to the F-measure.

5 Conference

The conference test case requires matching several moderately expressive ontologies from the conference organisation domain.

5.1 Test data

The data set consists of 16 ontologies in the domain of organising conferences. These ontologies have been developed within the OntoFarm project[7].

The main features of this test case are:

- *Generally understandable domain.* Most ontology engineers are familiar with organising conferences. Therefore, they can create their own ontologies as well as evaluate the alignments among their concepts with enough erudition.
- *Independence of ontologies.* Ontologies were developed independently and based on different resources, they thus capture the issues in organising conferences from different points of view and with different terminologies.

[7] http://owl.vse.cz:8080/ontofarm/

– *Relative richness in axioms.* Most ontologies were equipped with OWL DL axioms of various kinds; this opens a way to use semantic matchers.

Ontologies differ in their numbers of classes and properties, in expressivity, but also in underlying resources.

5.2 Results

We provide results in terms of F-measure, comparison with baseline matchers and results from previous OAEI editions and precision/recall triangular graph based on sharp reference alignments. This year we can provide comparison between OAEI editions of results based on the uncertain version of reference alignment and on violations of consistency and conservativity principles.

Evaluation based on sharp reference alignments We evaluated the results of participants against blind reference alignments (labelled as *rar2*). This includes all pairwise combinations between 7 different ontologies, i.e., 21 alignments.

These reference alignments have been made in two steps. First, we have generated them as a transitive closure computed on the original reference alignments. In order to obtain a coherent result, conflicting correspondences, i.e., those causing unsatisfiability, have been manually inspected and removed by evaluators. The resulting reference alignments are labelled as *ra2*. Second, we detected violations of conservativity using the approach from [39] and resolved them by an evaluator. The resulting reference alignments are labelled as *rar2*. As a result, the degree of correctness and completeness of the new reference alignments is probably slightly better than for the old one. However, the differences are relatively limited. Whereas the new reference alignments are not open, the old reference alignments (labeled as *ra1* on the conference web page) are available. These represent close approximations of the new ones.

Table 7 shows the results of all participants with regard to the reference alignment *rar2*. $F_{0.5}$-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. We employed two baseline matchers. edna (string edit distance matcher) is used within the benchmark test case and with regard to performance it is very similar as the previously used *baseline2* in the conference track; StringEquiv is used within the anatomy test case. This year these baselines divide matchers into two performance groups. The first group consists of matchers (CroMatcher, AML, LogMap, XMap, LogMapBio, FCA_Map, DKP-AOM, NAISC and LogMapLite) having better (or the same) results than both baselines in terms of highest average F_1-measure. Other matchers (Lily, LPHOM, Alin and LYAM) performed worse than both baselines. The performance of all matchers (except LYAM) regarding their precision, recall and F_1-measure is visualised in Figure 4. Matchers are represented as squares or triangles. Baselines are represented as circles.

Further, we evaluated the performance of matchers separately on classes and properties. We compared the position of tools within overall performance groups and within only classes and only properties performance groups. We observed that while the position of matchers changed slightly in overall performance groups in comparison with

Matcher	Prec.	$F_{0.5}$-m.	F_1-m.	F_2-m.	Rec.	Inc.Align.	Conser.V.	Consist.V.
CroMatcher	0.74	0.72	0.69	0.67	0.65	8	98	25
AML	0.78	0.74	0.69	0.65	0.62	0	52	0
LogMap	0.77	0.72	0.66	0.6	0.57	0	30	0
XMap	0.8	0.73	0.65	0.59	0.55	0	23	0
LogMapBio	0.72	0.67	0.61	0.56	0.53	0	30	0
FCA_Map	0.71	0.65	0.59	0.53	0.5	12	46	150
DKP-AOM	0.76	0.68	0.58	0.51	0.47	0	35	0
NAISC	0.77	0.67	0.57	0.49	0.45	20	321	701
edna	*0.74*	*0.66*	*0.56*	*0.49*	*0.45*			
LogMapLite	0.68	0.62	0.56	0.5	0.47	6	99	81
StringEquiv	*0.76*	*0.65*	*0.53*	*0.45*	*0.41*			
Lily	0.54	0.53	0.52	0.51	0.5	13	148	167
LPHOM	0.69	0.57	0.46	0.38	0.34	0	0	0
Alin	0.87	0.59	0.4	0.3	0.26	0	0	0
LYAM	0.4	0.31	0.23	0.18	0.16	1	75	3

Table 7. The highest average $F_{[0.5|1|2]}$-measure and their corresponding precision and recall for each matcher with its F_1-optimal threshold (ordered by F_1-measure). Inc.Align. means number of incoherent alignments. Conser.V. means total number of all conservativity principle violations. Consist.V. means total number of all consistency principle violations.

only classes performance groups, a couple of matchers (DKP-AOM and FCA_Map) worsen their position from overall performance groups with regard to their position in only properties performance groups due to the fact that they do not match properties at all (Alin and Lily also fall into this category). More details about these evaluation modalities are on the conference web page.

Comparison with previous years with regard to rar2 Seven matchers also participated in this test case in OAEI 2015. The largest improvement was achieved by CroMatcher (precision increased from .57 to .74 and recall increased from .47 to .65).

Evaluation based on uncertain version of reference alignments The confidence values of all matches in the sharp reference alignments for the conference track are all 1.0. For the uncertain version of this track, the confidence value of a match has been set equal to the percentage of a group of people who agreed with the match in question (this uncertain version is based on the reference alignment labeled *ra1*). One key thing to note is that the group was only asked to validate matches that were already present in the existing reference alignments – so some matches had their confidence value reduced from 1.0 to a number near 0, but no new match was added.

There are two ways that we can evaluate matchers according to these "uncertain" reference alignments, which we refer to as *discrete* and *continuous*. The discrete evaluation considers any match in the reference alignment with a confidence value of 0.5 or greater to be fully correct and those with a confidence less than 0.5 to be fully incorrect. Similarly, a matcher's match is considered a "yes" if the confidence value is greater than or equal to the matcher's threshold and a "no" otherwise. In essence, this is the same as the "sharp" evaluation approach, except that some matches have been removed because less than half of the crowdsourcing group agreed with them. The continuous evaluation

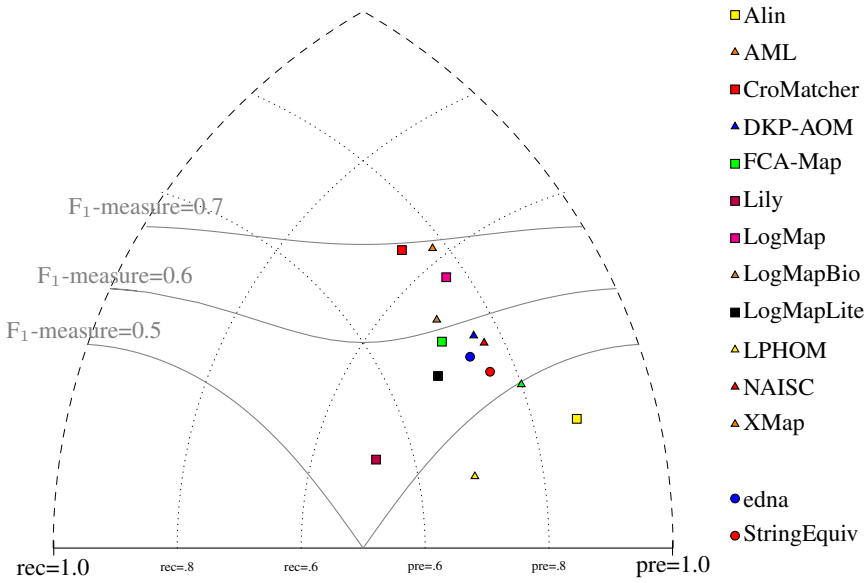


Fig. 4. Precision/recall triangular graph for the conference test case. Dotted lines depict level of precision/recall while values of F_1-measure are depicted by areas bordered by corresponding lines F_1-measure=0.[5|6|7].

strategy penalises a matcher more if it misses a match on which most people agree than if it misses a more controversial match. For instance, if $A \equiv B$ with a confidence of 0.85 in the reference alignment and a matcher gives that correspondence a confidence of 0.40, then that is counted as $0.85 \times 0.40 = 0.34$ of a true positive and $0.85–0.40 = 0.45$ of a false negative.

Out of the 13 matchers, three (DKP-AOM, FCA-Map and LogMapLite) use 1.0 as the confidence values for all matches they identify. Two of the remaining ten (Alin and CroMatcher) have some variation in confidence values, though the majority are 1.0. The rest of the systems have a fairly wide variation of confidence values. Last year, the majority of these values were near the upper end of the [0,1] range. This year we see much more variation in the average confidence values. For example, LopMap's confidence values range from 0.29 to 1.0 and average 0.78 whereas Lily's range from 0.22 to 0.41 with an average of 0.33.

Discussion When comparing the performance of the matchers on the uncertain reference alignments versus that on the sharp version, we see that in the discrete case all matchers performed slightly better. Improvement in F-measure ranged from 1 to 8 percentage points over the sharp reference alignment. This was driven by increased recall, which is a result of the presence of fewer "controversial" matches in the uncertain version of the reference alignment.

The performance of most matchers is similar regardless of whether a discrete or continuous evaluation methodology is used (provided that the threshold is optimised to achieve the highest possible F-measure in the discrete case). The primary exceptions

Matcher	Sharp			Discrete			Continuous		
	Prec.	F_1-m.	Rec.	Prec.	F_1-m.	Rec.	Prec.	F_1-m.	Rec.
Alin	0.89	0.40	0.26	0.89	0.48	0.33	0.89	0.48	0.33
AML	0.84	0.74	0.66	0.79	0.78	0.77	0.80	0.77	0.74
CroMatcher	0.79	0.73	0.68	0.71	0.74	0.77	0.72	0.74	0.77
DKP-AOM	0.82	0.62	0.50	0.78	0.67	0.59	0.78	0.69	0.61
FCA-Map	0.75	0.61	0.52	0.71	0.66	0.61	0.69	0.65	0.61
Lily	0.59	0.56	0.53	0.59	0.57	0.56	0.59	0.32	0.22
LogMap	0.82	0.69	0.59	0.78	0.73	0.68	0.80	0.67	0.57
LogMapBio	0.77	0.65	0.56	0.73	0.68	0.64	0.75	0.62	0.53
LogMapLite	0.73	0.59	0.50	0.73	0.67	0.62	0.72	0.67	0.63
LPHOM	0.76	0.47	0.34	0.81	0.59	0.46	0.48	0.47	0.47
Light YAM++	0.38	0.22	0.15	0.35	0.24	0.18	0.09	0.15	0.38
NAISC	0.85	0.61	0.47	0.87	0.69	0.57	0.34	0.45	0.68
XMap	0.85	0.68	0.57	0.81	0.73	0.67	0.83	0.74	0.67

Table 8. F-measure, precision, and recall of the different matchers when evaluated using the sharp (*ra1*), discrete uncertain and continuous uncertain metrics.

to this are Lily and NAISC. These matchers perform significantly worse when evaluated using the continuous version of the metrics. In Lily's case, this is because it assigns very low confidence values to some matches in which the labels are equivalent strings, which many crowdsourcers agreed with unless there was a compelling technical reason not to. This hurts recall, but using a low threshold value in the discrete version of the evaluation metrics "hides" this problem. NAISC has the opposite issue: it assigns relatively high confidence values to some matches that most people disagree with, such as "Assistant" and "Listener" (confidence value of 0.89). This hurts precision in the continuous case, but is taken care of by using a high threshold value (1.0) in the discrete case.

Seven matchers from this year also participated last year, and thus we are able to make some comparisons over time. The F-measures of all matchers either held constant or improved when evaluated against the uncertain reference alignments. Most matchers made modest gains (in the neighborhood of 1 to 6 percentage points). CroMatcher made the largest improvement, and it is now the second-best matcher when evaluated in this way. AgreementMakerLight remains the top performer.

Perhaps more importantly, the difference in the performance of most matchers between the discrete and continuous evaluation has shrunk between this year and last year. This is an indication that more matchers are providing confidence values that reflect the disagreement of humans on various matches.

Evaluation based on violations of consistency and conservativity principles We performed evaluation based on detection of conservativity and consistency violations [39,40]. The consistency principle states that correspondences should not lead to unsatisfiable classes in the merged ontology; the conservativity principle states that correspondences should not introduce new semantic relationships between concepts from one of the input ontologies.

Table 7 summarises statistics per matcher. The table shows the number of unsatisfiable TBoxes after the ontologies are merged (Inc. Align.), the total number of all

conservativity principle violations within all alignments (Conser.V.) and the total number of all consistency principle violations (Consist.V.).

Seven tools (Alin, AML, DKP-AOM, LogMap, LogMapBio, LPHOM and XMap) have no consistency principle violations (in comparison to five last year) and one tool (LYAM) generated only one incoherent alignment. There are two tools (Alin, LPHOM) that have no conservativity principle violations, and four more that have an average of only one conservativity principle violation (XMap, LogMap, LogMapBio and DKP-AOM). We should note that these conservativity principle violations can be "false positives" since the entailment in the aligned ontology can be correct although it was not derivable in the single input ontologies.

In conclusion, this year eight matchers performed better than both baselines on reference alignments which is not only consistent but also conservative. Further, this year seven matchers generated coherent alignments (against five matchers last year and four matchers the year before). This confirms the trend that increasingly matchers generate coherent alignments. Based on the uncertain reference alignments, more matchers are providing confidence values that reflect the disagreement of humans on various matches.

6 Large biomedical ontologies (largebio)

The largebio test case requires to match the large and semantically rich biomedical ontologies FMA, SNOMED-CT, and NCI, which contain 78,989, 306,591 and 66,724 classes, respectively.

6.1 Test data

The test case has been split into three matching problems: FMA-NCI, FMA-SNOMED and SNOMED-NCI. Each matching problem has been further divided in 2 tasks involving differently sized fragments of the input ontologies: small overlapping fragments versus whole ontologies (FMA and NCI) or large fragments (SNOMED-CT).

The UMLS Metathesaurus [5] has been selected as the basis for reference alignments. UMLS is currently the most comprehensive effort for integrating independently-developed medical thesauri and ontologies, including FMA, SNOMED-CT, and NCI.

Although the standard UMLS distribution does not directly provide alignments (in the sense of [21]) between the integrated ontologies, it is relatively straightforward to extract them from the information provided in the distribution files (see [25] for details).

It has been noticed, however, that although the creation of UMLS alignments combines expert assessment and auditing protocols they lead to a significant number of logical inconsistencies when integrated with the corresponding source ontologies [25].

Since alignment coherence is an aspect of ontology matching that we aim to promote, in previous editions we provided coherent reference alignments by refining the UMLS mappings using the Alcomo (alignment) debugging system [31], LogMap's (alignment) repair facility [24], or both [26].

However, concerns were raised about the validity and fairness of applying automated alignment repair techniques to make reference alignments coherent [35]. It is clear that using the original (incoherent) UMLS alignments would be penalising to ontology matching systems that perform alignment repair. However, using automatically repaired alignments would penalise systems that do not perform alignment repair and

also systems that employ a repair strategy that differs from that used on the reference alignments [35].

Thus, as of the 2014 edition, we arrived at a compromising solution that should be fair to all ontology matching systems. Instead of repairing the reference alignments as normal, by removing correspondences, we flagged the *incoherence-causing correspondences* in the alignments by setting the relation to *"?"* (unknown). These "?" correspondences will neither be considered as positive nor as negative when evaluating the participating ontology matching systems, but will simply be ignored. This way, systems that do not perform alignment repair are not penalised for finding correspondences that (despite causing incoherences) may or may not be correct, and systems that do perform alignment repair are not penalised for removing such correspondences.

To ensure that this solution was as fair as possible to all alignment repair strategies, we flagged as unknown all correspondences suppressed by any of Alcomo, LogMap or AML [**?**], as well as all correspondences suppressed from the reference alignments of last year's edition (using Alcomo and LogMap combined). Note that, we have used the (incomplete) repair modules of the above mentioned systems.

The flagged UMLS-based reference alignment for the OAEI 2016 campaign is summarised in Table 9.

