Ontology Matching

OM-2016

Proceedings of the ISWC Workshop

Introduction

Ontology matching[1] is a key interoperability enabler for the semantic web, as well as a useful tactic in some classical data integration tasks dealing with the semantic heterogeneity problem. It takes ontologies as input and determines as output an alignment, that is, a set of correspondences between the semantically related entities of those ontologies. These correspondences can be used for various tasks, such as ontology merging, data translation, query answering or navigation on the web of data. Thus, matching ontologies enables the knowledge and data expressed in the matched ontologies to interoperate.

The workshop has three goals:

- To bring together leaders from *academia*, *industry* and *user institutions* to assess how academic advances are addressing real-world requirements. The workshop strives to improve academic awareness of industrial and final user needs, and therefore, direct research towards those needs. Simultaneously, the workshop serves to inform industry and user representatives about existing research efforts that may meet their requirements. The workshop also investigated how the ontology matching technology is going to evolve.
- To conduct an extensive and rigorous evaluation of ontology matching and instance matching (link discovery) approaches through the OAEI (Ontology Alignment Evaluation Initiative) 2016 campaign[2]. Besides real-world specific matching tasks, involving e.g., large biomedical ontologies, OAEI 2016 introduced the process model matching track as well as a desease-phenotype track supported by the Pistoia Alliance Ontologies Mapping project within a specific matching scenario. Therefore, the ontology matching evaluation initiative itself provided a solid ground for discussion of how well the current approaches are meeting business needs.
- To examine new uses, similarities and differences from database schema matching, which has received decades of attention but is just beginning to transition to mainstream tools.

The program committee selected 6 submissions for oral presentation and 9 submissions for poster presentation. 21 matching systems participated in this year's OAEI campaign. Further information about the Ontology Matching workshop can be found at: `http://om2016.ontologymatching.org/`.

[1] `http://www.ontologymatching.org/`
[2] `http://oaei.ontologymatching.org/2016`

Acknowledgments. We thank all members of the program committee, authors and local organizers for their efforts. We appreciate support from the Trentino as a Lab[3] initiative of the European Network of the Living Labs[4] at Informatica Trentina[5], the EU SEALS (Semantic Evaluation at Large Scale)[6] project and the Pistoia Alliance Ontologies Mapping project[7].

Pavel Shvaiko
Jérôme Euzenat
Ernesto Jiménez-Ruiz
Michelle Cheatham
Oktie Hassanzadeh
Ryutaro Ichise

December 2016

[3] http://www.taslab.eu
[4] http://www.openlivinglabs.eu
[5] http://www.infotn.it
[6] http://www.development.seals-project.eu/
[7] http://www.pistoiaalliance.org/ontologies-mapping-plans-participate-oaei-2016/

Organization

Organizing Committee

Pavel Shvaiko, Informatica Trentina SpA, Italy
Jérôme Euzenat, INRIA & University Grenoble Alpes, France
Ernesto Jiménez-Ruiz, University of Oxford, UK
Michelle Cheatham, Wright State University, USA
Oktie Hassanzadeh, IBM Research, USA
Ryutaro Ichise, National Institute of Informatics, Japan

Program Committee

Alsayed Algergawy, Jena University, Germany
Zohra Bellahsene, LRIMM, France
Olivier Bodenreider, National Library of Medicine, USA
Marco Combetto, Informatica Trentina, Italy
Valerie Cross, Miami University, USA
Isabel Cruz, The University of Illinois at Chicago, USA
Warith Eddine Djeddi, LIPAH & LABGED, Tunisia
Jérôme David, University Grenoble Alpes & INRIA, France
Gayo Diallo, University of Bordeaux, France
Zlatan Dragisic, Linköpings Universitet, Sweden
Alfio Ferrara, University of Milan, Italy
Fausto Giunchiglia, University of Trento, Italy
Wei Hu, Nanjing University, China
Valentina Ivanova, Linköpings Universitet, Sweden
Antoine Isaac, Vrije Universiteit Amsterdam & Europeana, Netherlands
Daniel Faria, Instituto Gulbenkian de Ciência, Portugal
Patrick Lambrix, Linköpings Universitet, Sweden
Juanzi Li, Tsinghua University, China
Vincenzo Maltese, University of Trento, Italy
Fiona McNeill, University of Edinburgh, UK
Christian Meilicke, University of Mannheim, Germany
Andriy Nikolov, Open University, UK
Axel Ngonga, University of Leipzig, Germany
Leo Obrst, The MITRE Corporation, USA
Heiko Paulheim, University of Mannheim, Germany
Catia Pesquita, University of Lisbon, Portugal
Dominique Ritze, University of Mannheim, Germany
Umberto Straccia, ISTI-C.N.R., Italy
Ondřej Zamazal, Prague University of Economics, Czech Republic
Valentina Tamma, University of Liverpool, UK

Cássia Trojahn, IRIT, France
Ludger van Elst, DFKI, Germany
Songmao Zhang, Chinese Academy of Sciences, China

Table of Contents

Technical Papers

OAEI Papers

Posters

Towards Best Practices for Crowdsourcing Ontology Alignment Benchmarks

Reihaneh Amini, Michelle Cheatham, Pawel Grzebala, and Helena B. McCurdy

Data Semantics Laboratory, Wright State University, Dayton, Ohio.
{amini.2, michelle.cheatham, grzebala.2, mccurdy.18}@wright.edu

Abstract. Ontology alignment systems establish the links between ontologies that enable knowledge from various sources and domains to be used by applications in many different ways. Unfortunately, these systems are not perfect. Currently the results of even the best-performing alignment systems need to be manually verified in order to be fully trusted. Ontology alignment researchers have turned to crowdsourcing platforms such as Amazon's Mechanical Turk to accomplish this. However, there has been little systematic analysis of the accuracy of crowdsourcing for alignment verification and the establishment of best practices. In this work, we analyze the impact of the presentation of the context of potential matches and the way in which the question is presented to workers on the accuracy of crowdsourcing for alignment verification.

Keywords: Ontology Alignment, Crowdsourcing, Mechanical Turk

1 Introduction

While the amount of linked data on the Semantic Web has grown dramatically, links between different datasets have unfortunately not grown at the same rate, and data is less useful without context. Links between related things, particularly related things from different datasets, are what enable applications to move beyond individual silos of data towards synthesizing information from a variety of data sources. The goal of ontology alignment is to establish these links by determining when an entity in one ontology is semantically related to an entity in another ontology (for a comprehensive discussion of ontology alignment see [4]).

The performance of automated alignment systems is becoming quite good for certain types of mapping tasks; however, no existing system generates alignments that are completely correct [13]. As a result, there is significant ongoing research on alignment systems that allow users to contribute their knowledge and expertise to the mapping process. Interactive alignment systems exist on a spectrum ranging from entirely manual approaches to semi-automated techniques that ask humans to chime in only when the automated system is unable to make a definitive decision [10]. Because manual alignment is feasible only for small datasets, most research in this area focuses on semi-automated approaches that interact with the user only intermittently. The simplest approach is to send all or a subset of the matches produced through automated techniques to a

user for verification [5]. Other systems only ask the user for guidance at critical decision points during the mapping process, and then attempt to leverage this human-supplied knowledge to improve the scope and quality of the alignment [3].

The issue with the above methods is that ontology engineers and domain experts are very busy people, and they may not have time to devote to manual or semi-automated data integration projects. As a result, some ontology alignment researchers have turned to generic large-scale crowdsourcing platforms, such as Amazon's Mechanical Turk. Although the use of such crowdsourcing platforms to facilitate scalable ontology alignment is becoming quite common, there is some well-founded skepticism regarding the trustworthiness of crowdsourced alignment benchmarks. In this work we depart from existing efforts to improve performance of crowdsourcing alignment approaches (e.g. minimizing time and cost) and instead explore whether or not *design* choices made when employing crowdsourcing have a strong effect on the matching results. In particular, there is concern that the results may be sensitive to how the question is asked. The specific questions we seek to answer in this work are:

Q1: Does providing options beyond simple yes or no regarding the existence of a relationship between two entities improve worker accuracy?
Q2: What is the impact of question type (e.g. true/false versus multiple choice) on workers' accuracy?
Q3: What is the best way to present workers with the contextual information they need to make accurate decisions?
Q4: It is possible to detect scammers who produce inaccurate results on ontology alignment microtasks?

These are all important questions that must be addressed if researchers in the ontology alignment field are going to accept work on ontology alignments evaluated via crowdsourcing or a crowdsourced alignment benchmark as valid. Section 2 of this paper discusses previous research on crowdsourcing in semi-automated ontology alignment systems. In Section 3, we describe our experimental setup and methodology, and in Section 4 we evaluate the results of those experiments with respect to the research questions presented above. Section 5 summarizes the results and discusses plans for future work on this topic.

2 Background and Related Work

We leverage Amazon's Mechanical Turk platform in this work. Amazon publicly released Mechanical Turk in 2005. It is based on the idea that some types of tasks that are currently very difficult for machines to solve but are straightforward for humans. The platform provides a way to submit these types of problems, called Human Interface Tasks (HITs), to thousands of people at once. Anyone with a Mechanical Turk account can solve these tasks. People (called Requesters) who send their tasks to Amazon's servers compensate the people (called Workers or Turkers) who work on the tasks with a small amount of money. Requesters can require that workers have certain qualifications in order to work on their tasks.

For example, workers can be required to be from a certain geographical area, to have performed well on a certain number of HITs previously, or to have passed a qualification test designed by the requester[1].

The primary goal of this work is not to create a crowdsourcing-based ontology alignment system, but rather to begin to determine best practices related to how the crowdsourcing component of such a system should be configured for best results. There has been relatively little research into this topic thus far – most existing work focuses on evaluating the overall performance of a crowdsourcing-based alignment system. An example is CrowdMap, developed in 2012 by Sarasua, Simperl and Noy. This work indicates that working on validation tasks (determining whether or not a given relationship between two entities holds) or identification tasks (finding relationships between entities) are both feasible for workers [12]. Our own previous work has used crowdsourcing to verify existing alignment benchmarks [1] and evaluate the results of an automated alignment system on matching tasks for which no reference alignments are available [2].

The majority of work related to presenting matching questions via a crowdsourcing platform has been done by Mortensen and his colleagues [6–8]. It focused on using crowdsourcing to assess the validity of relationships between entities in a single (biomedical) ontology rather than on aligning two different ontologies, but these goals have much in common. Mortensen noted that in some cases workers who passed qualification tests in order to be eligible to work on the rest of their ontology validation tasks were not necessarily the most accurate, as some of them seemed to rely on their intuition rather than the provided definitions. This led the researchers to try providing the definition of the concepts involved in a potential relationship, which increased the accuracy of workers. The results also indicate that phrasing questions in a positive manner led to better results on the part of workers, e.g. asking whether "A computer is a kind of machine" produced better results than asking whether "Not every computer is a machine."

Our own work on crowdsourcing ontology alignment and the work of Mortensen describe somewhat ad hoc approaches to finding appropriate question presentation formats and screening policies for workers in order to achieve good results. The work presented here differs from previous efforts by conducting a systematic review of a range of options in an attempt to identify some best practices.

3 Experiment Design

This section describes the experimental setup, datasets, and Mechanical Turk configuration in enough detail for other researchers to replicate these results. The code used is available from https://github.com/prl-dase-wsu/Ontology Alignment-mTurk. The ontologies and reference alignments are from the Conference track of the Ontology Alignment Evaluation Initiative (OAEI).[2]

[1] http://docs.aws.amazon.com/AWSMechTurk/latest/AWSMechanicalTurkRequester
[2] http://oaei.ontologymatching.org/2015/

3.1 Potential Matches

In order to evaluate the effect of question type, format, and other parameters on worker accuracy, we established a set of 20 potential matches that workers were asked to verify. These matches are all 1-to-1 equivalence relations between pairs of entities drawn from ontologies within the Conference track of the OAEI. Ten of the 20 potential matches are valid. These were taken from the reference alignments. The remaining ten potential matches are invalid. These were chosen based on the most common mistakes within the alignments produced by the 15 alignment systems from the OAEI that performed better than the baseline. For both the valid and invalid matches, we balanced the number of matches in which the entity labels had high string similarity (e.g. "Topic" and "Research_Topic") and low string similarity (e.g. "Paper" and "Contribution").

Even though all relations are equivalence, some of our tests offered workers a choice of subsumption relationships. Unfortunately, a primary hindrance to ontology alignment research is the lack of any widely accepted benchmark involving more than 1-to-1 equivalence relations. Until such a benchmark is available, we have limited options. However, the main idea behind out approach here was to provide users with more than a yes-or-no choice. This, together with the precision-oriented and recall-oriented interpretation of responses[3], allows researchers to mitigate some of the impacts between people who only answer "yes" in clear-cut cases and those who answer "yes" unless it is obviously not the case.