Reference alignment	"=" corresp.	"?" corresp.
FMA-NCI	2,686	338
FMA-SNOMED	6,026	2,982
SNOMED-NCI	17,210	1,634

Table 9. Respective sizes of reference alignments

6.2 Evaluation setting, participation and success

We have run the evaluation on a Ubuntu Laptop with an Intel Core i7-4600U CPU @ 2.10GHz x 4 and allocating 15Gb of RAM. Precision, recall and F-measure have been computed with respect to the UMLS-based reference alignment. Systems have been ordered in terms of F-measure.

This year, out of the 21 systems participating in OAEI 2016, 13 were registered to participate in the largebio track, and 11 of these were able to cope with at least one of the largebio tasks within a 2 hour time frame. However, only 6 systems were able to complete more than one task, and only 4 systems completed all 6 tasks in this time frame.

6.3 Background knowledge

Regarding the use of background knowledge, LogMap-Bio uses BioPortal as a mediating ontology provider, that is, it retrieves from BioPortal the most suitable top-10 ontologies for the matching task.

LogMap uses normalisations and spelling variants from the general (biomedical) purpose UMLS Lexicon.

AML has three sources of background knowledge which can be used as mediators between the input ontologies: the Uber Anatomy Ontology (Uberon), the Human Disease Ontology (DOID) and the Medical Subject Headings (MeSH).

System	FMA-NCI		FMA-SNOMED		SNOMED-NCI		Average	#
	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6		
LogMapLite	1	10	2	18	8	18	10	6
AML	35	72	98	166	537	376	214	6
LogMap	10	80	60	433	177	699	243	6
LogMapBio	1,712	1,188	1,180	2,156	3,757	4,322	2,386	6
XMap	17	116	54	366	267	-	164	5
FCA-Map	236	-	1,865	-	-	-	1,051	2
Lily	699	-	-	-	-	-	699	1
LYAM	1,043	-	-	-	-	-	1,043	1
DKP-AOM	1,547	-	-	-	-	-	1,547	1
DKP-AOM-Lite	1,698	-	-	-	-	-	1,698	1
Alin	5,811	-	-	-	-	-	5,811	1
# Systems	**11**	**6**	**5**	**5**	**5**	**4**	**1,351**	**36**

Table 10. System runtimes (s) and task completion.

XMap uses synonyms provided by the UMLS Metathesaurus. Note that matching systems using UMLS Metathesaurus as background knowledge will have a *notable advantage* since the largebio reference alignment is also based on the UMLS Metathesaurus.

6.4 Alignment coherence

Together with precision, recall, F-measure and run times we have also evaluated the coherence of alignments. We report (1) the number of unsatisfiabilities when reasoning with the input ontologies together with the computed alignments, and (2) the ratio of unsatisfiable classes with respect to the size of the union of the input ontologies.

We have used the OWL 2 reasoner HermiT [33] to compute the number of unsatisfiable classes. For the cases in which HermiT could not cope with the input ontologies and the alignments (in less than 2 hours) we have provided a lower bound on the number of unsatisfiable classes (indicated by $\geq$) using the OWL 2 EL reasoner ELK [27].

In this OAEI edition, only three distinct systems have shown alignment repair facilities: AML, LogMap and its LogMap-Bio variant, and XMap (which reuses the repair techniques from Alcomo [31]). Tables 11-12 (see last two columns) show that even the most precise alignment sets may lead to a huge number of unsatisfiable classes. This proves the importance of using techniques to assess the coherence of the generated alignments if they are to be used in tasks involving reasoning. We encourage ontology matching system developers to develop their own repair techniques or to use state-of-the-art techniques such as Alcomo [31], the repair module of LogMap (LogMap-Repair) [24] or the repair module of AML [**?**], which have worked well in practice [26,22].

6.5 Runtimes and task completion

Table 10 shows which systems were able to complete each of the matching tasks in less than 24 hours and the required computation times. Systems have been ordered with respect to the number of completed tasks and the average time required to complete them. Times are reported in seconds.

The last column reports the number of tasks that a system could complete. For example, 8 system were able to complete all six tasks. The last row shows the number

Task 1: small FMA and NCI fragments							
System	Time (s)	# Corresp.	Scores			Incoherence	
			Prec.	F-m.	Rec.	Unsat.	Degree
XMap[8]	17	2,649	0.98	0.94	0.90	2	0.019%
FCA-Map	236	2,834	0.95	0.94	0.92	4,729	46.0%
AML	35	2,691	0.96	0.93	0.90	2	0.019%
LogMap	10	2,747	0.95	0.92	0.90	2	0.019%
LogMapBio	1,712	2,817	0.94	0.92	0.91	2	0.019%
LogMapLite	1	2,483	0.97	0.89	0.82	2,045	19.9%
Average	1,164	2,677	0.85	0.80	0.78	2,434	23.7%
LYAM	1,043	3,534	0.72	0.80	0.89	6,880	66.9%
Lily	699	3,374	0.60	0.66	0.72	9,273	90.2%
Alin	5,811	1,300	1.00	0.62	0.46	0	0.0%
DKP-AOM-Lite	1,698	2,513	0.65	0.61	0.58	1,924	18.7%
DKP-AOM	1,547	2,513	0.65	0.61	0.58	1,924	18.7%

Task 2: whole FMA and NCI ontologies							
System	Time (s)	# Corresp.	Scores			Incoherence	
			Prec.	F-m.	Rec.	Unsat.	Degree
XMap[8]	116	2,681	0.90	0.87	0.85	9	0.006%
AML	72	2,968	0.84	0.85	0.87	10	0.007%
LogMap	80	2,693	0.85	0.83	0.80	9	0.006%
LogMapBio	1,188	2,924	0.82	0.83	0.84	9	0.006%
Average	293	2,948	0.82	0.82	0.84	5,303	3.6%
LogMapLite	10	3,477	0.67	0.74	0.82	26,478	18.1%

Table 11. Results for the FMA-NCI matching problem.

of systems that could finish each of the tasks. The tasks involving SNOMED were also harder with respect to both computation times and the number of systems that completed the tasks.

6.6 Results for the FMA-NCI matching problem

Table 11 summarises the results for the tasks in the FMA-NCI matching problem.

XMap and FCA-Map achieved the highest F-measure in Task 1; XMap and AML in Task 2. Note however that the use of background knowledge based on the UMLS Metathesaurus has an important impact in the performance of XMap[8]. The use of background knowledge led to an improvement in recall from LogMap-Bio over LogMap in both tasks, but this came at the cost of precision, resulting in the two variants of the system having identical F-measures.

Note that the effectiveness of the systems decreased from Task 1 to Task 2. One reason for this is that with larger ontologies there are more plausible mapping candidates, and thus it is harder to attain both a high precision and a high recall. Another reason is that the very scale of the problem constrains the matching strategies that systems can

[8] Uses background knowledge based on the UMLS Metathesaurus which is the base of the largebio reference alignments.

Task 3: small FMA and SNOMED fragments							
System	Time (s)	# Corresp.	Scores			Incoherence	
			Prec.	F-m.	Rec.	Unsat.	Degree
XMap[8]	54	7,311	0.99	0.91	0.85	0	0.0%
FCA-Map	1,865	7,649	0.94	0.86	0.80	14,603	61.8%
AML	98	6,554	0.95	0.82	0.73	0	0.0%
LogMapBio	1,180	6,357	0.94	0.80	0.70	1	0.004%
LogMap	60	6,282	0.95	0.80	0.69	1	0.004%
Average	543	5,966	0.96	0.76	0.66	2,562	10.8%
LogMapLite	2	1,644	0.97	0.34	0.21	771	3.3%

Task 4: whole FMA ontology with SNOMED large fragment							
System	Time (s)	# Corresp.	Scores			Incoherence	
			Prec.	F-m.	Rec.	Unsat.	Degree
XMap[8]	366	7,361	0.97	0.90	0.84	0	0.0%
AML	166	6,571	0.88	0.77	0.69	0	0.0%
LogMap	433	6,281	0.84	0.72	0.63	0	0.0%
LogMapBio	2,156	6,520	0.81	0.71	0.64	0	0.0%
Average	627	5,711	0.87	0.69	0.60	877	0.4%
LogMapLite	18	1,822	0.85	0.34	0.21	4,389	2.2%

Table 12. Results for the FMA-SNOMED matching problem.

employ: AML for example, foregoes its matching algorithms that are computationally more complex when handling very large ontologies, due to efficiency concerns.

The size of Task 2 proves a problem for several systems, which were unable to complete it within the allotted time: FCA-Map, LYAM, LiLy, Alin, DKP-AOM-Lite and DKP-AOM.

6.7 Results for the FMA-SNOMED matching problem

Table 12 summarises the results for the tasks in the FMA-SNOMED matching problem.

XMap produced the best results in terms of both recall and F-measure in Task 3 and Task 4, but again, we must highlight that it uses background knowledge based on the UMLS Metathesaurus. Among the other systems, FCA-Map and AML achieved the highest F-measure in Tasks 3 and 4, respectively.

Overall, the quality of the results was lower than that observed in the FMA-NCI matching problem, as the matching problem is considerably larger. Indeed, several systems were unable to complete even the smaller Task 3 within the allotted time: LYAM, LiLy, Alin, DKP-AOM-Lite and DKP-AOM.

Like in the FMA-NCI matching problem, the effectiveness of all systems decreases as the ontology size increases from Task 3 to Task 4; FCA-Map could complete the former but not the latter.

6.8 Results for the SNOMED-NCI matching problem

Table 13 summarises the results for the tasks in the SNOMED-NCI matching problem.

AML achieved the best results in terms of both recall and F-measure in Tasks 5 and 6, while LogMap and AML achieved the best results in terms of precision in Tasks 5 and 6, respectively.

Task 5: small SNOMED and NCI fragments							
System	Time (s)	# Corresp.	Scores			Incoherence	
			Prec.	F-m.	Rec.	Unsat.	Degree
AML	537	13,584	0.90	0.80	0.71	0	0.0%
LogMap	177	12,371	0.92	0.77	0.66	0	0.0%
LogMapBio	3,757	12,960	0.90	0.77	0.68	0	0.0%
Average	949	13,302	0.91	0.75	0.64	≥12,090	≥16.1%
XMap[8]	267	16,657	0.91	0.70	0.56	0	0.0%
LogMapLite	8	10,942	0.89	0.69	0.57	≥60,450	≥80.4%

Task 6: whole NCI ontology with SNOMED large fragment							
System	Time (s)	# Corresp.	Scores			Incoherence	
			Prec.	F-m.	Rec.	Unsat.	Degree
AML	376	13,175	0.90	0.77	0.67	≥2	≥0.001%
LogMapBio	4,322	13,477	0.84	0.72	0.64	≥6	≥0.003%
Average	1,353	12,942	0.85	0.72	0.62	37,667	19.9%
LogMap	699	12,222	0.87	0.71	0.60	≥4	≥0.002%
LogMapLite	18	12,894	0.80	0.66	0.57	≥150,656	≥79.5%

Table 13. Results for the SNOMED-NCI matching problem.

The overall performance of the systems was lower than in the FMA-SNOMED case, as this test case is even larger. As such, LiLy, DKP-AOM-Lite, DKP-AOM, FCA-Map, Alin and LYAM could not complete even the smaller Task 5 within 2 hours.

As in the previous matching problems, effectiveness decreases as the ontology size increases, and XMap completed Task 5 but failed to complete Task 6 within the given time frame.

Unlike in the FMA-NCI and FMA-SNOMED matching problems, the use of the UMLS Metathesaurus did not positively impact the performance of XMap, which obtained lower results than expected.

7 Disease and Phenotype Track (phenotype)

The Pistoia Alliance Ontologies Mapping project team[9] has organised this track based on a real use case where it is required to find alignments between disease and phenotype ontologies. Specifically, the selected ontologies are the Human Phenotype Ontology (HPO), the Mammalian Phenotype Ontology (MP), the Human Disease Ontology (DOID), and the Orphanet and Rare Diseases Ontology (ORDO).

7.1 Test data

There are two tasks in this track which comprise the pairwise alignment of:

- Human Phenotype Ontology (HPO) to Mammalian Phenotype Ontology (MP), and
- Human Disease Ontology (DOID) to Orphanet and Rare Diseases Ontology (ORDO).

The first task is important for translational science, since mammal model animals such as mice are widely used to study human diseases and their underlying genetics.

[9] http://www.pistoiaalliance.org/projects/ontologies-mapping/

Mapping human phenotypes to mammalian phenotypes greatly facilitates the extrapolation from model animals to humans.

The second task is critical to ensure interoperability between two disease ontologies: the more generic DOID and the more specific ORDO, in the domain of rare human diseases. These are fundamental for understanding how genetic variation can cause disease.

Currently, mappings between these ontologies are mostly curated by bioinformatics and disease experts who would benefit from the use of automated ontology matching algorithms into their workflows.

7.2 Evaluation setting

We have run the evaluation on a Ubuntu Laptop with an Intel Core i7-4600U CPU @ 2.10GHz x 4, allocating 15Gb of RAM.

In the OAEI 2016 phenotype track, 11 out of the 21 participating OAEI 2016 systems have been able to cope with at least one of the tasks within a 24 hour time frame.

7.3 Evaluation criteria

Systems have been evaluated according to the following criteria:

- Semantic precision and recall with respect to silver standards automatically generated by voting based on the outputs of all participating systems (we have used vote=2 and vote=3)[10].
- Semantic recall with respect to manually generated correspondences for three areas (carbohydrate, obesity and breast cancer).
- Manual assessment of a subset of the generated correspondences, specially the ones that are not suggested by other systems, i.e., unique mapping.

We have used the OWL 2 reasoner HermiT to calculate the semantic precision and recall. For example, a positive hit will mean that a mapping in the reference has been (explicitly) included in the output mappings or it can be inferred using reasoning from the input ontologies and the output mappings. The use of semantic values for precision and recall also allowed us to provide a fair comparison for the systems PhenoMF, PhenoMM and PhenoMP which discover many subsumption mappings that are not explicitly in the silver standards but may still be valid, i.e., inferred.

7.4 Use of background knowledge

LogMapBio uses BioPortal as a mediating ontology provider, that is, it retrieves from BioPortal the most suitable top-10 ontologies for the matching task.

LogMap uses normalisations and spelling variants from the general (biomedical) purpose UMLS Lexicon.

AML has three sources of background knowledge which can be used as mediators between the input ontologies: the Uber Anatomy Ontology (Uberon), the Human Disease Ontology (DOID) and the Medical Subject Headings (MeSH). Additionally, for the HPO-MP test case, it uses the logical definitions of both ontologies, which define

[10] When there are several systems of the same family, only one of them votes for avoiding bias. There still can be some bias through systems exploiting the same resource, e.g., UMLS.