3.2 Experiment Dimensions

Researchers in this area are so familiar with ontologies and ontology alignment that they risk presenting crowdsourcing workers with questions in a form that makes sense to them but is unintuitive to the uninitiated. We therefore selected the following common methods of alignment presentation for evaluation.

Factor 1: Question Type Previous work has used two different approaches to asking about the relationship between two entities: true/false, in which a person is asked if two entities are equivalent [9], and multiple choice questions, in which the person is asked about the precise relationship between two entities, such as equivalence, subsumption, or no relation [2, 12].

A typical true/false question is "Can Paper be matched with Contribution"? Workers can then simply answer "Yes" or "No." A multiple choice question regarding the same two entities takes the form "What is the relationship between Paper and Contribution?" and has four possible answers:"Paper and Contribution are the same," "Any thing that is a Paper is also a Contribution, but anything that is a Contribution is not necessarily a Paper," "Any thing that is a Contribution is also a Paper, but anything that is a Paper is not necessarily a Contribution" and "There is no relationship between Paper and Contribution." The motivation for the second of these approaches is that as automated alignment systems attempt to move beyond finding 1-to-1 equivalence relationships

[3] These evaluation metrics will be discussed in details in Section 4.1 .

towards identifying subsumption relations and more complex mappings involving multiple entities from both ontologies, the ability to accurately crowdsource information about these more complex relationships becomes more important. Additionally, a common approach taken by many current alignment systems is to identify a pool of potential matches for each entity in an ontology and then employ more computationally intensive similarity comparisons to determine which, if any, of those potential matches are valid. If crowdsourcing were to be used in this manner for semi-automated ontology alignment, one approach might be to use the multiple choice question type to cast a wide net regarding related entities, and then feed those into the automated component of the system.

Factor 2: Question Format A primary purpose of ontologies is to contextualize entities within a domain. Therefore, context is very important when deciding whether or not two entities are related. Even in cases where the entities have the same name or label, they may not be used in the same way. These situations are very challenging for current alignment systems [1]. Providing context is particularly important in crowdsourcing, because workers are not domain experts and so may need some additional information about the entities in order to understand the relation between them. For this reason, we explored the impact of providing workers with four different types of contextual information:

Label Only entity labels (no context) is provided.

Definition A definition of each entity's label is provided. Definitions were obtained from Wiktionary.[4] If a label had multiple definitions, the one most related to conferences (the domain of the ontologies) was manually selected.[5]

Relationships (Textual) The worker is presented with a textual description of all of the super class, sub class, super property, sub property, domain and range relationships involving the entities. The axioms specifying these relations were extracted from the ontologies and "translated" using Open University's OWL to English tool.[6] An example for "Evaluated_Paper" is:

- *No camera ready paper is an evaluated paper.*
- *An accepted paper is an evaluated paper.*
- *A rejected paper is an evaluated paper.*
- *An evaluated paper is an assigned paper.*

Relationships (Graphical) The worker is presented with the same information as above, but as a graph rather than as text. The relationships involving both entities from the potential match are shown in the same graph, with an edge labeled "equivalent?" between the entities in question. Figure 1 shows an example for "Place."

[4] https://en.wiktionary.org/wiki/Wiktionary:Main_Page

[5] Note that the goal of this work is to determine the best way in which to prevent matching-related questions rather than to create a fully automated approach; however, the step of choosing the most relevant definition of a label could be automated in future work.

[6] http://swat.open.ac.uk/tools

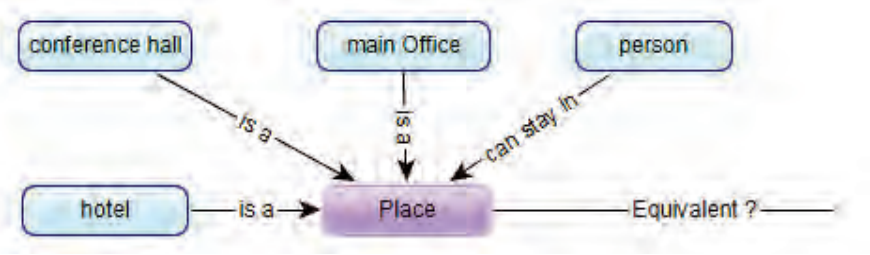


Fig. 1: Graphical depiction of the relationships involving the entity "Place"

3.3 Mechanical Turk Setup

We tested all combinations of question type and format described above, for a total of 8 treatment groups. HITs for each of these tests contained the 20 questions described in Section 3.1. 160 workers were divided among the treatment groups. They were paid 20 cents to complete the task.

One important missing point in current related work is whether workers were prevented from participating in more than one treatment group of the experiment, a potential source of bias. For example, if workers participate in the

Fig. 2: An example of multiple choice HIT containing entity definitions on Amazon's Mechanical Turk server

definitions treatment group and then work on the graphical relationships tasks, they may remember some definitions and that may influence their answers. In order to avoid this source of bias, we created a Mechanical Turk qualification, assigned this to any worker who completed one of our HITs and specified that our HITs were only available to workers who did *not* possess this qualification.

Finding capable and diligent workers is always a difficult problem when using a crowdsourcing platform. One common approach is to require a worker to pass a qualification test before they are allowed to work on the actual tasks. Although this strategy seems quite reasonable, qualification tasks are generally short and

contain only basic questions, so a worker's performance on it is not always reflective of their performance on the actual tasks. Furthermore, sometimes workers will take the qualification task very seriously but then not apply the same level of diligence to the actual tasks. Additionally, workers tend to expect to be compensated more if they had to pass a qualification test. Another approach to attracting good workers is to offer a bonus for good performance [14]. Many requesters also use "candy questions" that have an obviously correct answer, in order to detect bots or people who have just randomly clicked answers without reading the questions. Requesters generally ignore the entire submission of any worker who misses a candy question. We have employed all of these strategies in the course of this work. The results we obtained from workers who passed a qualification test containing simple questions of the type we intended to study were not encouraging – we qualified workers who achieved greater than 80% accuracy on a qualification test; however, those workers delivered poor performance on the actual tasks (average accuracy 51%). As mentioned previously, other researchers experienced a similar problem [11]. As a result, we decided against using qualification tests and settled on offering workers a $2 bonus if they answered 80% or more of the questions correctly. Of course, this particular strategy is only applicable in situations in which the correct answers to the questions are known in advance. In the future, we plan to more systematically explore the ramifications of different methods for dealing with unqualified, unethical, and lazy workers.

4 Analysis of Results

4.1 Impact of Question Type

Ontology alignments are typically evaluated based on precision (how many of the answers given by a person or system are correct) and recall (how many of the correct answers were given by a person or system). These metrics are based on the number of true positives, false positives and false negatives. The meaning for this is clear when we are discussing 1-to-1 equivalence relations (i.e. in the true/false case) but it is less obvious how to classify each result in the multiple choice case, where subsumption relations are possible. For example, consider the multiple choice question in Figure 2. According to the reference alignment, "Topic" and "Research_Topic" are equivalent. It is therefore clear that if the user selects the first multiple choice option, it should be classified as a true positive, whereas selecting the last option should count as a false negative. But how should the middle two options be classified? Unfortunately, most previous work that allows users to specify either equivalence or subsumption relations is vague about how this is handled [12].

In this work we take two different approaches to classifying results as true positives, false positives, or false negatives. In what we call a **recall-oriented** analysis, we consider a subsumption answer to be effectively the same as an equivalence (i.e. identification of *any* relationship between the entities is considered as agreement with the potential match). In the example above, this would result in the middle two options being considered true positives. This approach

allows us to evaluate how accurate workers are at separating pairs of entities that are related in some way from those that are not related at all. This capability is useful in alignments systems to avoid finding only obvious matches – entities related in a variety of ways to a particular entity can be gathered first and then further processing can filter the set down to only equivalence relations. The other approach, which we call a **precision-oriented** analysis, a subsumption relationship is considered distinct from equivalence (i.e. a potential match is only considered validated by a user if they explicitly state that the two entities are equivalent). This would result in options two and three from the example above being classified as false negatives. This interpretation may be useful for evaluating an alignment system that is attempting to find high-quality equivalence relations between entities, which it may subsequently use as a seed for further processing.

The overall results based on question type provided in Figure 3 show that workers have more balanced precision and recall on True/False questions than on Multiple Choice ones. While this is intuitive [2], it is helpful to have quantitative data for the different question types on the same set of potential matches. Also, some interesting observations can be made based on these results, including:

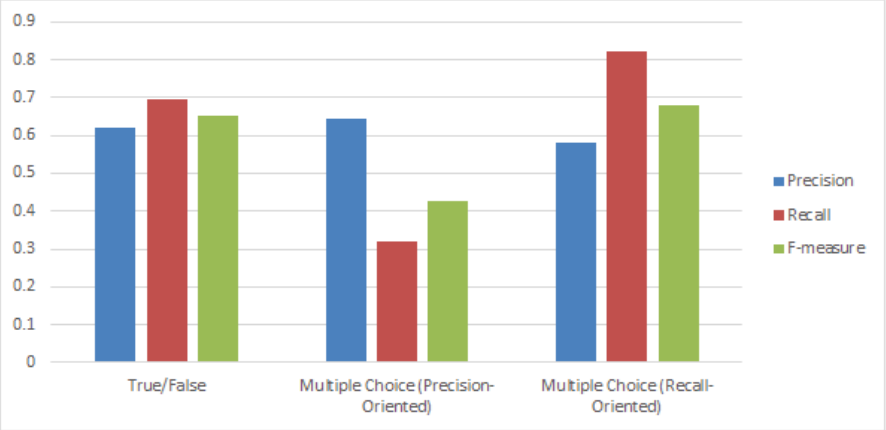


Fig. 3: Workers' performance on true/false and multiple choice questions

Workers are relatively adept at recognizing when *some* type of relationship exists between two entities. The F-measure of 0.65 on the true/false questions and 0.67 using the recall-oriented analysis of the multiple choice questions tells us that workers can fairly accurately distinguish the entities that are somehow related to each other from those that are not, regardless of the question type used to solicit this information from them. In fact, the multiple choice type of question resulted in significantly higher recall (0.82 versus 0.69

for true/false), making it an enticing option for ontology alignment researchers interested in collecting a somewhat comprehensive set of potential matches.

Workers appear to perform poorly at identifying the *type* of relationship that exists between two entities. This claim is less strong than the previous one, because according to our reference alignments, the only relationship that ever held between entities was equivalence. Unfortunately, there are no currently accepted alignment benchmarks that contain subsumption relations, so confirmation of these results is a subject for future work. However, the F-measure of the precision-oriented analysis of the multiple choice questions (0.42, as shown in Figure 3) clearly indicates that the workers did not do well at classifying nuanced relationships between entities.

If precision is paramount, it is best to use true/false questions. While the precision-oriented analysis of the multiple choice questions results is very slightly higher precision than the true/false questions (0.62 versus 0.64), its recall is so low as to be unusable (0.32). If ontology alignment researchers wish to validate 1-to-1 equivalence relationships generated by their system or establish high-quality "anchor" mappings that can be used to seed an alignment algorithm, we recommend that they present their queries to workers as true/false questions.

4.2 Impact of Question Format

As shown in Figure 4, there is a fairly wide range in F-measure for the four question formats, 0.54 to 0.67. Within a single question type, for example true/false, the F-measure varies from 0.59 when no context is provided to 0.73 when workers are provided with the definitions of both terms. This is somewhat surprising, since the domain covered by these ontologies is not particularly esoteric or likely to contain many labels that people are not already familiar with. We note the following observations related to this experiment.

Workers leverage contextual information when it is provided, and this improves their accuracy. Other researchers have speculated that workers may rely on their intuition more than the provided information to complete this type of task, but that hypothesis is not supported by the results here – there is a distinct difference in precision, recall, and F-measure when workers have some contextual information than when they are forced to decide without any context.

When precision is important, providing workers with definitions is effective. The previous section indicated that when the task is to accurately identify equivalent entities, the True-False question style is the best approach. Now Figure 4 indicates that the best accuracy in this situation occurs when workers are provided with entity definitions (F-measure 0.73), while the worst case is when workers are given a piece of the ontology's schema or just the entities' names (F-measure 0.61 and 0.58, respectively).

When finding entity pairs that have *any* relationship is the goal, a graphical depiction is helpful. The recall-oriented analysis of multiple choice questions showed relatively high recall and F-measure for all question formats, with recall of the graphical format slightly edging out that of label definitions. Furthermore, by calculating the True Negative Rate (TNR) of these different formats for multiple choice questions, we discovered that when provided with a graphical depiction of entity relationships, workers more accurately identified when the two entities in the potential match were not related at all (TNR 0.70).