OM algorithm	Track Task	Mappings	Precision Silver 2	Recall Silver 2	F-Score Silver 2	Sum F Scores Silver 2	Precision Silver 3	Recall Silver 3	F-Score Silver 3	Sum F-Scores Silver 3
AML	**HP-MP**	**1755**	**0.9305**	**0.7998**	**0.8602**	**1.7684**	**0.8536**	**0.9446**	**0.8968**	**1.7714**
AML	**DOID-ORDO**	**2098**	**0.8532**	**0.9708**	**0.9082**		**0.7784**	**0.9981**	**0.8747**	
DiSMatch	HP-MP	644	0.5481	0.2058	0.2993	0.3818	0.4550	0.1971	0.2750	0.3423
DiSMatch	DOID-ORDO	335	0.2269	0.0505	0.0825		0.1910	0.0408	0.0673	
FCA_Map	**HP-MP**	**1590**	**0.9836**	**0.7543**	**0.8539**	**1.8162**	**0.9421**	**0.9244**	**0.9332**	**1.8706**
FCA_Map	**DOID-ORDO**	**1803**	**0.9662**	**0.9586**	**0.9624**		**0.8880**	**0.9926**	**0.9374**	
LogMap	**HP-MP**	**2011**	**0.9354**	**0.9125**	**0.9238**	**1.8372**	**0.7732**	**0.9729**	**0.8617**	**1.7828**
LogMap	**DOID-ORDO**	**1667**	**0.9520**	**0.8779**	**0.9134**		**0.9052**	**0.9375**	**0.9211**	
LogMapBio	**HP-MP**	**2151**	**0.9182**	**0.9315**	**0.9248**	**1.8338**	**0.7545**	**0.9824**	**0.8535**	**1.7580**
LogMapBio	**DOID-ORDO**	**1804**	**0.9202**	**0.8980**	**0.9090**		**0.8642**	**0.9487**	**0.9045**	
LogMapLite	HP-MP	667	1.0000	0.3449	0.5129	1.2284	0.9985	0.4471	0.6176	1.3814
LogMapLite	DOID-ORDO	1000	0.9930	0.5592	0.7155		0.9890	0.6221	0.7638	
LYAM	HP-MP	381	0.4068	0.0685	0.1172	0.1172	0.1654	0.0359	0.0590	0.0590
LYAM	DOID-ORDO	0	0.0	0.0	0.0		0.0	0.0	0.0	
PhenoMF	**HP-MP**	**204089**	**0.7568**	**0.9164**	**0.8290**	**1.7149**	**0.6292**	**0.9452**	**0.7555**	**1.6905**
PhenoMF	**DOID-ORDO**	**40612**	**0.9498**	**0.8301**	**0.8859**		**0.9472**	**0.9233**	**0.9351**	
PhenoMM	HP-MP	198140	0.7708	0.9051	0.8326	0.8326	0.6441	0.9389	0.7641	0.7641
PhenoMM	DOID-ORDO	0	0.0	0.0	0.0		0.0	0.0	0.0	
PhenoMP	HP-MP	169660	0.7828	0.5784	0.6653	0.6653	0.6391	0.5076	0.5658	0.5658
PhenoMP	DOID-ORDO	0	0.0	0.0	0.0		0.0	0.0	0.0	
XMap	HP-MP	650	1.0000	0.3332	0.4998	1.2213	1.0000	0.4351	0.6064	1.3739
XMap	DOID-ORDO	1030	0.9845	0.5693	0.7214		0.9767	0.6320	0.7674	

Table 14. Results against silver standard with vote 2 and 3.

some of their classes as being a combination of an anatomic term, i.e., a class from either FMA or Uberon, with a phenotype modifier term, i.e., a class from the Phenotypic Quality Ontology.

XMap uses synonyms provided by the UMLS Metathesaurus.

PhenoMM, PhenoMF and PhenoMP rely on different versions of the PhenomeNET[11] ontology with variable complexity.

7.5 Results

AML, FCA-Map, LogMap, LogMapBio, and PhenoMF produced the most complete results according to both the automatic and manual evaluation.

Results against the silver standards The silver standards with vote 2 and 3 for HP-MP contain 2,308 and 1,588 mappings, respectively; while for DOID-ORDO they include 1,883 and 1,617 mappings respectively. Table 14 shows the results achieved by each of the participating systems. We deliberately did not rank the systems since the silver standards only allow us to assess how systems perform in comparison with one another. On the one hand, some of the mappings in the silver standard may be erroneous (false positives), as all it takes for that is that 2 or 3 systems agree on part of the erroneous mappings they find. On the other hand, the silver standard is not complete, as there will likely be correct mappings that no system is able to find, and as we will show in the manual evaluation, there are a number of mappings found by only one system (and therefore not in the silver standard) which are correct. Nevertheless, the results with respect to the silver standards do provide some insights into the performance of the systems, which is why we highlighted in the table the 5 systems that produce results closest to the silver standards: AML, FCA-Map, LogMap, LogMapBio, and PhenoMF.

Results against manually created mappings The manually generated mappings for three areas (carbohydrate, obesity and breast cancer) include 29 mappings between HP and MP and 60 mappings between DOID and ORDO. Most of them representing subsumption relationships. Table 15 shows the results in terms of recall for each of the systems. PhenoMF, PhenoMP and PhenoMM achieve very good results for HP-MP since

[11] http://aber-owl.net/ontology/PhenomeNET

OM Algorithm	HP-MP	DOID-ORDO
PhenoMF	0.8966	0.0000
PhenoMM	0.8966	0.0000
PhenoMP	0.8276	0.0000
AML	0.7586	0.0000
LogMapBio	0.6897	0.1667
LogMap	0.6552	0.1167
FCA-Map	0.6207	0.0000
LogMapLite	0.5172	0.0000
XMAP	0.5172	0.0000
DiSMatch	0.1379	0.0333
LYAM	0.0000	0.0000

Table 15. Recall against manually created mappings.

OM algorithm	Track Task	Unique Equivalence Mappings	Precision (manual assessment)	Positive contribution (true positives)	Negative contribution (false positives)
AML	HP-MP	122	0.8667	8.63%	1.33%
DiSMatch	HP-MP	291	0.8333	19.80%	3.96%
FCA_Map	HP-MP	26	0.9615	2.04%	0.08%
LogMap	HP-MP	130	0.9330	9.90%	0.71%
LogMapLite	HP-MP	0	0.0000	0.00%	0.00%
LogMapBio	HP-MP	176	0.9330	13.40%	0.96%
LYAM++	HP-MP	226	0.7000	12.91%	5.53%
PhenoMF	HP-MP	89	1.0000	7.27%	0.00%
PhenoMM	HP-MP	85	1.0000	6.94%	0.00%
PhenoMP	HP-MP	80	1.0000	6.53%	0.00%
XMap	HP-MP	0	0.0000	0.00%	0.00%
Totals	HP-MP	**1225**		**87.42%**	**12.58%**

Table 16. Unique mappings in the HP-MP task.

OM algorithm	Track Task	Unique Equivalence Mappings	Precision (manual assessment)	Positive contribution (true positives)	Negative contribution (false positives)
AML	DOID-ORDO	308	0.8667	30.40%	4.68%
DiSMatch	DOID-ORDO	259	0.4000	11.80%	17.70%
FCA_Map	DOID-ORDO	61	0.8330	5.79%	1.16%
LogMap	DOID-ORDO	80	0.9000	8.20%	0.91%
LogMapLite	DOID-ORDO	7	0.5000	0.40%	0.40%
LogMapBio	DOID-ORDO	144	0.9667	15.85%	0.55%
LYAM++	DOID-ORDO	0	0.0000	0.00%	0.00%
PhenoMF	DOID-ORDO	3	1.0000	0.34%	0.00%
PhenoMM	DOID-ORDO	0	0.0000	0.00%	0.00%
PhenoMP	DOID-ORDO	0	0.0000	0.00%	0.00%
XMap	DOID-ORDO	16	0.5625	1.03%	0.80%
Totals	DOID-ORDO	**878**		**73.81%**	**26.19%**

Table 17. Unique mappings in the DOID-ORDO task.

they discover a large number of subsumption mappings. However, for DOID-ORDO only LogMap, LogMapBio and DisMatch discover some of the mappings in the curated set.

Manual assessment of unique mappings Tables 16 and 17 show the precision results of the manual assessment of the unique mappings generated by the participating systems. Unique mappings are correspondences that no other system (explicitly) provided in the output. We manually evaluated up to 30 mappings and we focused the assessment on unique equivalence mappings.

For example DiSMatch's output contains 291 unique mappings in the HP-MP task. The manual assessment revealed an (estimated) precision of 0.8333. In order to also take into account the number of unique mappings that a system is able to discover, Tables 16

and 17 also include the positive and negative contribution of the unique mappings with respect to the total unique mappings discovered by all participating systems.

8 MultiFarm

The MultiFarm data set [32] aims at evaluating the ability of matching systems to deal with ontologies in different natural languages. This data set results from the translation of 7 ontologies from the conference track (cmt, conference, confOf, iasted, sigkdd, ekaw and edas) into 10 languages: Arabic, Chinese, Czech, Dutch, French, German, Italian, Portuguese, Russian, and Spanish. It is composed of 55 pairs of languages (see [32] for details on how the original MultiFarm data set was generated). For each pair, taking into account the alignment direction (cmt_{en}–confOf_{de} and cmt_{de}–confOf_{en}, for instance, as two distinct matching tasks), we have 49 matching tasks. The whole data set is composed of 55×49 matching tasks.

8.1 Experimental setting

Part of the data set is used for blind evaluation. This subset includes all matching tasks involving the edas and ekaw ontologies (resulting in 55×24 matching tasks). This year, we have conducted a *minimalistic* evaluation and focused on the blind data set. Participants were able to test their systems on the available subset of matching tasks (*open evaluation*), available via the SEALS repository. The open subset covers 45×25 tasks. The open subset does not include Italian translations.

We distinguish two types of matching tasks: *(i)* those tasks where two different ontologies (cmt–confOf, for instance) have been translated into two different languages; and *(ii)* those tasks where the same ontology (cmt–cmt) has been translated into two different languages. For the tasks of type (ii), good results are not directly related to the use of specific techniques for dealing with cross-lingual ontologies, but on the ability to exploit the identical structure of the ontologies.

For the sake of simplicity, we refer in the following to cross-lingual systems those implementing cross-lingual matching strategies and non-cross-lingual systems those without that feature.

This year, there were on 7 cross-lingual systems (out of 21): AML, CroLOM-Lite, IOMAP (renamed SimCat-Lite), LogMap, LPHOM, LYAM++, and XMap. Among these systems, only CroLOM-Lite and SimCat-Lite are specifically designed to this task. The reader can refer to the OAEI papers for a detailed description of the strategies adopted by each system.

The number of participants in fact increased with respect to the last campaign (5 in 2015, 3 in 2014, 7 in 2013, and 7 in 2012).

Following the OAEI evaluation rules, all systems should be evaluated in all tracks although it is expected that some system produce bad or no results. For this track, we observed different behaviours:

- CroMatcher and LYAM have experimented internal errors but were able to generated alignments for less than half of the tasks;
- Alin and Lily have generated no errors but empty alignments for all tasks;
- DKP-AOM and DKP-AOM-Lite were executed without errors but generated alignments for less than half of the tasks;

– NAISC has mostly generated erroneous correspondences (for very few tasks) and RiMOM has basically generated correspondences between ontology annotations;
– Dedicated systems (FCA-Map, LogMapBio, PhenoMF, PhenoMM and PhenoMP) required more than 30 minutes (in average) for completing a single task and were not evaluated;

In the following, we report the results for the systems dedicated to the task or that have been able to provide non-empty alignments for some tasks. We count on 12 systems (out of 21 participants).

8.2 Execution setting and runtime

The systems have been executed on a Ubuntu Linux machine configured with 8GB of RAM running under a Intel Core CPU 2.00GHz x4 processors. All measurements are based on a single run. As Table 18, we can observe large differences in the time required for a system to complete the 55 x 24 matching tasks. Note as well that the concurrent access to the SEALS repositories during the evaluation period may have an impact in the time required for completing the tasks.

8.3 Evaluation results

Table 18 presents the aggregated results for the matching tasks involving edas and ekaw ontologies. They have been computed using the Alignment API 4.6 and can slightly differ from those computed with the SEALS client. We haven't applied any threshold on the generated alignments. They are measured in terms of classical precision and recall (future evaluations should include weighted and semantic metrics).

For both types of tasks, most systems favor precision to the detriment of recall. The exception is LPHOM that has generated huge sets of correspondences (together with LYAM). As expected, (most) systems cross-lingual systems outperform the non-cross-lingual ones (the exceptions are LPHOM, LYAM and XMap, which have low performance for different reasons, i.e., many internal exceptions or poor ability to deal with the specifics of the task). On the other hand, this year, many non-cross-lingual systems dealing with matching at schema level have been executed with errors (CroMatcher, GA4OM) or were not able to deal at with the tasks (Alin, Lily, NAISC). Hence, their structural strategies could not be in fact evaluated (tasks of type ii). For both tasks, DKP-AOM and DKP-AOM-Lite have good performance in terms of precision but generating few correspondences for less than half of the matching tasks.

In particular, for the tasks of type (i), AML outperforms all other systems in terms of F-measure, followed by LogMap, CroLOM-Lite and SimCat-Lite. However, LogMap outperforms all systems in terms of precision, keeping a relatively good performance in terms of recall. For tasks of type (ii), AML decreases in performance with LogMap keeping its good results and outperforming all systems, followed by CroLOM-Lite and SimCat-Lite.

With respect to the pairs of languages for test cases of type (i), for the sake of brevity, we do not present them here. The reader can refer to the OAEI results web page for detailed results for each of the 55 pairs. While non-cross-lingual systems were not able to deal with many pairs of languages (in particular those involving the ar, cn, and ru languages), only 4 cross-lingual systems were able to deal with all pairs of languages

			Type (i) – 22 tests per pair				Type (ii) – 2 tests per pair			
System	Time	#pairs	Size	Prec.	F-m.	Rec.	Size	Prec.	F-m.	Rec.
AML	102	55	13.45	$.51_{(.51)}$	$.45_{(.45)}$	$.40_{(.40)}$	39.99	$.92_{(.92)}$	$.31_{(.31)}$	$.19_{(.19)}$
CroLOM-Lite	5501	55	8.56	$.55_{(.55)}$	$.36_{(.36)}$	$.28_{(.28)}$	38.76	$.89_{(.90)}$	$.40_{(.40)}$	$.26_{(.26)}$
LogMap	166	55	7.27	$.71_{(.71)}$	$.37_{(.37)}$	$.26_{(.26)}$	52.81	$.96_{(.96)}$	$.44_{(.44)}$	$.30_{(.30)}$
LPHOM	2497	34	84.22	$.01_{(.02)}$	$.02_{(.04)}$	$.08_{(.08)}$	127.91	$.13_{(.22)}$	$.13_{(.21)}$	$.13_{(.13)}$
LYAM	1367	24	177.30	$.01_{(.00)}$	$.006_{(.01)}$	$.00_{(.00)}$	283.95	$.03_{(.07)}$	$.02_{(.07)}$	$.03_{(.03)}$
SimCat-Lite	3938	54	7.07	$.59_{(.60)}$	$.34_{(.35)}$	$.25_{(.25)}$	30.11	$.90_{(.93)}$	$.33_{(.34)}$	$.21_{(.21)}$
XMap	134	31	3.93	$.30_{(.54)}$	$.01_{(.01)}$	$.00_{(.00)}$	.00	$.00_{(.00)}$	$.00_{(.00)}$	$.00_{(.00)}$
CroMatcher	65	25	2.91	$.29_{(.64)}$	$.004_{(.01)}$	$.00_{(.00)}$	.00	$.00_{(.00)}$	$.00_{(.00)}$	$.00_{(.00)}$
DKP-AOM	34	24	2.58	$.42_{(.98)}$	$.03_{(.08)}$	$.02_{(.02)}$	4.37	$.49_{(1.0)}$	$.01_{(.03)}$	$.01_{(.07)}$
DKP-AOM-Lite	35	24	2.58	$.42_{(.98)}$	$.03_{(.08)}$	$.02_{(.02)}$	4.37	$.49_{(1.0)}$	$.01_{(.03)}$	$.01_{(.01)}$
LogMapLite	21	55	1.16	$.35_{(.35)}$	$.04_{(.09)}$	$.02_{(.02)}$	94.50	$.01_{(.01)}$	$.01_{(.01)}$	$.01_{(.01)}$
NAISC	905	55	1.94	$.00_{(.00)}$	$.00_{(.01)}$	$.00_{(.00)}$	1.84	$.01_{(.01)}$	$.00_{(.01)}$	$.00_{(.01)}$

Table 18. MultiFarm aggregated results per matcher, for each type of matching task—different ontologies (i) and same ontologies (ii). Time is measured in minutes (for completing the 55×24 matching tasks). #pairs indicates the number of pairs of languages for which the tool is able to generated (non empty) alignments. Size indicates the average of the number of generated correspondences for the tests where an (non empty) alignment has been generated. Two kinds of results are reported: those do not distinguishing empty and erroneous (or not generated) alignments and those—indicated between parenthesis—considering only non empty generated alignments for a pair of languages.

(AML, CroLOM-Lite, LogMap and SimCat-Lite). LPHOM has particularly experimented problems with the pairs involving cn and cz.

Non-cross-lingual systems take advantage of the similarities in the lexicon of some languages, in the absence of specific strategies. This can be corroborated by the fact that most of them generate their best F-measure for the pairs es-pt (followed by de-en, fr-pt and it-pt). This (expected) fact has been observed along the campaigns. Another previously observed behaviour is related to the fact that, although it is likely harder to find correspondences between cz-pt than es-pt, for some non-cross-lingual systems this pair is present in their top F-measure.