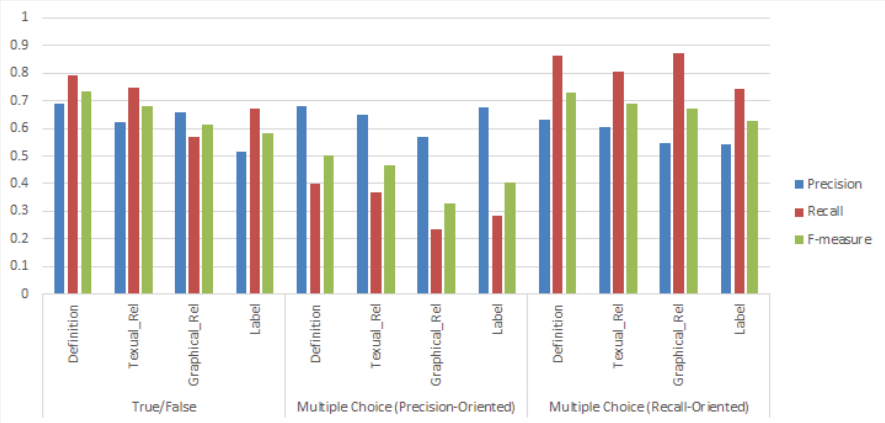


Fig. 4: Workers' performance based on question format

4.3 Dealing with Scammers

Avoiding or handling scammers (people who try to optimize their earnings per time spent) is a recurring theme in crowdsourcing-related subjects. During the presentation of the authors' own work related to crowdsourcing in ontology alignment [1], several attendees expressed the notion that time is likely a useful feature with which to recognize scammers. The intuition is that scammers rush through tasks and quickly answer all of the questions without taking the time to understand and consider each one. To test this hypothesis, we examined the relationship between the time workers spent on a HIT and their accuracy across all question types and formats. For this, we used the "Accept" and "Submit" timestamps included with the Mechanical Turk results available from Amazon. Following is a list of our observations based on this data.

Time spent on a task is a poor indicator of accuracy. We first looked at the average time spent on the HIT by high-performing workers (those who answered more than 80% of the questions within the HIT correctly) and low-performing workers (those who answered fewer than half of the questions correctly). The results were unexpected: high-performing workers spent less than five minutes on the task while low-performers averaged seven minutes.

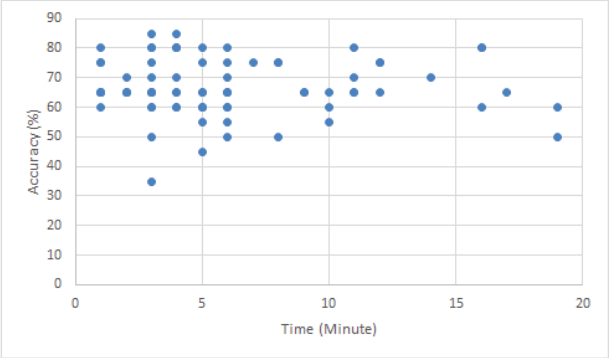


Fig. 5: Average accuracy of workers based on time spent

The above observation holds even at the extreme ends of the time spectrum. Even workers who answered all 20 questions in an extremely short time, such as one or two minutes, did not always have poor accuracy. For instance, multiple workers who spent less than a minute on true/false questions had an accuracy between 60% and 70%, which is close to the overall average on that question type. Conversely, several workers who spent more than 8 minutes had an accuracy between 45% and 55%. It therefore seems that setting thresholds for time to recognize scammers is not a viable strategy.

5 Conclusions and Future Work

The idea of using crowdsourcing for ontology alignment has been gaining in popularity over the past several years. However, very little systematic work has yet gone into how best to present potential matches to users and solicit their responses. This work has begun an effort towards establishing some best practices in this area, by exploring the impact of question type and question format on worker accuracy. Additionally, a popular strategy of mitigating the impact of scammers on accuracy was explored. The results of some experiments confirm common intuition (e.g. workers are better able to determine when any relationship exists between two entities than they are at specifying the precise nature of that relationship), while other results refute popularly held beliefs (e.g. scammers cannot be reliably identified solely by the amount of time they spend on a task). Our overall recommendations are that users interested in verifying the accuracy of an existing alignment or establishing high-quality anchor matches from which to expand are likely to achieve the best results by presenting the definitions of the entity labels and allowing workers to respond with true/false to the question of whether or not an equivalence relationship exists. Conversely, if the alignment researcher is interested in finding entity pairs in which *any* relationship holds, they are better off presenting workers with a graphical depiction of the entity relationships and a set of options about the type of relation that exists, if any.

This work is relevant not only to crowdsourcing approaches to ontology alignment, but also to interactive alignment systems, as well as to user interfaces that attempt to display the rationale behind the matches that make up an alignment generated through other means. However, there are other aspects that are specific to crowdsourcing that should be further explored such as, the best way of enticing large numbers of capable workers to complete alignment tasks in a timely manner. We plan to address this challenge in our future work on this topic.