Comparison with previous campaigns. The number of cross-lingual participants increased this year with respect to the last 2 campaigns (7 in 2016, 5 in 2015, 3 in 2014, 7 in 2013 and 2012 and 3 in 2011.5). This year, 4 systems have also participated last year (AML, LogMap, LYAM, and XMap) and we count on 3 new systems (CroLOM-Lite, LPHOM, SimCat-Lite).

Comparing the results from last year, in terms F-measure and with respect to the blind evaluation (cases of type i), AML slightly decreases its performance (.45 in 2016 and .47 in 2015). LogMap (and LogMap-Lite maintained its performance (.37), with XMap decreasing considerably in terms of recall but largely improving its execution time. Newcomers, specifically dedicated to the task, (CroLOM-Lite) and (SimCat-Lite) obtained F-measure near to (LogMap).

With respect to non-cross-lingual systems, last year CroMatcher finished the task without errors what explains its better performance, while DKP-AOM keeped the same results.

8.4 Conclusion

From 21 participants, half of them have been evaluated in this track. While some cross-lingual systems were not able to fully deal with the difficulties of the task, some others were not able to complete many tests due to internal errors, what is also the case for some non-cross-lingual systems.

In terms of performance, the F-measure for blind tests remains relatively stable across campaigns. AML and LogMap keep their positions with respect to the previous campaigns, followed this year by the new systems CroLOM-Lite and SimCat-Lite. Still, all systems privilege precision to the detriment of recall.

As expected, systems implementing specific methods for dealing with ontologies in different languages outperform non specific systems. Still, cross-lingual approaches are mainly based on translation strategies and the combination of other resources (such as cross-lingual links in Wikipedia or BabelNet) and strategies (machine learning, indirect alignment composition) remains underexploited. For most systems, the strategy consists of integrating one translation step before the matching itself.

Finally, this year, a *minimalistic* evaluation has been conducted (results have not been reported for the open data set). Furthermore, systems should also be evaluated using weighted and semantic measures. Multilingual tasks should also be considered and compared against cross-lingual settings.

9 Interactive matching

The interactive matching track was organised at OAEI 2016 for the fourth time. The goal of this evaluation is to simulate interactive matching [34,13], where a human expert is involved to validate correspondences found by the matching system. In the evaluation, we look at how interacting with the user improves the matching results. Further, we look at how the results of the matching systems are influenced when the experts make mistakes. Currently, this track does not evaluate the user experience nor the user interfaces of the systems [23].

9.1 Data sets

In this edition, we expanded the Interactive track and used data sets from four other OAEI tracks: Anatomy (Section 4), Conference (Section 5), LargeBio (Section 6), and Phenotype (Section 7). For details on the data sets, please refer to their respective sections.

9.2 Experimental setting

The Interactive track relies on the SEALS client's oracle class to simulate user interactions. An interactive matching system can present a correspondence to the oracle, which will tell the system whether that correspondence is correct or wrong. This year we have extended this functionality by allowing a user to present a collection of mappings simultaneously to the oracle.

To simulate the possibility of user errors, the oracle can be set to reply with a given error probability (randomly, from a uniform distribution). We evaluated systems with four different error rates: 0.0 (perfect oracle), 0.1, 0.2, and 0.3.

The evaluations of the Conference and Anatomy data sets were run on a server with 3.46 GHz (6 cores) and 8GB RAM allocated to the matching systems. Each system was

run ten times and the final result of a system for each error rate represents the average of these runs. This is the same configuration which was used in the non-interactive version of the Anatomy track and runtimes in the interactive version of this track are therefore directly comparable. For the Conference data set with the ra1 alignment, we considered macro-average of precision and recall of different ontology pairs, while the number of interactions represent the total number of interactions in all tasks. Finally, the ten runs are averaged.

The Phenotype and LargeBio evaluation was run on a Ubuntu Laptop with an Intel Core i7-4600U CPU @ 2.10GHz x 4 and allocating 15Gb of RAM. Each system was run only once due to the time required to run some of the systems. Since errors are randomly introduced we expect minor variations between runs.

9.3 Evaluation

The results are presented for each data set separately: Tables 19-22 and Figure 5 for the Anatomy data set, Tables 23-26 and Figure 6 for the Conference data set, Tables 27-30 and Figures 7-8 for the Disease and Phenotype data set[12], and Tables 35-42 and Figures -10 for the LargeBio data set[13].

For the tables we present the following information (column names in parentheses).

- The running time of the systems (Time) in seconds.
- The number of unsatisfiable classes resulting from the alignments computed as detailed in Section 6 - only for the LargeBio data set.
- The performance of the systems is measured using Precision (Prec.), Recall (Rec.) and F-measure (F-m.) with respect to the fixed reference alignment. For the Anatomy track we also present Recall+ (Rec.+) as in Section 4.
- To be able to compare the systems with and without interaction we also provide the performance results from the original tracks in Precision non-interactive (Prec. non), Recall non-interactive (Rec. non), F-measure non-interactive (F-m. non) and Recall+ non-interactive (Rec.+ non). For the ease of reading the tables this information is duplicated for each table.
- When the oracle makes mistakes, the oracle uses essentially a modified reference alignment. The performance of the system with respect to this modified reference alignment is given in Precision oracle (Prec. oracle), Recall oracle (Rec. oracle) and F-measure oracle (F-m. oracle). We note that for a perfect oracle these values are the same as the Precision (Prec.), Recall (Rec.) and F-measure (F-m.) values, respectively.
- Total requests (Tot Reqs.) represents the number of distinct user interactions with the tool, where each interaction can contain one or more correspondences that could be analysed simultaneously.

[12] Alin could not complete any of the Phenotype tasks, while XMap did not request any user interaction in the HP-MP data set and thus only participated *de facto* in the DOID-ORDO data set.

[13] We have used only the small FMA-NCI and SNOMED-NCI matching tasks of the *LargeBio* track (see Section 6) for interactive evaluation. Alin could only complete the small FMA-NCI task.

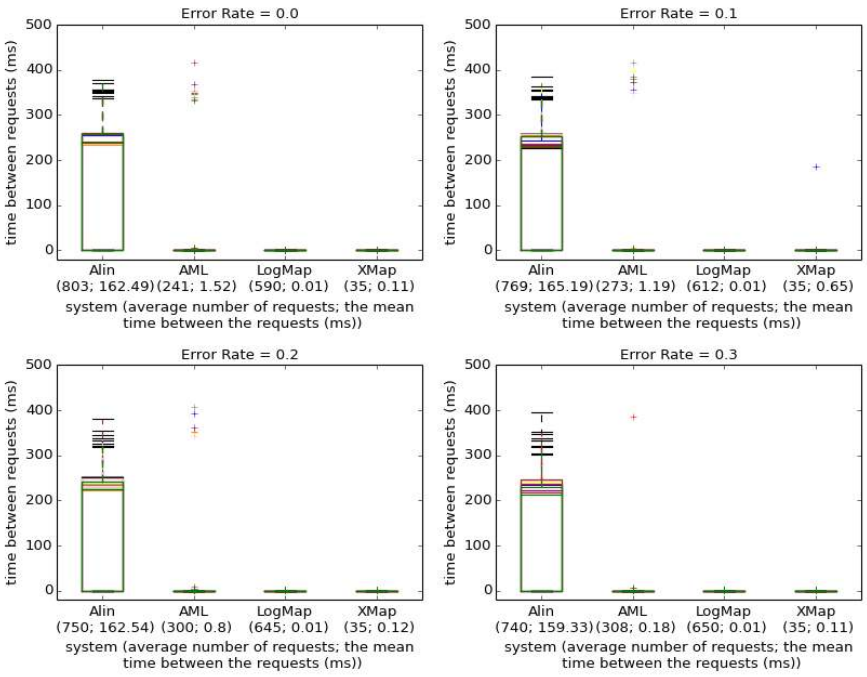


Fig. 5. Time intervals between requests to the user/oracle for the Anatomy data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the average number of requests and the mean time between the requests for the ten runs.

– In distinct mappings (Dist. Mapps) the mappings that are not conflicting are counted individually; and if more than three mappings are given, they are all counted independently, regardless of whether they are conflicting.
– We provide the true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) regarding the distinct mapping requests.
– Finally, we provide the performance of the oracle in positive precision (Pos. Prec.) and negative precision (Neg. Prec.). These are the fraction of positive, repectively, negative answers given by the oracle that are correct. We note that for a perfect oracle these values are always equal to 1.

The figures show the time intervals between the questions to the user/oracle for the different systems and error rates. Different runs are depicted with different colours.

9.4 Discussion

In this paper we provide our general observations and lessons learned. For more details we refer to the OAEI 2016 web site.

The different systems use different strategies for using the oracle. While LogMap, XMap and AML make use of user interactions exclusively in the post-matching steps to filter their candidate mappings, Alin can also add new candidate mappings to its ini-

Tool	Time	Prec.	Rec.	F-m.	Rec.+	Prec. non	Rec. non	F-m. non	Rec.+ Non.	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	505	0.99	0.75	0.85	0.35	0.98	0.34	0.50	0.00	0.99	0.75	0.85	803	1221	626	594	0	0	1.00	1.00
AML	48	0.97	0.95	0.96	0.86	0.95	0.94	0.94	0.83	0.97	0.95	0.96	241	240	51	189	0	0	1.00	1.00
LogMap	27	0.98	0.85	0.91	0.60	0.91	0.85	0.88	0.59	0.98	0.85	0.91	590	590	287	303	0	0	1.00	1.00
XMap	49	0.93	0.87	0.90	0.65	0.93	0.87	0.90	0.65	0.93	0.87	0.90	35	35	5	30	0	0	1.00	1.00

Table 19. Anatomy data set–perfect oracle

Tool	Time	Prec.	Rec.	F-m.	Rec.+	Prec. non	Rec. non	F-m. non	Rec.+ Non.	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	489	0.95	0.70	0.81	0.31	0.98	0.34	0.50	0.00	0.99	0.74	0.85	769	1123	554	451	51	67	0.92	0.87
AML	50	0.95	0.95	0.95	0.86	0.95	0.94	0.94	0.83	0.97	0.95	0.96	273	272	47	194	23	6	0.67	0.97
LogMap	24	0.96	0.83	0.89	0.57	0.91	0.85	0.88	0.59	0.96	0.83	0.89	612	612	258	290	35	28	0.88	0.91
XMap	46	0.93	0.87	0.90	0.65	0.93	0.87	0.90	0.65	0.93	0.87	0.90	35	35	4.2	27.5	2.7	0.8	0.61	0.97

Table 20. Anatomy data set–error rate 0.1

Tool	Time	Prec.	Rec.	F-m.	Rec.+	Prec. non	Rec. non	F-m. non	Rec.+ Non.	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	481	0.91	0.66	0.77	0.27	0.98	0.34	0.50	0.00	0.99	0.74	0.85	750	1077	493	368	94	121	0.84	0.75
AML	48	0.94	0.94	0.94	0.85	0.95	0.94	0.94	0.83	0.97	0.95	0.96	300	299	46	193	46	13	0.50	0.94
LogMap	24	0.94	0.82	0.88	0.54	0.91	0.85	0.88	0.59	0.94	0.81	0.87	645	645	225	287	70	61	0.76	0.82
XMap	47	0.93	0.87	0.90	0.65	0.93	0.87	0.90	0.65	0.93	0.87	0.90	35	35	4.3	25.1	5.4	0.7	0.45	0.97

Table 21. Anatomy data set–error rate 0.2

Tool	Time	Prec.	Rec.	F-m.	Rec.+	Prec. non	Rec. non	F-m. non	Rec.+ Non.	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	472	0.87	0.62	0.72	0.23	0.98	0.34	0.50	0.00	0.99	0.73	0.84	740	1058	430	311	134	182	0.76	0.63
AML	47	0.93	0.94	0.93	0.84	0.95	0.94	0.94	0.83	0.97	0.95	0.96	308	307	40	177	71	18	0.36	0.91
LogMap	24	0.93	0.82	0.87	0.54	0.91	0.85	0.88	0.59	0.93	0.80	0.86	650	650	202	256	106	84	0.66	0.75
XMap	47	0.93	0.87	0.90	0.65	0.93	0.87	0.90	0.65	0.93	0.86	0.90	35	35	3.1	21.8	8.9	1.9	0.27	0.92

Table 22. Anatomy data set–error rate 0.3

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	101	0.96	0.74	0.83	0.89	0.26	0.40	0.96	0.74	0.83	326	574	144	429	0	0	1.00	1.00
AML	29	0.91	0.71	0.80	0.84	0.66	0.74	0.91	0.71	0.80	271	270	47	223	0	0	1.00	1.00
LogMap	26	0.89	0.61	0.72	0.82	0.59	0.69	0.89	0.61	0.72	142	142	49	93	0	0	1.00	1.00
XMap	21	0.84	0.57	0.68	0.84	0.57	0.68	0.84	0.57	0.68	4	4	0	4	0	0	0.00	1.00

Table 23. Conference data set–perfect oracle

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	101	0.79	0.67	0.73	0.89	0.26	0.40	0.96	0.74	0.84	315	557	124	375	42	15	0.75	0.96
AML	30	0.85	0.70	0.77	0.84	0.66	0.74	0.92	0.73	0.82	285	279	51	204	18	5	0.74	0.98
LogMap	26	0.85	0.60	0.70	0.82	0.59	0.69	0.86	0.59	0.70	140	140	45	81	10	3	0.82	0.97
XMap	22	0.84	0.57	0.68	0.84	0.57	0.68	0.84	0.57	0.68	4	4	0	3.6	0.4	0	0.00	1.00

Table 24. Conference data set–error rate 0.1

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	100	0.67	0.62	0.64	0.89	0.26	0.40	0.96	0.75	0.84	303	538	108	321	81	27	0.57	0.92
AML	33	0.77	0.68	0.72	0.84	0.66	0.74	0.93	0.75	0.83	290	277	53	170	42	11	0.56	0.94
LogMap	26	0.82	0.59	0.69	0.82	0.59	0.69	0.83	0.58	0.68	143	143	38	75	18	10	0.68	0.88
XMap	21	0.84	0.57	0.68	0.84	0.57	0.68	0.84	0.57	0.68	4	4	0	3.2	0.8	0	0.00	1.00

Table 25. Conference data set–error rate 0.2

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	99	0.57	0.57	0.57	0.89	0.26	0.40	0.97	0.77	0.86	303	535	93	279	120	42	0.44	0.87
AML	30	0.72	0.65	0.68	0.84	0.66	0.74	0.93	0.75	0.83	284	269	47	143	58	20	0.45	0.88
LogMap	26	0.80	0.59	0.68	0.82	0.59	0.69	0.80	0.56	0.66	144	144	33	67	28	15	0.54	0.82
XMap	22	0.84	0.57	0.68	0.84	0.57	0.68	0.84	0.57	0.68	4	4	0	2.9	1.1	0	0.00	1.00

Table 26. Conference data set–error rate 0.3

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	308	0.945	0.899	0.921	0.9	0.851	0.875	0.945	0.899	0.921	388	388	192	196	0	0	1	1
LogMap	329	0.96	0.96	0.96	0.755	0.957	0.844	0.96	0.96	0.96	1,928	1,928	551	1,377	0	0	1	1

Table 27. Phenotype: HP-MP data set–perfect oracle

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	306	0.932	0.886	0.908	0.9	0.851	0.875	0.945	0.899	0.921	388	388	171	176	20	21	0.895	0.893
LogMap	346	0.888	0.932	0.909	0.755	0.957	0.844	0.912	0.912	0.912	1,891	1,891	498	1,208	132	53	0.79	0.958

Table 28. Phenotype: HP-MP data set–error rate 0.1

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	309	0.923	0.868	0.895	0.9	0.851	0.875	0.944	0.894	0.918	358	358	144	140	32	42	0.818	0.769
LogMap	367	0.836	0.915	0.874	0.755	0.957	0.844	0.871	0.871	0.871	1,855	1,855	440	1,042	262	111	0.627	0.904

Table 29. Phenotype: HP-MP data set–error rate 0.2

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	299	0.905	0.856	0.88	0.9	0.851	0.875	0.944	0.899	0.921	390	390	124	138	58	70	0.681	0.663
LogMap	263	0.796	0.907	0.848	0.755	0.957	0.844	0.83	0.83	0.83	1,827	1,827	387	892	384	164	0.502	0.845

Table 30. Phenotype: HP-MP data set–error rate 0.3

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	525	0.929	0.993	0.96	0.824	0.99	0.899	0.929	0.993	0.96	413	413	115	298	0	0	1	1
LogMap	440	0.994	0.972	0.983	0.904	0.932	0.918	0.994	0.972	0.983	1,602	1,602	780	822	0	0	1	1
XMap	2,352	0.933	0.714	0.809	0.977	0.622	0.76	0.933	0.714	0.809	11	11	3	8	0	0	1	1

Table 31. Phenotype: DOID-ORDO data set–perfect oracle

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	507	0.912	0.988	0.948	0.824	0.99	0.899	0.93	0.992	0.96	413	413	108	266	32	7	0.771	0.974
LogMap	492	0.949	0.927	0.938	0.904	0.932	0.918	0.961	0.939	0.95	1,677	1,677	698	815	82	82	0.895	0.909
XMap	2,603	0.931	0.713	0.808	0.977	0.622	0.76	0.932	0.713	0.808	11	11	3	7	1	0	0.75	1

Table 32. Phenotype: DOID-ORDO data set–error rate 0.1

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	513	0.899	0.979	0.937	0.824	0.99	0.899	0.931	0.992	0.961	413	413	94	242	56	21	0.627	0.92
LogMap	428	0.906	0.91	0.908	0.904	0.932	0.918	0.913	0.892	0.902	1,699	1,699	621	716	203	159	0.754	0.818
XMap	2,302	0.931	0.712	0.807	0.977	0.622	0.76	0.932	0.713	0.808	11	11	1	7	1	2	0.5	0.778

Table 33. Phenotype: DOID-ORDO data set–error rate 0.2

Tool	Time	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	540	0.881	0.975	0.926	0.824	0.99	0.899	0.933	0.992	0.962	413	413	88	204	93	28	0.486	0.879
LogMap	427	0.883	0.904	0.893	0.904	0.932	0.918	0.864	0.845	0.854	1,760	1,760	555	681	299	225	0.65	0.752
XMap	2,260	0.931	0.712	0.807	0.977	0.622	0.76	0.932	0.713	0.808	11	11	1	7	1	2	0.5	0.778

Table 34. Phenotype: DOID-ORDO data set–error rate 0.3

Tool	Time	Unsat.	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	5,859	2	0.996	0.63	0.772	0.995	0.455	0.624	0.996	0.63	0.772	653	1,019	470	549	0	0	1	1
AML	60	2	0.99	0.913	0.95	0.963	0.902	0.932	0.99	0.913	0.95	449	447	217	230	0	0	1	1
LogMap	38	2	0.992	0.901	0.944	0.944	0.897	0.92	0.992	0.901	0.944	1,131	1,131	594	537	0	0	1	1
XMap	50	2	0.991	0.9	0.943	0.977	0.901	0.937	0.991	0.9	0.943	188	188	114	74	0	0	1	1

Table 35. LargeBio: FMA-NCI small data set–perfect oracle

Tool	Time	Unsat.	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	5,616	85	0.971	0.614	0.752	0.995	0.455	0.624	0.996	0.63	0.772	629	932	426	420	43	43	0.908	0.907
AML	61	222	0.98	0.908	0.943	0.963	0.902	0.932	0.99	0.914	0.95	497	484	224	219	26	15	0.896	0.936
LogMap	39	2	0.98	0.881	0.928	0.944	0.897	0.92	0.983	0.892	0.935	1,209	1,209	536	582	33	58	0.942	0.909
XMap	51	2	0.988	0.895	0.939	0.977	0.901	0.937	0.99	0.9	0.943	187	187	100	68	4	15	0.962	0.819

Table 36. LargeBio: FMA-NCI small data set–error rate 0.1

Tool	Time	Unsat.	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	5,398	152	0.958	0.593	0.733	0.995	0.455	0.624	0.996	0.624	0.767	605	881	370	353	63	95	0.855	0.788
AML	63	2	0.974	0.894	0.932	0.963	0.902	0.932	0.987	0.91	0.947	450	450	166	185	43	56	0.794	0.768
LogMap	38	2	0.967	0.874	0.918	0.944	0.897	0.92	0.964	0.875	0.917	1,247	1,247	488	558	95	106	0.837	0.84
XMap	58	2	0.988	0.892	0.938	0.977	0.901	0.937	0.99	0.899	0.942	187	187	92	67	6	22	0.939	0.753

Table 37. LargeBio: FMA-NCI small data set–error rate 0.2

Tool	Time	Unsat.	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
Alin	5,347	91	0.937	0.58	0.716	0.995	0.455	0.624	0.996	0.623	0.767	589	855	335	293	99	128	0.772	0.696
AML	63	2	0.966	0.894	0.929	0.963	0.902	0.932	0.981	0.911	0.945	450	450	160	174	53	63	0.751	0.734
LogMap	39	2	0.963	0.872	0.915	0.944	0.897	0.92	0.935	0.849	0.89	1,327	1,327	429	572	161	165	0.727	0.776
XMap	53	2	0.985	0.887	0.933	0.977	0.901	0.937	0.99	0.899	0.942	188	188	80	59	14	35	0.851	0.628

Table 38. LargeBio: FMA-NCI small data set–error rate 0.3

Tool	Time	Unsat.	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	730	0	0.972	0.726	0.831	0.904	0.713	0.797	0.972	0.726	0.831	2,730	2,730	1,657	1,073	0	0	1	1
LogMap	628	0	0.985	0.669	0.797	0.922	0.663	0.771	0.985	0.669	0.797	5,596	5,596	3,742	1,854	0	0	1	1
XMap	984	35,869	0.924	0.59	0.72	0.911	0.564	0.697	0.924	0.59	0.72	11,932	11,689	10,090	1,599	0	0	1	1

Table 39. LargeBio: SNOMED-NCI small data set–perfect oracle

Tool	Time	Unsat.	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	759	0	0.967	0.717	0.823	0.904	0.713	0.797	0.972	0.724	0.83	2,730	2,730	1,495	979	92	164	0.942	0.857
LogMap	619	16	0.974	0.651	0.78	0.922	0.663	0.771	0.971	0.656	0.783	6,201	6,201	3,357	2,263	196	385	0.945	0.855
XMap	957	35,455	0.923	0.591	0.721	0.911	0.564	0.697	0.84	0.568	0.678	11,931	11,694	9,095	1,512	89	998	0.99	0.602

Table 40. LargeBio: SNOMED-NCI small data set–error rate 0.1

Tool	Time	Unsat.	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	762	0	0.961	0.707	0.815	0.904	0.713	0.797	0.972	0.721	0.828	2,730	2,730	1,331	891	181	327	0.88	0.732
LogMap	625	16	0.965	0.64	0.77	0.922	0.663	0.771	0.948	0.639	0.763	6,737	6,737	2,977	2,505	490	765	0.859	0.766
XMap	943	35,968	0.921	0.591	0.72	0.911	0.564	0.697	0.754	0.541	0.63	11,911	11,682	8,052	1,403	204	2023	0.975	0.41

Table 41. LargeBio: SNOMED-NCI small data set–error rate 0.2

Tool	Time	Unsat.	Prec.	Rec.	F-m.	Prec. non	Rec. non	F-m. non	Prec. oracle	Rec. oracle	F-m. oracle	Tot. Reqs.	Dist. Mapps.	TP	TN	FP	FN	Pos. Prec.	Neg. Prec.
AML	758	0	0.955	0.697	0.806	0.904	0.713	0.797	0.972	0.719	0.827	2,730	2,730	1,184	798	264	484	0.818	0.622
LogMap	635	16	0.959	0.635	0.764	0.922	0.663	0.771	0.92	0.62	0.741	7,159	7,159	2,607	2,563	854	1,135	0.753	0.693
XMap	984	36,619	0.919	0.592	0.72	0.911	0.564	0.697	0.676	0.514	0.584	11,903	11,693	7,090	1,266	347	2,990	0.953	0.297

Table 42. LargeBio: SNOMED-NCI small data set–error rate 0.3

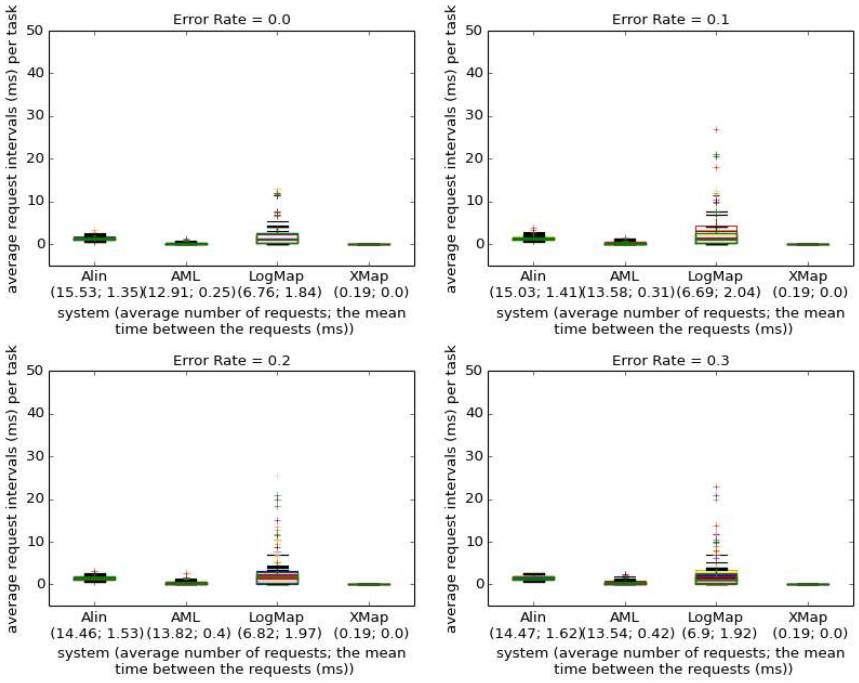


Fig. 6. Average time between requests per task in the Conference data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the average number of requests and the mean time between the requests (calculated by taking the average of the average request intervals per task) for the ten runs and all tasks.

tial set. LogMap and AML both request feedback on only selected mapping candidates (based on their similarity patterns or their involvement in unsatisfiabilities) and only present one mapping at a time to the user. XMap also presents one mapping at a time and asks mainly for true negatives. Only Alin employs the new feature in this year's evaluation: analysing several conflicting mappings simultaneously, whereby a system can present up to three mappings together to the oracle, provided that each mapping presented has a mapped entity, i.e., class or property, in common with at least one other mapping presented.

The performance of the systems improves when interacting with a perfect oracle compared to no interaction. Although systems' performance deteriorates when *moving towards larger error rates* there are still benefits from the user interaction—some of the systems' measures stay above their non-interactive values even for the larger error rates. For the Anatomy track Alin detects only trivial correspondences in the non-interactive version while user interactions led to detecting some non-trivial correspondences.

The *impact of the oracle's errors* is linear for Alin, AML and XMap and supra-linear for LogMap for all data sets. The "Positive Precision" value affects the true positives and false positives, and the "Negative Precision" value affects the true negatives and

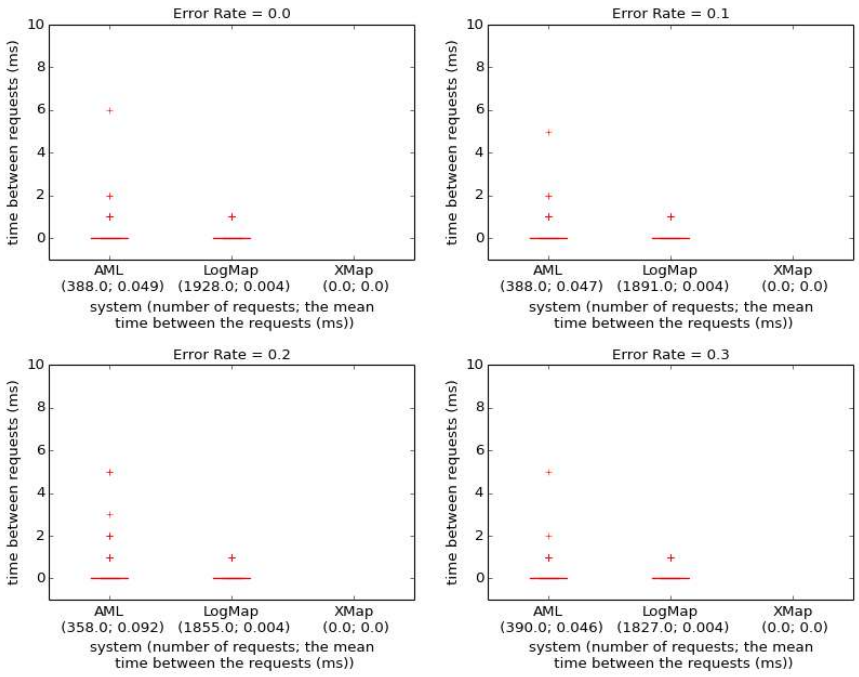


Fig. 7. Time between requests per task in the HP-MP data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the number of requests and the mean time between the requests.

false negatives. The more a system relies on the oracle, the more sensitive it will be to its errors.

In general, XMap performs very few requests to the oracle compared to the other systems.

Two models for system *response times* are frequently used in the literature [10]: Shneiderman and Seow take different approaches to categorise the response times. Shneiderman takes a task-centred view and sorts the response times in four categories according to task complexity: typing, mouse movement (50-150 ms), simple frequent tasks (1 s), common tasks (2-4 s) and complex tasks (8-12 s). He suggests that the user is more tolerable to delays with the growing complexity of the task at hand. Unfortunately, no clear definition is given for how to define the task complexity. Seow's model looks at the problem from a user-centred perspective by considering the user expectations towards the execution of a task: instantaneous (100-200 ms), immediate (0.5-1 s), continuous (2-5 s), captive (7-10 s); Ontology alignment is a cognitively demanding task and can fall into the third or fourth categories in both models. In this regard the response times (request intervals as we call them above) observed in all data sets fall into the tolerable and acceptable response times, and even into the first categories, in both models. The request intervals for both AML and LogMap stay under 3 ms for all

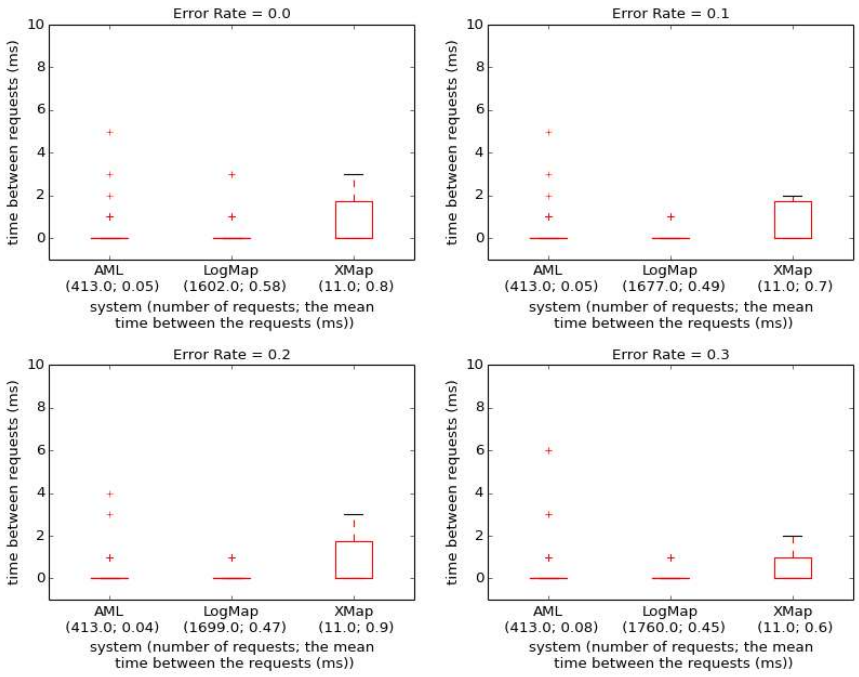


Fig. 8. Time between requests per task in the DOID-ORDO data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the number of requests and the mean time between the requests.

data sets. Alin's request intervals are higher, but still in the tenth of second range. It could be the case however that the user could not take advantage of very low response times because the task complexity may result in higher user response time (analogically it measures the time the user needs to respond to the system after the system is ready).