References

1. Cheatham, M., Hitzler, P.: Conference v2.0: An uncertain version of the OAEI Conference benchmark. In: Proceedings of the International Semantic Web Conference, pp. 33–48. Springer (2014)
2. Cheatham, M., Hitzler, P.: The properties of property alignment. In: Proceedings of the 9th International Conference on Ontology Matching. vol. 1317, pp. 13–24. CEUR-WS. org (2014)
3. Cruz, I.F., Stroe, C., Palmonari, M.: Interactive user feedback in ontology matching using signature vectors. In: Proceedings of the International Conference on Data Engineering (ICDE). pp. 1321–1324. IEEE (2012)
4. Euzenat, J., Shvaiko, P.: Ontology Matching, vol. 333. Springer (2007)
5. Kheder, N., Diallo, G.: Servombi at OAEI 2015. Proceedings of the 12th International Workshop on Ontology Matching p. 200 (2015)
6. Mortensen, J., Musen, M.A., Noy, N.F.: Crowdsourcing the verification of relationships in biomedical ontologies. In: Proceedings of the AMIA Annual Symposium (2013)
7. Mortensen, J.M.: Crowdsourcing ontology verification. In: Proceedings of the International Semantic Web Conference, pp. 448–455. Springer (2013)
8. Mortensen, J.M., Musen, M.A., Noy, N.F.: Ontology quality assurance with the crowd. In: AAAI Conference on Human Computation and Crowdsourcing (2013)
9. Noy, N.F., Mortensen, J., Musen, M.A., Alexander, P.R.: Mechanical Turk as an ontology engineer?: Using microtasks as a component of an ontology-engineering workflow. In: Proceedings of the 5th Annual ACM Web Science Conference. pp. 262–271. ACM (2013)
10. Noy, N.F., Musen, M.A., et al.: Algorithm and tool for automated ontology merging and alignment. In: Proceedings of the 17th National Conference on Artificial Intelligence (AAAI-00). Available as SMI technical report SMI-2000-0831 (2000)
11. Oleson, D., Sorokin, A., Laughlin, G.P., Hester, V., Le, J., Biewald, L.: Programmatic gold: Targeted and scalable quality assurance in crowdsourcing. Human Computation 11(11) (2011)
12. Sarasua, C., Simperl, E., Noy, N.F.: Crowdmap: Crowdsourcing ontology alignment with microtasks. Proceedings of the International Semantic Web Conference pp. 525–541 (2012)
13. Shvaiko, P., Euzenat, J.: Ontology matching: State of the art and future challenges. IEEE Transactions on Knowledge and Data Engineering 25(1), 158–176 (2013)
14. Wang, J., Ghose, A., Ipeirotis, P.: Bonus, disclosure, and choice: What motivates the creation of high-quality paid reviews? (2012)

Analysing Top-level and Domain Ontology Alignments from Matching Systems

Daniela Schmidt*, Cassia Trojahn†, Renata Vieira*

*Pontifical Catholic University of Rio Grande do Sul (Brazil)
daniela.schmidt@acad.pucrs.br, renata.vieira@pucrs.br
†Université de Toulouse 2 & IRIT (France)
cassia.trojahn@irit.fr

Abstract. Top-level ontologies play an important role in the construction and integration of domain ontologies, providing a well-founded reference model that can be shared across knowledge domains. While most efforts in ontology matching have been particularly dedicated to domain ontologies, the problem of matching domain and top-level ontologies has been addressed to a lesser extent. This is a challenging task, specially due to the different levels of abstraction of these ontologies. In this paper, we present a comprehensive analysis of the alignments between one domain ontology from the OAEI Conference track and three well known top-level ontologies (DOLCE, GFO and SUMO), as generated by a set of matching tools. A discussion of the problem is presented on the basis of the alignments generated by the tools, compared to the analysis of three evaluators. This study provides insights for improving matching tools to better deal with this particular task.

1 Introduction

Guarino [5] classifies ontologies according to their "level of generality": *(i) top-level ontologies* describe very general concepts (e.g., space, time, object, etc.), which are independent of a particular problem or domain. These ontologies, also named upper or foundational ontologies [16], are usually equipped with a rich axiomatic layer; *(ii) domain ontologies* and *task ontologies* that describe, respectively, the entities and other information related to a generic domain (e.g., biology or aeronautic), or a generic task or activity (e.g., diagnosis) by specializing the concepts represented in top-level ontologies; and finally *(iii) application ontologies*, which describe the roles played by domain entities when performing an activity (which are, respectively, described by domain and activity ontologies). While the rich semantics and formalization of top-level ontologies are important requirements for ontology design [11], they act as well as semantic bridges supporting very broad semantic interoperability between ontologies [9,10]. In that sense, they play as well a key role in ontology matching.

However, most efforts in ontology matching have been particularly dedicated to domain ontologies and the problem of matching domain and top-level ontologies has been addressed to a lesser extent. This problem poses different challenges in the field, in particular due to the different levels of abstraction of these ontologies. This is a complex task, even manually, that requires to deeply identify the semantic context of concepts.

It involves going beyond the frontiers of the knowledge encoded in the ontologies and, in particular, the identification of subsumption relations. The latter is largely neglect by most matchers. In fact, when having different levels of abstraction it might be the case that the matching process is rather capable of identify subsumption correspondences than equivalence, since the top ontology has concepts at a higher level. Approaches dealing with this task are mostly based on manual matching [1,12].

This paper tackles the problem of matching domain and top-level ontologies in a different way. We aim at evaluating how a set of available matching tools, applying different matching strategies, performs in this task. Even though they were not exactly developed for that purpose, their output might help us to investigate the problem. We chose three well-known top-level ontologies (DOLCE, GFO, and SUMO) and one domain ontology from the OAEI Conference data set. Nine matching tools have been used in our experiments. Qualitative and quantitative analyses are based on the point of view of three evaluators at each generated alignment. The aim is to provide an analysis of the alignments provided by the tools for the task of aligning ontologies with different levels of abstraction as well as to discuss our insights on the topic and to provide directions for future improvements.

The rest of the paper is organised as follows. §2 introduces top-level ontologies and discusses related work. §3 presents the material and methods used in the experiments, the results and discussion. Finally, §4 concludes the paper and presents future work.

2 Background

2.1 Top-level ontologies

A top-level ontology is a high-level and domain independent ontology. The concepts expressed are intended to be basic and universal to ensure generality and expressiveness for a wide range of domains. It is often characterized as representing common sense concepts and is limited to concepts which are meta, generic, abstract and philosophical. There are two approaches for the use of top-level ontologies [16], *top-down* and *bottom-up*. The top-down approach uses the ontology as a foundation for deriving concepts in the domain ontology. In this way, we take the advantage of the knowledge and experience already expressed in the top-level ontology. In a bottom-up approach, one usually matches a new or existing domain ontology to the top-level ontology. This approach represents more challenges since inconsistencies may exist between domain and top-level ontologies [16]. This paper focuses on the latter approach.

Several top-level ontologies have been proposed in the literature. The reader can refer to [9] for a review of them. Here, we briefly introduce some well-known and largely used top-level ontologies which are used further in our evaluation:

– DOLCE [4]: Descriptive Ontology for Linguistic and Cognitive Engineering has been proposed by Nicola Guarino and his team at LOA (Laboratory for Applied Ontology). DOLCE is the first module of the WonderWeb Foundational Ontologies Library. The focus of the DOLCE is to grasp the underlying categories of human cognitive tasks and the socio-cultural environment. It is an ontology of particulars and include concepts such as abstract quality, abstract region, physical object, process, and so on.