Regarding the number of unsatisfiable classes resulting from the alignments we observe some expected variations as the error increases. We note that, with interaction, the alignments produced by the systems are typically larger than without interaction, which makes the repair process harder. The introduction of oracle errors complicates the process further, and may make an alignment irreparable if the system follows the oracle's feedback blindly.

10 Instance matching

The instance matching track aims at evaluating the performance of matching tools when the goal is to detect the degree of similarity between pairs of items/instances expressed in the form of RDF data. The track is organized in three independent tasks called *SABINE*, *SYNTHETIC* and *DOREMUS*. Each test is based on two data sets called source and target and the goal is to discover the matching pairs, i.e., mappings, among the instances in the source data set and the instances in the target data set.

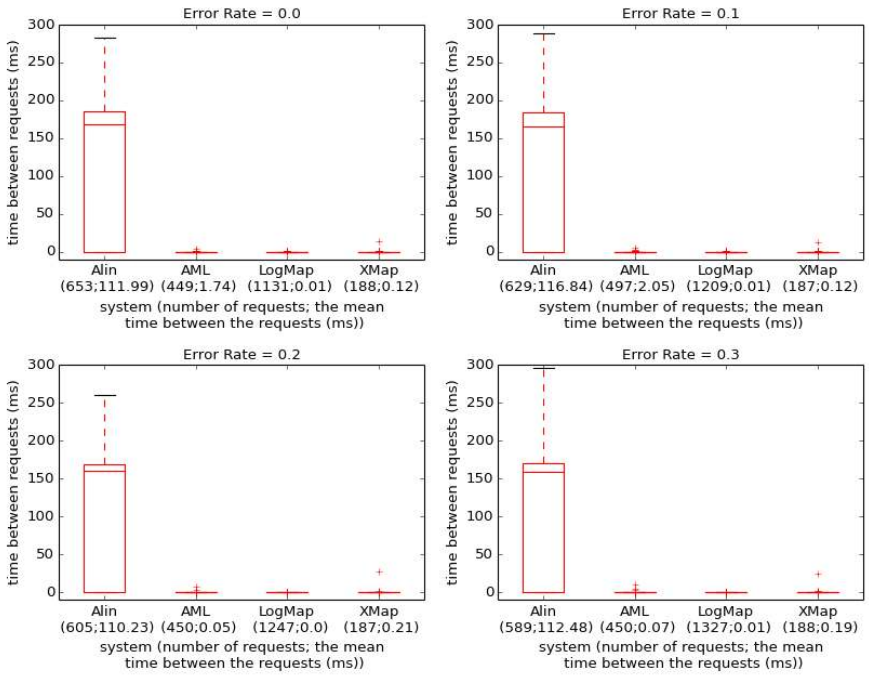


Fig. 9. Time between requests per task in the FMA-NCI data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the number of requests and the mean time between the requests.

For the sake of clarity, we split the presentation of task results in three different sections as follows.

10.1 Results of the SABINE task

SABINE is a modular benchmark in the domain of European politics for Social Business Intelligence (SBI) and it includes an ontology with 500 topics, both in English and Italian languages. The task is articulated in two sub-tasks called *inter-lingual mapping* and *data linking*.

In inter-lingual mapping, source and target datasets are OWL ontologies containing topics as instances of the class “Topic”. The source ontology contains topics in the English language; the target ontology contains other topics in the Italian language. The goal is to discover mappings between English and Italian topics by also defining the kind of relation which is most suitable for describing the discovered mapping between two matching topics.

In data linking, just the source dataset is defined and it is given to the participants as an OWL ontology containing topics as instances of the class “Topic”. The goal is to discover the best corresponding DBpedia entity for each topic in the source ontology.

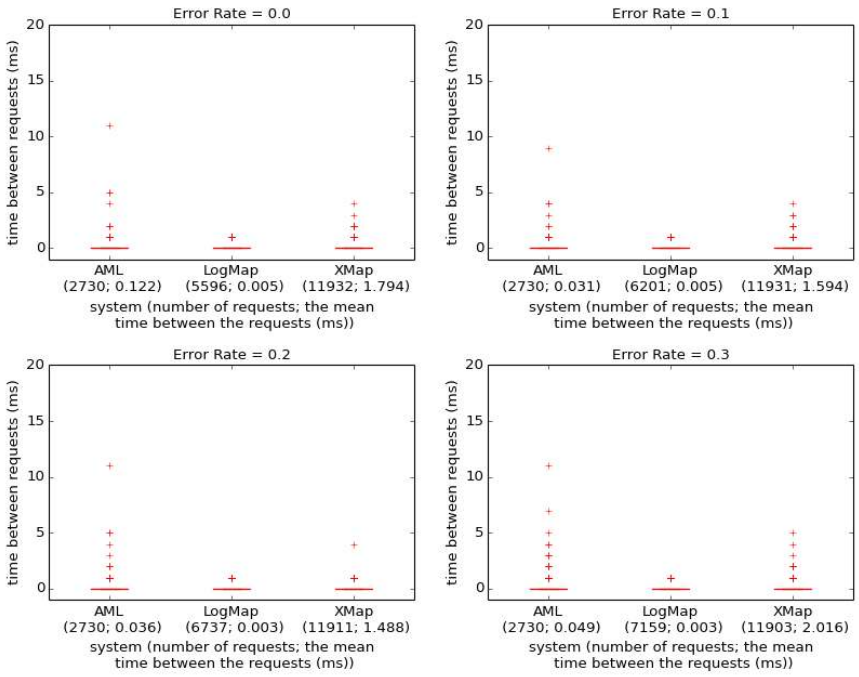


Fig. 10. Time between requests per task in the SNOMED-NCI data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the number of requests and the mean time between the requests.

The SABINE sub-tasks are defined as open tests, meaning that the set of expected mappings, i.e., reference alignment, is given in advance to the participants and it constitutes the gold standard for result evaluation. The task size is around 23K ontology instances to consider. The gold standard has been defined through crowdsourcing validation though the Argo system[14]. For creating the gold standard, workers are called to recognize and confirm the mapping between instance topics of source and target ontologies. In particular, a task is represented as a choice question in which a topic of the source ontology is specified and a number of instance topics of the target ontology are provided as possible mappings. A worker receiving a task to execute has to consider a source topic and to choose the most appropriate mapping with a target topic among those provided as possible options. Multi-worker task assignment and consensus evaluation techniques are defined in Argo for quality assessment of the task result. A task is assigned to a group G of 6 different workers. A group member autonomously executes a task and independently produces the answer according to her/his personal feeling and judgement. Given a task, its result is defined as an answer agreement, i.e., consensus, among the members of the group that executed the task. Two workers agree on a task result when they selected the same target topic as mapping with the given source topic.

[14] `http://island.ricerca.di.unimi.it/projects/argo/` (in Italian).

The mapping between the source and the target topics is confirmed and inserted in the gold standard when the task answer having the highest degree of consensus within the group G is supported by a qualified majority larger than 50%. Conversely, when a qualified majority of workers is not found in G, the task is uncommitted and it is scheduled for re-execution by a different group of workers with higher reliability. Further details on the Argo techniques for task management are provided in [7]. The gold standard of the SABINE task contains 249 crowd-validated mappings for the inter-lingual sub-task and 338 crowd-validated mappings for the data linking sub-task.

Participants to the SABINE sub-tasks are *LogMapIm*, *AML*, *LogMapLite*, and *RiMOM*. Results are shown in Table 43. For each test, the tool performances are expressed in terms of precision, recall, and F-measure.

	Inter-lingual mapping			Data linkinging		
	Precision	F-measure	Recall	Precision	F-measure	Recall
LogMapIm	0.012	0.014	0.016	NaN	NaN	0.0
AML	0.919	0.917	0.916	0.926	0.889	0.855
LogMapLite	0.358	0.214	0.153	NaN	NaN	0.0
RiMOM	0.955	0.943	0.932	0.424	0.580	0.917

Table 43. Instance matching results.

We focus our considerations on AML and RiMOM that provided high-value results for precision, recall, and F-measure on both inter-lingual mapping and data linking sub-tasks. In particular, RiMOM outperforms AML on the inter-lingual mapping sub-task, in that both precision and recall values of RiMOM are higher than the corresponding values of AML. However, both tools are over 90% for precision and recall values, meaning that mapping corresponding instances of different languages is a successfully-addressed task by RiMOM and AML. In the data linking sub-task, AML outperforms RiMOM on precision and the difference between the tools on this result value is significant (i.e., AML $>$90% and RiMOM $<$50% on the precision value). On the opposite, for recall, we note that the RiMOM value is higher than the AML value, and the result of both tools is very positive (i.e., $>$85%). We argue that these results on the data linking sub-task are due to the problem of selecting the most appropriate mapping when a number of possible alternatives are available. Both AML and RiMOM are successful in providing a set of candidate DBpedia entities as target mapping with a given OWL instance (i.e., high recall value). On the opposite, the capability to choose/select the most appropriate mapping among the set of available options is still challenging and only AML succeeds in providing high-quality results on this task (i.e., high precision value).

10.2 Results of the SYNTHETIC task

UOBM and SPIMBENCH tasks are two of the evaluation tasks of instance matching tools where the goal is to determine when two OWL instances describe the same real world object. For the first task, the data sets have been produced by altering a set of source data and generated by SPIMBENCH [37] with the aim to generate descriptions of the same entity where value-based, structure-based and semantics-aware transformations are employed in order to create the target data. While, for the latter task the data

sets have been generated with the University Ontology Benchmark (UOBM) [30] and transformed with the LANCE benchmark generator [36].

For both tasks, the transformations applied were a combination of value-based, structure-based, and semantics-aware test cases. The value-based transformations consider mainly typographical errors and different data formats, the structure-based transformations consider transformations applied on the structure of object and datatype properties and the semantics-aware transformations are transformations at the instance level considering the TBox information. The latter are used to examine if the matching systems take into account RDFS and OWL semantics in order to discover correspondences between instances that can be found only by considering information found in the TBox.

We stress that an instance in the source data set can have none or one matching counterpart in the target data set. A data set is composed of a TBox and a corresponding ABox. Source and target data sets share almost the same TBox (differences in the properties, due to the structure-based transformations). For SPIMBENCH, the sandbox scale is 10K triples ≈380 instances while the mainbox scale is 50K triples ≈1800 instances. We asked the participants to match the Creative Works instances (NewsItem, BlogPost and Programme) in the source data set against the instances of the corresponding class in the target data set. For UOBM, the sandbox scale is 14K triples ≈2.5K instances while the mainbox scale is 60K triples ≈10K instances. We asked the participants to match all the instances that are not common to the two data sets. For both tasks, we expected to receive a set of links denoting the pairs of matching instances that they found to refer to the same entity.

The participants to these tasks are LogMap, AML and RiMOM. For evaluation, we built a ground truth containing the set of expected links where an instance i_1 in the source data set is associated with an instance in the target data set that has been generated as an altered description of i_1.

The way that the transformations were done, was to apply value-based, structure-based and semantics-aware transformations, on different triples pertaining to one class instance.

The systems were judged on the basis of precision, recall and F-measure results that are shown in Tables 44 and 45.

	Sandbox task			Mainbox task		
	Precision	F-measure	Recall	Precision	F-measure	Recall
LogMap	0.958	0.851	0.766	0.981	0.814	0.695
AML	0.907	0.82	0.749	0.9	0.816	0.747
RiMOM	0.984	0.992	1	0.991	0.995	1

Table 44. Results of the SPIMBENCH task.

LogMap responds well regarding the SPIMBENCH task, while the performance drops when matching the data sets of the UOBM task. LogMap is automatic and does not require the definition of a configuration file in contrast to AML and RiMOM.

	Sandbox task			Mainbox task		
	Precision	F-measure	Recall	Precision	F-measure	Recall
LogMap	0.701	0.32	0.207	0.625	0.044	0.023
AML	0.785	0.665	0.577	0.509	0.512	0.515
RiMOM	0.771	0.821	0.877	0.443	0.477	0.516

Table 45. Results of the UOBM task.

AML responds well regarding the SPIMBENCH task, while the performance drops when matching the data sets of the UOBM task. AML had to turn off the reasoner in order to handle missing information about the domain and range of TBox properties.

LogMap and AML produce links that are quite often correct (resulting in a good precision) but fail in capturing a large number of the expected links (resulting in a lower recall).

RiMOM performs better than any other system for most of the tasks; it performs excellent in the case of SPIMBENCH but, although it exhibits the best results for the Sandbox track of UOBM, its performance drops for the Mainbox track. For RiMOM, the probability of capturing a correct link is high, but the probability of a retrieved link to be correct is lower, resulting in a high recall but not a high precision.

The main comments for the SPIMBENCH and UOBM tasks are:

- LogMap and AML have consistent behaviour regarding Sandbox and Mainbox.
- RiMOM has a consistent behaviour for the SPIMBENCH task and an inconsistent behaviour for the UOBM task.
- All systems performed well for the SPIMBENCH task.
- The UOBM data sets seem to be more "difficult" for both IM systems, and this difficulty stems from the data set itself, rather than from the transformations imposed by LANCE.
- The UOBM data sets seem to be more difficult for both IM systems, and this difficulty stems from the data set itself, rather than from the transformations imposed by LANCE. In particular, an important source of difficulty for the systems is that the URIs of the instances in the data set look very similar to each other, so even the change of a URI can lead to false positives or false negatives.

10.3 Results of the DOREMUS task

The DOREMUS task, having its premier at OAEI, contains real world data sets coming from two major French cultural institutions—The BnF (French National Library) and the PP (Philharmonie de Paris). The data are about classical music works and follow the DOREMUS model (one single vocabulary for both data sets) issued from the DOREMUS project[15]. Each data entry, or instance, is a bibliographical record about a musical piece, containing properties such as the composer, the title(s) of the work, the year of creation, the key, the genre, the instruments, to name a few. These data have been converted to RDF from their original UNI- and INTER-MARC format and anchored to the DOREMUS ontology and a set of domain controlled vocabularies by the help of the *marc2rdf* converter[16], developed for this purpose within the DOREMUS Project (for

[15] http://www.doremus.org

[16] https://github.com/DOREMUS-ANR/marc2rdf

more details on the conversion method and on the ontology we refer to [1] and [29]). Note that these data are highly heterogeneous. We have selected works described both at the BnF and at the PP with different degrees of heterogeneity in their descriptions. The data sets have been selected in three sub-tasks.

Nine heterogeneities. This task consists in aligning two small data sets, BnF-1 and PP-1, containing about 40 instances each, by discovering 1:1 equivalence relations between their instances. There are 9 types of heterogeneities that these data manifest, that have been identified by the music library experts, such as multilingualism, differences in catalogues, differences in spelling, different degrees of description (number of properties).

Four heterogeneities. This task consists in aligning two larger data sets, BnF-2 and PP-2, containing about 200 instances each, by discovering 1:1 equivalence relations between the instances that they contain. There are 4 types of heterogeneities that these data manifest, that we have selected from the nine in Task 1 and that appear to be the most problematic: 1) Orthographical differences, 2) Multilingual titles, 3) Missing properties, 4) Missing titles.

The False Positives Trap. This task consists in correctly disambiguating the instances contained in two data sets, BnF-3 and PP-3, by discovering 1:1 equivalence relations between the instances that they contain. We have selected several groups of pairs of works with highly similar descriptions where there exists only one correct match in each group. The goal is to challenge the linking tools capacity to avoid the generation of false positives and match correctly instances in the presence of highly similar but still distinct candidates.

	9 heterogeneities			4 heterogeneities			False positive trap		
	Prec.	F-m.	Rec.	Prec.	F-m.	Rec.	Prec.	F-m.	Rec.
AML (th=0.2)	0.966	0.918	0.875	0.934	0.848	0.776	0.921	0.886	0.854
AML (th=0.6)	0.962	0.862	0.781	0.943	0.83	0.741	0.853	0.773	0.707
RiMOM	0.813	0.813	0.813	0.746	0.746	0.746	0.707	0.707	0.707

Table 46. Results of the DOREMUS task

Results Only two systems returned results on the track: AML and RiMOM. Note that AML has been configured with two different thresholds. The results of their performances, evaluated by using precision, recall and F-measure, on each of the three tasks can be seen in Table 46. The best performance in terms of F-measure is provided by the AML tool with a threshold of 0.2 on all tasks.

11 Process Model Matching

In 2013 and in 2015 the community interested in business process modelling conducted an evaluation campaign similar to OAEI [3]. Instead of matching ontologies, the task was to match process models described in different formalisms like BPMN and Petri Nets. Within this track we offer a subset of the tasks from the Process Model Matching Contest as OAEI track by converting the process models to an ontological representation. By offering this track, we hope to gain insights in how far ontology matching

systems are capable of solving the more specific problem of matching process models. This track is also motivated by the discussions at the end of the 2015 Ontology Matching workshop, where many participants showed their interest in such a track.

11.1 Experimental Settings

We were using the first data set from the 2015 Process Matching Contest. This data set deals with processing applications to a university. It consists of nine different process models where each describes the concrete process of a specific German university. The models are encoded as BPMN process models. We converted the BPMN representation of the process models to a set of assertions (ABox) using the vocabulary defined in the BPMN 2.0 ontology (TBox). For that reason the resulting matching task is an instance matching task where each ABox is described by the same TBox. For each pair of processes manually generated reference alignments are available. Typical activities within that domain are *Sending acceptance*, *Invite student for interview*, or *Wait for response*. These examples illustrate one of the main differences from the ontology matching task. The labels are usually verb-object phrases that are sometimes extended with more words. Another important difference is related to the existence of an execution order, i.e., the model is a complex sequence of activities, which can be understood as the counterpart to a type hierarchy.

Only few systems have been marked as capable of generating alignments for the Process Model Matching track. We have tried to execute all these systems, however, some of them generated only trivial TBox mappings instead of mappings between activities. After contacting the developer of the systems, we received the feedback that the systems have been marked mistakenly and are designed for terminological matching only. We have excluded them from the evaluation. Moreover, we tried to run all systems that were marked as instance matching tools, which have been submitted as executable SEALS bundles. One of these tools (LogMap), generated meaningful results and was also added to the set of systems that we evaluated. Finally we evaluated three systems (AML, LogMap, and DKP), one of these systems was configured in two different settings related to the treatment of events-to-activity mappings. This was the tool DKP. Thus we distinguish between DKP and DKP*.

In our evaluation, we computed standard precision and recall, as well as the harmonic mean known as F-measure. The data set we used consists of several test cases. We aggregated the results and present the micro average results. The gold standard we used for our first set of evaluation experiments is based on the gold standard that has also been used at the Process Model Matching Contest in 2015 [3]. We modified only some minor mistakes (resulting in changes less than 0.5 percentage points). In order to compare the results to the results obtained by the process model matching community, we present also the recomputed values of the submissions to the 2015 contest.

Moreover, we extended our evaluation ("Standard" in Table 47) by a new evaluation measure that makes use of a probabilistic reference alignment ("Probabilistic" in Table 47). This probabilistic measure is based on a gold standard which is manually and independently generated by several domain experts. The number of votes of these annotators are applied as support values in the probabilistic evaluation. For a detailed discussion, please refer to [28].

11.2 Results

Table 47 summarises the results of our evaluation. "P" abbreviates precision, "R" is recall, "FM" stands for F-measure and "Rk" means rank. The prefix "Pro" indicates the probabilistic versions of the precision, recall, F-measure and the associated rank. These metrics are explained below. Participants of the Process Model Matching Contest in 2015 (PMMC 2015) are depicted in grey font, while OAEI 2016 participants are shown in black font. The OAEI participants are ranked on position 1, 8, 9 and 11 with an overall number of 16 systems listed in the table (when using the standard metrics). Note that AML-PM at the PMMC 2015 was a matching system that was based on a predecessor of AML participating at OAEI 2016. The good results of AML are surprising, since we expected that matching systems specifically developed for the purpose of process model matching would outperform ontology matching systems applied to the special case of process model matching. While AML contains also components that are specifically designed for the process matching task (a flooding-like structural matching algorithm), its relevant main components are components developed for ontology matching and the sub-problem of instance matching.

Participants			Standard				Probabilistic			
Matcher	**Contest**	**Size**	**P**	**R**	**FM**	**Rk**	**ProP**	**ProR**	**ProFM**	**Rk**
AML	OAEI-16	221	0,719	0,685	0,702	1	0,742	0,283	0,410	2
AML-PM	PMMC-15	579	0,269	0,672	0,385	14	0,377	0,398	0,387	4
BPLangMatch	PMMC-15	277	0,368	0,440	0,401	12	0,532	0,272	0,360	8
DKP	OAEI-16	177	0,621	0,474	0,538	8	0,686	0,219	0,333	9
DKP*	OAEI-16	150	0,680	0,440	0,534	9	0,772	0,211	0,331	10
KnoMa-Proc	PMMC-15	326	0,337	0,474	0,394	13	0,506	0,302	0,378	5
KMatch-SSS	PMMC-15	261	0,513	0,578	0,544	6	0,563	0,274	0,368	7
LogMap	OAEI-16	267	0,449	0,517	0,481	11	0,594	0,291	0,390	3
Match-SSS	PMMC-15	140	0,807	0,487	0,608	4	0,761	0,192	0,307	12
OPBOT	PMMC-15	234	0,603	0,608	0,605	5	0,648	0,258	0,369	6
pPalm-DS	PMMC-15	828	0,162	0,578	0,253	16	0,210	0,335	0,258	16
RMM-NHCM	PMMC-15	220	0,691	0,655	0,673	2	0,783	0,297	0,431	1
RMM-NLM	PMMC-15	164	0,768	0,543	0,636	3	0,681	0,197	0,306	13
RMM-SMSL	PMMC-15	262	0,511	0,578	0,543	7	0,516	0,242	0,329	11
RMM-VM2	PMMC-15	505	0,216	0,470	0,296	15	0,309	0,294	0,301	14
TripleS	PMMC-15	230	0,487	0,483	0,485	10	0,486	0,210	0,293	15

Table 47. Results of the process model matching track

In the probabilistic evaluation, however, the OAEI participants gain position 2, 3, 9 and 10, respectively. LogMap rises from position 11 to 3. The (probabilistic) precision improves over-proportionally for this matcher, because LogMap generates many correspondences which are not included in the binary gold standard but are included in the probabilistic one. The ranking of LogMap demonstrates that a strength of the probabilistic metric lies in the broadened definition of the gold standard where weak mappings are included but softened (via the support values).

Figures 11(a)-(b) show the probabilistic precision (ProP) and the probabilistic recall (ProR) with rising threshold τ on the reference alignment (0,000; 0,375; 0,500; 0,750). The matcher LogMap mainly identifies correspondences with high support (of which many are not included in the binary gold standard). This can be observed by the minor change in the ProP and the significant increase in the ProR with higher τ. For the matcher AML, the opposite effect can be observed. The ProP decreases dramatically with rising τ (accompanied by a weak increase of the ProR). This indicates that the matcher computes a high fraction of correspondences with low support value (which are partly included in the binary gold standard). For the matchers DKP and DKP*, with increasing τ, a minor decrease in ProP and increase in ProR can be observed. The ProP decreases, since the number of correspondences in the non-binary gold standard decreases (with rising τ). At the same time, the ProR increases with a lower number of correspondences (with rising τ). Figure 11(c) displays the probabilistic F-measure (ProFM) with rising threshold τ on the reference alignment. AML achieves best results with τ = 0,375 since this matcher identifies a high fraction of correspondences with low support value (which can also be trivial correspondences). For details about the probabilistic metric, please refer to [28].

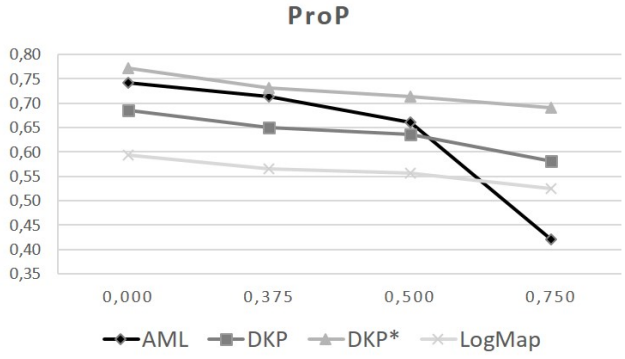
The results depicted in Table 47 and Figure 11 indicate that the progress made in ontology matching has also a positive impact on other related matching problems, like it is the case for process model matching. While it might require to reconfigure, adapt, and extend some parts of the ontology matching systems, such a system seems to offer a good starting point which can be turned with a reasonable amount of work into a good process matching tool. We have to emphasise that our observations are so far based on only one data set. Moreover, only three participants decided to apply their systems to the new track of process model matching. Thus, we have to be cautious to generalise the results we observed so far. In the future we might be able to attract more participants integrating more data sets in the evaluation.

12 Lesson learned and suggestions

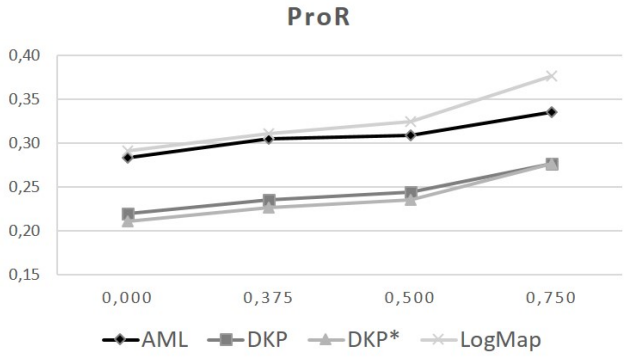
The lessons learned from running OAEI 2016 were the following:

A) This year, as suggested in previous campaigns, we requested tool registration in June and preliminary submission of wrapped systems by the end of July. This measure was successful in reducing the number of systems with errors and incompatibilities with the SEALS client during the evaluation phase as had happened in the past. However, not all systems complied with the deadlines, and some did have problems, which still delayed the evaluation. In future editions, we must be more strict in enforcing the participation protocol.

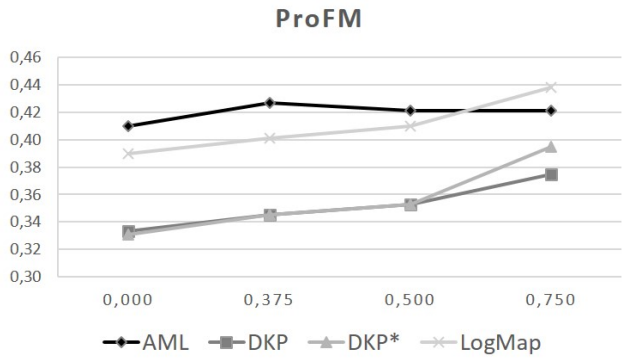
B) Thanks in part to the new submission schedule, this marked the first OAEI edition where all participants and all tracks were evaluated using the SEALS client. Nevertheless, some system developers still struggled to get their systems working with the client, mostly due to incompatible versions of libraries. This recurring problem, plus the effort required to update the SEALS client's libraries, lead to the consideration of whether it would not be better to develop a simpler, more streamlined evaluation solution.



(a) Probabilistic precision



(b) Probabilistic recall



(c) Probabilistic F-measure

Fig. 11. Change in metric values with rising threshold τ.

C) The continued absence of the SEALS web portal did not seem to affect participation, as the Google drive solution for submission was well received by the participants. OAEI may move towards a cloud-based solution.
D) While the number of participants this year was similar to that of recent years, their distribution through the tracks was uneven. Long-standing tracks had no shortage of participants, but alas the same was not true for the Interactive, Process Model (new) or Instance (new data sets) tracks. One reason for this is that the OAEI data sets have been released too close to the submission deadline to allow system developers to develop their systems to tackle them all—the timing is barely sufficient to allow serious development focusing on one new data set. Thus, with prize money on offer on one of the new tracks, it is no surprise that system developers were polarised towards that track and eschewed the other new ones. We should consider anticipating the deadline for initial release of OAEI data sets, particular for those that are new, in order to give system developers more time to tackle them, thereby increasing participation.
E) The increasing variety of OAEI tracks also poses difficulties to system developers in configuring their systems to handle different types of tasks. It is noteworthy that only two systems, both of which are long-term OAEI participants, have tackled all tracks—and one of them did so using external configuration files specifying the type of task. One solution to facilitate participation in multiple tracks would be to have the evaluation client transmit to the system the specifications of the task, e.g., whether classes, properties, and/or individuals are to be matched, and whether only a specific subset of them are to be matched. This would also make the tasks more realistic, in the sense that in normal use, a user would provide to the ontology matching system this type of information.
F) With regard to the low participation in the Process Model and Instance tracks, it merits considering whether enforcing adherence to the SEALS client and ontology-based data sets were not deterrent factors. It should be noted that the Process Model Matching Contest (PMMC) received a much larger number of participants in 2015 than did the Process Model track, and that there is a considerable number of publications on data interlinking systems, but only one of these participated in the Instance track.
G) In previous years we identified the need for considering non-binary forms of evaluation, namely in cases where there is uncertainty about some of the reference mappings. A first non-binary evaluation type was implemented in last year's Conference track, but this year two new tracks followed suit: Disease and Phenotype where the evaluation was semantic, and Process Model, where it was probabilistic. These new strategies should provide a fairer evaluation of the systems in complex test cases.

The lessons learned in the various OAEI 2016 track were the following:

largebio: While the current reference alignments, with incoherence-causing mappings flagged as uncertain, make the evaluation fair to all systems, they are only a compromise solution, not an ideal one. Thus, we should aim for manually repairing and validating the reference alignments for future editions.

phenotype: The prize offered in this track, thanks to the kind sponsorship of the Pistoia Alliance Ontologies Mapping project, was positively accepted by the community and helped attract new participants. However, it also had a polarising effect, with some systems focusing exclusively in this track. In future editions, we will consider including a prize across OAEI tracks in order to motivate developers to successfully participate in more than one track.

interactive: The new functionality of the Oracle allowing systems to submit a set of up to three conflicting mappings, rather than a mapping at a time, was successfully exploited by one new participating system. Nevertheless, this track's participation has remained low, as most systems participating in OAEI focussed exclusively on fully automatic matching. We hope to draw more participants to this track in the future and will continue to expand it so as to better approximate real user interactions.

process model: The results of the new Process Model track have shown that the participating ontology matching systems are capable of generating very good results for the specific problem of process model matching. This shows that the basic components of an ontology matching system can also be successfully applied to other kind of matching problems.

instance: In order to attract more instance matching systems to participate in value semantics (val-sem), value structure (val-struct), and value structure semantics (val-struct-sem) tasks, we need to produce benchmarks that have fewer instances (in the order of 10000), of the same type (in our benchmark we asked systems to compare instances of different types). To balance those aspects, we must then produce data sets with more complex transformations.

13 Conclusions

OAEI 2016 saw the same number (21) of participants as in recent years, with a healthy mix of new and returning systems. While some new participants were mainly drawn by the allure of prize money in the new Disease and Phenotype track, the very fact that there was prize money on offer shows that interest in ontology matching is not waning, which bodes well for the future of OAEI. All the test cases were performed on the SEALS client, including those in the instance matching track, which is good news regarding the interoperability of matching systems. Furthermore, the fact that the SEALS client can be used for such a variety of tasks is a good sign of its relevance.

Unlike previous years, this year there was no noticeable improvement with regard to system run times—for instance, the distribution of run times in Anatomy and Large Biomedical Ontologies was approximately the same as last year. There was also no progress with regard to the ability to handle large ontologies and data sets, as the number of systems able to cope with the Large Biomedical Ontologies data set in full was the same as last year, and all systems able to cope with the Instance Synthetic data set were established systems already known for their ability to handle large data sets. Finally, there was no progress with regard to alignment repair systems, with only a few returning systems employing them. As a consequence, incoherent alignments are common.

With regard to F-measure, some returning systems showed substantial improvements, but overall, the improvements in F-measure were subtle in Anatomy and Large Biomedical Ontologies, and non-existent in Conference. As has been the trend, most systems favour precision over recall.