– GFO [6]: General Formal Ontology is a top-level ontology for conceptual modeling that has been proposed by the Onto-Med Research Group. It includes elaborations of categories such as objects, processes, time and space, properties, relations, roles, functions, facts, and situations. The work is in progress on the integration with the notion of levels of reality in order to more appropriately capture entities in the material, mental, and social areas.
– SUMO [13]: Suggested Upper Merged Ontology is an upper level ontology that has been proposed as a starter document for The Standard Upper Ontology Working Group, an IEEE working group of collaborators from the fields of engineering, philosophy, and information science. The SUMO provides definitions for general-purpose terms and acts as a foundation for more specific domain ontologies. It is being used for research and applications in search, linguistics and reasoning.

2.2 Related work

In the literature, we see the growing importance of aligning domain and top-level ontology. Recently, in [14], correspondences between DBPedia ontology and DOLCE-Zero [3], a module of DOLCE, are used to identify inconsistent statements in DBPedia. The authors focus on finding systematic errors or anti-patterns in DBPedia. For this task, they exploit previously established alignments between the DBpedia ontology and DOLCE-Zero. They argued that by aligning these ontologies and by combining reasoning and clustering of the reasoning results, errors affecting statements can be identified at a minimal human workload.

In several proposals, alignments between top and domain ontologies are manually generated. In [12], the authors align a domain ontology describing web services (OWL-S) with DOLCE, in order to overcome conceptual ambiguity, poor axiomatization, loose design and narrow scope of the domain ontology. They developed a core ontology of services to serve as middle level between the foundational and domain ontologies and used a module for DOLCE called Descriptions and Situations (D&S) previously developed. The alignment process has been manually done and combined both bottom-up and top-down approaches. First, they used DOLCE as foundational ontology and extended it with the D&S module. This basis has been then used for developing the core ontology of services. Next, they manually aligned OWL-S to the core ontology.

In [1], two domain ontologies of GeoScience (GeoSciML and SWEET) were manually aligned with DOLCE Lite. The authors discussed about the matter of aligning foundational ontologies with these domain ones as a basis for integrating knowledge in this specific domain. The aim is to produce a unified ontology in which both GeoSciML and SWEET are aligned to DOLCE. The alignment process was done in two steps. First, each domain ontology was individually aligned with DOLCE. Then, both ontologies were manually aligned to each other.

In [17], a manually generated alignment between a upper and a biomedical ontology is used for filtering out correspondences at domain level that relate two different kinds of ontology entities. The matching approach is based on a set of similarity measures and the use of top-level ontologies as a parameter for better understanding the conceptual nature of terms within the similarity calculation step. That allows for reducing the possibility of associations between terms derived from different categories. A set of initial

experiments showed an improvement on the alignment quality when using this kind of approach. Evaluation of the generated correspondences has been manually done.

A closer approach to ours has been presented in [7,8], where a repository of ontologies called ROMULUS aims at improving semantic interoperability between foundational ontologies. In order to provide the alignments available in ROMULUS, the authors aligned three foundational ontologies (DOLCE, BFO and GFO) with each other in a semi-automatic way. The alignment process used seven available matching tools (H-Match, PROMPT, LogMap, YAM++, HotMatch, Hertuda, Optima). The resulting manual alignment consists of 35 manual correspondences between DOLCE and GFO, 17 between DOLCE and BFO, and 23 between BFO and GFO. It has been used as a gold standard for comparison with the output of the tools. However, here we focus on the alignment of top and domain ontologies.

Analysing the impact of using top ontologies as semantic bridges (as in [17]) has been done in [10]. A set of algorithms exploiting such semantic bridges are applied and the authors studied under which circumstances upper ontologies improves traditional matching approaches that do no exploit them. They developed different algorithms : one that does not look at the ontology structure; one that looks at the identity and structural information of concepts to decide when the concepts are related; and another one aggregating the structural algorithm with another that does not use upper ontologies. The experiments involved 17 ontologies and 3 top-level ontologies (SUMO, Cyc and DOLCE) used as bridges for matching domain ontologies. 10 tests cases were designed and for each, a reference alignment was manually created including only concepts.

These works use top-level ontologies as a resource for producing better domain ontologies and alignments. Some have used alignments with top-level ontologies that were manually made and others apply automatic approaches for matching ontologies of same level or for analysing the impact of using top ontologies as semantic bridges in the matching process. In fact, the best part of efforts in ontology matching research are targeted to align same domain ontologies while matching domain ontologies with top-level ontologies poses different challenges. This paper tackles the problem in a different way and analyse the behaviour of available matching tools when aligning domain with top-level ontologies. The analysis is more qualitative than quantitative, therefore it is based on a reduced data set, one domain ontology against three of the most well known top-level ontologies available. The experiments are described in the next section.

3 Experiments

3.1 Data set and matchers

OAEI Conference data set. The OAEI Conference data set[1] contains 16 ontologies covering the domain of conference organization. A subset of 21 reference alignments involving 7 ontologies (Ekaw, Conference, Sigkdd, Iasted, ConfOf, Cmt and Edas) has been published. We have chosen this data set because it provides expressive ontologies and is one of the most popular data set in the ontology matching evaluation community [2]. In the experiments presented below, we have used one ontology (the *Conference*

[1] http://oaei.ontologymatching.org

ontology[2]). This ontology has 60 concepts, 46 object properties and 18 data properties. Here, we focus on the alignment of concepts.

Top-level ontologies. The top-level ontologies DOLCE Lite, GFO Basic, and SUMO-OWL were aligned with the *Conference* ontology:

- DOLCE Lite[3]: the lite version is freely available and it is composed by 37 Concepts and 70 Object properties.
- GFO Basic[4]: the basic version is freely available and it is composed by 45 Concepts and 41 Object properties.
- SUMO[5]: the OWL version is freely available and composed by about 4.500 Concepts and 778 Object properties.

Ontology matching tools. A set of tools, publicly available, from previous OAEI campaigns (not limited to Conference track top participants), and implementing different matching strategies was selected. Even though they are not exhaustive and were not exactly developed for that purpose, their output might help us to investigate the problem of aligning domain and top-level ontologies. Aroma[6] is a hybrid tool based on association rules; Falcon-AO[7] applies linguistic and structural approaches, as Lily[8], which includes debugging strategies; LogMap[9] applies logical reasoning and repair strategies and its variant LogMap-Lite is essentially based on string similarities; MaasMatch adopts a similarity cube and a disambiguation phase as described in [15]; WeSeE-Match[10] uses web search results for improving similarity measures; WikiMatch[11] uses Wikipedia as external knowledge source and YAM++[12] applies both linguistic and graph-based approaches together with machine learning. MaasMatch and YAM++ use WordNet as background knowledge. All the tools were run with their default configuration settings.