Most of the participants have provided a description of their systems and their experience in the evaluation. These OAEI papers, like the present one, have not been peer reviewed. However, they are full contributions to this evaluation exercise and reflect the hard work and clever insight people put into the development of participating systems. Reading the papers of the participants should help people involved in ontology matching find out what makes these algorithms work and what could be improved.

The Ontology Alignment Evaluation Initiative will strive to continue to be a reference to the ontology matching community by improving both the test cases and the testing methodology to better reflect the actual needs of the community. Evaluating ontology matching systems remains a challenging but critical topic, which is essential to enable the progress of this field [38]. More information can be found at:

`http://oaei.ontologymatching.org`.

Acknowledgements

We warmly thank the participants of this campaign. We know that they have worked hard to have their matching tools executable in time and they provided useful reports on their experience. The best way to learn about the results remains to read the papers that follow.

We would also like to thank the Pistoia Alliance[9] which sponsored the Disease and Phenotype track and funded the prize for the winners.

We are very grateful to the Universidad Politécnica de Madrid (UPM), especially to Nandana Mihindukulasooriya and Asunción Gómez Pérez, for moving, setting up and providing the necessary infrastructure to run the SEALS repositories.

We are also grateful to Martin Ringwald and Terry Hayamizu for providing the reference alignment for the anatomy ontologies and thank Elena Beisswanger for her thorough support on improving the quality of the data set.

We thank Khiat Abderrahmane for his support in the Arabic data set and Catherine Comparot for her feedback and support in the MultiFarm test case.

We also thank for their support the other members of the Ontology Alignment Evaluation Initiative steering committee: Yannis Kalfoglou (Ricoh laboratories, UK), Miklos Nagy (The Open University (UK), Natasha Noy (Stanford University, USA), Yuzhong Qu (Southeast University, CN), York Sure (Leibniz Gemeinschaft, DE), Jie Tang (Tsinghua University, CN), George Vouros (University of the Aegean, GR).

Michelle Cheatham has been supported by the National Science Foundation award ICER-1440202 "EarthCube Building Blocks: Collaborative Proposal: GeoLink".

Jérôme Euzenat, Ernesto Jimenez-Ruiz, Christian Meilicke, Heiner Stuckenschmidt and Cássia Trojahn dos Santos have been partially supported by the SEALS (IST-2009-238975) European project in previous years.

Daniel Faria was supported by the ELIXIR-EXCELERATE project (INFRADEV-3-2015).

Ernesto Jimenez-Ruiz has also been partially supported by the Seventh Framework Program (FP7) of the European Commission under Grant Agreement 318338, "Optique", the EPSRC projects DBOnto and ED3, the Research Council of Norway project BigMed, and the Centre for Scalable Data Access (SIRIUS).

Catia Pesquita was supported by the FCT through the LASIGE Strategic Project (UID/CEC/00408/2013) and the research grant PTDC/EEI-ESS/4633/2014.

Ondřej Zamazal has been supported by the CSF grant no. 14-14076P.

References

1. Manel Achichi, Rodolphe Bailly, Cécile Cecconi, Marie Destandau, Konstantin Todorov, and Raphaël Troncy. Doremus: Doing reusable musical data. In *ISWC PD: International Semantic Web Conference Posters and Demos*, 2015.
2. José Luis Aguirre, Bernardo Cuenca Grau, Kai Eckert, Jérôme Euzenat, Alfio Ferrara, Robert Willem van Hague, Laura Hollink, Ernesto Jiménez-Ruiz, Christian Meilicke, Andriy Nikolov, Dominique Ritze, François Scharffe, Pavel Shvaiko, Ondrej Sváb-Zamazal, Cássia Trojahn, and Benjamin Zapilko. Results of the ontology alignment evaluation initiative 2012. In *Proc. 7th ISWC ontology matching workshop (OM), Boston (MA US)*, pages 73–115, 2012.
3. Goncalo Antunes, Marzieh Bakhshandeh, Jose Borbinha, Joao Cardoso, Sharam Dadashnia, Chiara Di Francescomarino, Mauro Dragoni, Peter Fettke, Avigdor Gal, Chiara Ghidini, Philip Hake, Abderrahmane Khiat, Christopher Klinkmüller, Elena Kuss, Henrik Leopold, Peter Loos, Christian Meilicke, Tim Niesen, Catia Pesquita, Timo Péus, Andreas Schoknecht, Eitam Sheetrit, Andreas Sonntag, Heiner Stuckenschmidt, Tom Thaler, Ingo Weber, and Matthias Weidlich. The process model matching contest 2015. In *6th International Workshop on Enterprise Modelling and Information Systems Architectures, September 3-4, 2015 Innsbruck, Austria*, pages 127–155, 2015.
4. Benjamin Ashpole, Marc Ehrig, Jérôme Euzenat, and Heiner Stuckenschmidt, editors. *Proc. K-Cap Workshop on Integrating Ontologies*, Banff (Canada), 2005.
5. Olivier Bodenreider. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32:267–270, 2004.
6. Caterina Caracciolo, Jérôme Euzenat, Laura Hollink, Ryutaro Ichise, Antoine Isaac, Véronique Malaisé, Christian Meilicke, Juan Pane, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Sváb-Zamazal, and Vojtech Svátek. Results of the ontology alignment evaluation initiative 2008. In *Proc. 3rd ISWC ontology matching workshop (OM), Karlsruhe (DE)*, pages 73–120, 2008.
7. Silvana Castano, Alfio Ferrara, Lorenzo Genta, and Stefano Montanelli. Combining Crowd Consensus and User Trustworthiness for Managing Collective Tasks. *Future Generation Computer Systems*, 54, 2016.
8. Michelle Cheatham, Zlatan Dragisic, Jérôme Euzenat, Daniel Faria, Alfio Ferrara, Giorgos Flouris, Irini Fundulaki, Roger Granada, Valentina Ivanova, Ernesto Jiménez-Ruiz, et al. Results of the ontology alignment evaluation initiative 2015. In *10th ISWC workshop on ontology matching (OM)*, pages 60–115, 2015.
9. Bernardo Cuenca Grau, Zlatan Dragisic, Kai Eckert, Jérôme Euzenat, Alfio Ferrara, Roger Granada, Valentina Ivanova, Ernesto Jiménez-Ruiz, Andreas Oskar Kempf, Patrick Lambrix, Andriy Nikolov, Heiko Paulheim, Dominique Ritze, François Scharffe, Pavel Shvaiko, Cássia Trojahn dos Santos, and Ondrej Zamazal. Results of the ontology alignment evaluation initiative 2013. In Pavel Shvaiko, Jérôme Euzenat, Kavitha Srinivas, Ming Mao, and Ernesto Jiménez-Ruiz, editors, *Proc. 8th ISWC workshop on ontology matching (OM), Sydney (NSW AU)*, pages 61–100, 2013.
10. Jim Dabrowski and Ethan V. Munson. 40 years of searching for the best computer system response time. *Interacting with Computers*, 23(5):555–564, 2011.
11. Jérôme David, Jérôme Euzenat, François Scharffe, and Cássia Trojahn dos Santos. The alignment API 4.0. *Semantic web journal*, 2(1):3–10, 2011.
12. Zlatan Dragisic, Kai Eckert, Jérôme Euzenat, Daniel Faria, Alfio Ferrara, Roger Granada, Valentina Ivanova, Ernesto Jiménez-Ruiz, Andreas Oskar Kempf, Patrick Lambrix, Stefano Montanelli, Heiko Paulheim, Dominique Ritze, Pavel Shvaiko, Alessandro Solimando, Cássia Trojahn dos Santos, Ondrej Zamazal, and Bernardo Cuenca Grau. Results of the

ontology alignment evaluation initiative 2014. In *Proceedings of the 9th International Workshop on Ontology Matching collocated with the 13th International Semantic Web Conference (ISWC), Riva del Garda, Trentino, Italy*, pages 61–104, 2014.

13. Zlatan Dragisic, Valentina Ivanova, Patrick Lambrix, Daniel Faria, Ernesto Jiménez-Ruiz, and Catia Pesquita. User validation in ontology alignment. In *The Semantic Web - ISWC 2016 - 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016, Proceedings, Part I*, pages 200–217, 2016.
14. Jérôme Euzenat, Alfio Ferrara, Laura Hollink, Antoine Isaac, Cliff Joslyn, Véronique Malaisé, Christian Meilicke, Andriy Nikolov, Juan Pane, Marta Sabou, François Scharffe, Pavel Shvaiko, Vassilis Spiliopoulos, Heiner Stuckenschmidt, Ondrej Sváb-Zamazal, Vojtech Svátek, Cássia Trojahn dos Santos, George Vouros, and Shenghui Wang. Results of the ontology alignment evaluation initiative 2009. In *Proc. 4th ISWC ontology matching workshop (OM), Chantilly (VA US)*, pages 73–126, 2009.
15. Jérôme Euzenat, Alfio Ferrara, Christian Meilicke, Andriy Nikolov, Juan Pane, François Scharffe, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Sváb-Zamazal, Vojtech Svátek, and Cássia Trojahn dos Santos. Results of the ontology alignment evaluation initiative 2010. In *Proc. 5th ISWC ontology matching workshop (OM), Shanghai (CN)*, pages 85–117, 2010.
16. Jérôme Euzenat, Alfio Ferrara, Robert Willem van Hague, Laura Hollink, Christian Meilicke, Andriy Nikolov, François Scharffe, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Sváb-Zamazal, and Cássia Trojahn dos Santos. Results of the ontology alignment evaluation initiative 2011. In *Proc. 6th ISWC ontology matching workshop (OM), Bonn (DE)*, pages 85–110, 2011.
17. Jérôme Euzenat, Antoine Isaac, Christian Meilicke, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Svab, Vojtech Svatek, Willem Robert van Hage, and Mikalai Yatskevich. Results of the ontology alignment evaluation initiative 2007. In *Proc. 2nd ISWC ontology matching workshop (OM), Busan (KR)*, pages 96–132, 2007.
18. Jérôme Euzenat, Christian Meilicke, Pavel Shvaiko, Heiner Stuckenschmidt, and Cássia Trojahn dos Santos. Ontology alignment evaluation initiative: six years of experience. *Journal on Data Semantics*, XV:158–192, 2011.
19. Jérôme Euzenat, Malgorzata Mochol, Pavel Shvaiko, Heiner Stuckenschmidt, Ondrej Svab, Vojtech Svatek, Willem Robert van Hage, and Mikalai Yatskevich. Results of the ontology alignment evaluation initiative 2006. In *Proc. 1st ISWC ontology matching workshop (OM), Athens (GA US)*, pages 73–95, 2006.
20. Jérôme Euzenat, Maria Rosoiu, and Cássia Trojahn dos Santos. Ontology matching benchmarks: generation, stability, and discriminability. *Journal of web semantics*, 21:30–48, 2013.
21. Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer-Verlag, Heidelberg (DE), 2nd edition, 2013.
22. Daniel Faria, Ernesto Jiménez-Ruiz, Catia Pesquita, Emanuel Santos, and Francisco M. Couto. Towards Annotating Potential Incoherences in BioPortal Mappings. In *13th International Semantic Web Conference*, volume 8797 of *Lecture Notes in Computer Science*, pages 17–32. Springer, 2014.
23. Valentina Ivanova, Patrick Lambrix, and Johan Åberg. Requirements for and evaluation of user support for large-scale ontology alignment. In *The Semantic Web. Latest Advances and New Domains 12th European Semantic Web Conference, ESWC 2015, Portoroz, Slovenia, May 31 – June 4, 2015. Proceedings*, pages 3–20, 2015.
24. Ernesto Jiménez-Ruiz and Bernardo Cuenca Grau. LogMap: Logic-based and scalable ontology matching. In *Proc. 10th International Semantic Web Conference (ISWC), Bonn (DE)*, pages 273–288, 2011.
25. Ernesto Jiménez-Ruiz, Bernardo Cuenca Grau, Ian Horrocks, and Rafael Berlanga. Logic-based assessment of the compatibility of UMLS ontology sources. *J. Biomed. Sem.*, 2, 2011.

26. Ernesto Jiménez-Ruiz, Christian Meilicke, Bernardo Cuenca Grau, and Ian Horrocks. Evaluating mapping repair systems with large biomedical ontologies. In *Proc. 26th Description Logics Workshop*, 2013.
27. Yevgeny Kazakov, Markus Krötzsch, and Frantisek Simancik. Concurrent classification of EL ontologies. In *Proc. 10th International Semantic Web Conference (ISWC), Bonn (DE)*, pages 305–320, 2011.
28. Elena Kuss, Henrik Leopold, Han Van der Aa, Heiner Stuckenschmidt, and Hajo A. Reijers. Probabilistic evaluation of process model matching techniques. In *Lecture notes in computer science. Conceptual modeling: 35th international conference, ER 2016, Gifu, Japan, November 14-17, 2016*, pages 279–292, 2016.
29. Pasquale Lisena, Manel Achichi, Eva Fernández, Konstantin Todorov, and Raphaël Troncy. Exploring linked classical music catalogs with overture. In *ISWC PD: International Semantic Web Conference Posters and Demos*, 2016.
30. L. Ma, Y. Yang, Z. Qiu, G, Xie, Y. Pan, and S. Liu. Towards a Complete OWL Ontology Benchmark. In *ESWC*, 2006.
31. Christian Meilicke. *Alignment Incoherence in Ontology Matching*. PhD thesis, University Mannheim, 2011.
32. Christian Meilicke, Raúl García Castro, Frederico Freitas, Willem Robert van Hage, Elena Montiel-Ponsoda, Ryan Ribeiro de Azevedo, Heiner Stuckenschmidt, Ondrej Sváb-Zamazal, Vojtech Svátek, Andrei Tamilin, Cássia Trojahn, and Shenghui Wang. MultiFarm: A benchmark for multilingual ontology matching. *Journal of web semantics*, 15(3):62–68, 2012.
33. Boris Motik, Rob Shearer, and Ian Horrocks. Hypertableau reasoning for description logics. *Journal of Artificial Intelligence Research*, 36:165–228, 2009.
34. Heiko Paulheim, Sven Hertling, and Dominique Ritze. Towards evaluating interactive ontology matching tools. In *Proc. 10th Extended Semantic Web Conference (ESWC), Montpellier (FR)*, pages 31–45, 2013.
35. Catia Pesquita, Daniel Faria, Emanuel Santos, and Francisco Couto. To repair or not to repair: reconciling correctness and coherence in ontology reference alignments. In *Proc. 8th ISWC ontology matching workshop (OM), Sydney (AU)*, pages 13–24, 2013.
36. Tzanina Saveta, Evangelia Daskalaki, Giorgos Flouris, Irini Fundulaki, Melanie Herschel, and Axel-Cyrille Ngonga Ngomo. Lance: Piercing to the heart of instance matching tools. In *International Semantic Web Conference*, pages 375–391. Springer, 2015.
37. Tzanina Saveta, Evangelia Daskalaki, Giorgos Flouris, Irini Fundulaki, Melanie Herschel, and Axel-Cyrille Ngonga Ngomo. Pushing the limits of instance matching systems: A semantics-aware benchmark for linked data. In *WWW, Companion Volume*, 2015.
38. Pavel Shvaiko and Jérôme Euzenat. Ontology matching: state of the art and future challenges. *IEEE Transactions on Knowledge and Data Engineering*, 25(1):158–176, 2013.
39. Alessandro Solimando, Ernesto Jiménez-Ruiz, and Giovanna Guerrini. Detecting and correcting conservativity principle violations in ontology-to-ontology mappings. In *The Semantic Web–ISWC 2014*, pages 1–16. Springer, 2014.
40. Alessandro Solimando, Ernesto Jimenez-Ruiz, and Giovanna Guerrini. Minimizing conservativity violations in ontology alignments: Algorithms and evaluation. *Knowledge and Information Systems*, 2016.
41. York Sure, Oscar Corcho, Jérôme Euzenat, and Todd Hughes, editors. *Proc. ISWC Workshop on Evaluation of Ontology-based Tools (EON), Hiroshima (JP)*, 2004.

Montpellier, Dayton, Linköping, Grenoble, Lisboa, Milano, Heraklion,
Kent, Oslo, Oxford, Mannheim, Amsterdam, Trento, Basel, Toulouse, Prague
December 2016