3.2 Results and discussion

Manual evaluation. For our experiments, we ran each of the above mentioned systems for the pairs composed by the *Conference* ontology against each top-level ontology. We then merge the alignments generated by the matchers, resulting in 28 correspondences (Table 1), and submitted the resulting merge to the analysis of three evaluators. The evaluators are researchers that have common-sense knowledge about conferences (the domain ontology), with a strong background in Computer Science and well-familiarised

[2] http://oaei.ontologymatching.org/2015/conference/data/Conference.owl
[3] http://www.loa.istc.cnr.it/old/DOLCE.html
[4] http://onto.eva.mpg.de/gfo-bio/gfo-bio.owl
[5] http://www.adampease.org/OP/SUMO.owl
[6] https://exmo.inrialpes.fr/software/aroma/
[7] http://ws.nju.edu.cn/falcon-ao/
[8] http://cse.seu.edu.cn/people/pwang/lily.htm
[9] https://www.cs.ox.ac.uk/isg/tools/LogMap/
[10] http://www.ke.tu-darmstadt.de/resources/ontology-matching/wesee-match
[11] http://www.ke.tu-darmstadt.de/resources/ontology-matching/wikimatch
[12] http://www.lirmm.fr/yam-plus-plus/

with ontology matching. Each of the 28 correspondences (pairs of concepts) were presented to the evaluators, separately, via an online evaluation form (Figure 1). In this form, the first concept in the pair denotes the domain concept and the second one denotes the top concept. For the top concepts, a description (as provided by the top-level ontology) is presented in the form. Checking the ontologies could be done outside the evaluation form. Figure 1 shows one example for the pair 'Abstract' - 'Abstract (DOLCE)' presented to the evaluators. The evaluators analysed each correspondence and selected one type of relation – Equivalent, Sub/Super concept, or None – according to the relation they judged as correct.

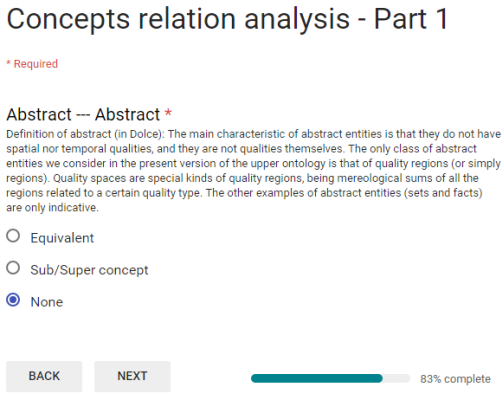


Fig. 1. Example of correspondence as shown in the online evaluation form.

A summary of the correspondences generated by the matchers together with the results of the manual annotation is presented in Table 1. In this table, the first column presents the concepts of the domain ontology for which one correspondence was found by at least one matcher. The second column shows the top-level concept that was aligned with the corresponding domain concept. The concept hierarchy is included for all concepts. The third column identifies the top-level ontology involved in the alignment. The fourth, fifth, and sixth columns are used to show the evaluators judgment about the pair of concepts. The numbers indicate how many evaluators voted for each type of correspondence. Finally, the last column summarizes how many tools aligned the corresponding pairs of concepts.

Regarding the evaluators judgement, there was total agreement among them in 20 (out of 28 correspondences). However, for 14 of them, no relation has been identified so that half of the automatically aligned concepts were considered neither equivalent nor subsumed. In 3 cases there was total agreement regarding "Subsumption", and in 3 cases total agreement for "Equivalence". From the 8 pairs resulting in a disagreement, only 2 of those corresponded to a full disagreement. These 2 cases of total disagreement were discussed among the evaluators, and in one case a total agreement for subsumption was reconsidered. For the other case, a partial agreement for 'None' (no relation) was achieved. The results in Table 1 correspond to the final agreement.

We note that, regarding the 28 correspondences, only 18 concepts of a total of 60 from the domain ontology participated in a correspondence.

Tools alignment evaluation. The evaluation of the alignments generated by the tools is based on their precision with respect to the manual analysis.We consider 4 sets of alignments:

- P_1 considers the cases of total agreement, where a correspondence is considered as correct if it has been marked either as equivalent or subsumed by the evaluators (21 correspondences regardless the type of relation – equivalence, subsumption or none – where 7 of them correspond to either equivalence or subsumption);
- P_2 considers the cases involving both total and partial agreements (28 correspondences regardless the type of relation with 14 corresponding to either equivalence or subsumption);
- P_3 considers only total agreement for equivalences (matchers have generated only equivalences) (21 correspondences with 3 equivalences);
- P_4 considers both total and partial agreements only for equivalences (28 correspondences with 4 equivalences).

Table 2 presents the precision of each tool (average of the results for the 3 pairs of ontologies). Here we have a total of 49 correspondences to be analysed, since more than one matcher may indicate a correspondence for the same pair. While some tools were able to generate alignments between *Conference* and the three top-level ontologies (LogMap, LogMapLite and YAM++), other systems have generated alignments for only one pair of ontologies (Conference-DOLCE for Aroma and Conference-GFO for Falcon-AO). Moreover, some systems were not able to generate any alignment (Lily, WeSeE and WikiMatch) and some only generate incorrect ones (Falcon-AO).

For those systems generating non empty alignments, MaasMatch and YAM++ were able to generate more correspondences than the other systems (with LogMap and its variant coming just behind). These 2 systems use WordNet in their matching approaches. This background knowledge resource is a source of lexical relations and can potentially be exploited for finding other relations than equivalence. This can explain the fact that these systems find more alignments. Their best results were obtained for P_2 (however, the best results for this set have been obtained by LogMap). Contrary to what would be expected, these systems (and all others, in fact) were not able to generate subsumption (even though some have been designed to). They generated only equivalences, even when they were in fact subsumptions.

Looking to the different sets, in P_1, LogMap, LogMapLite and MaasMatch outperformed YAM++. In P_2, LogMap achieves the best results followed by MaasMatch. When only equivalence (P_3) is considered, the numbers drop for some matchers. When relaxing to both partial and total agreements the results drop even more (P_4). Some matchers are doing equivalence consistently (LogMapLite), whereas others are also indicating correspondences which were in fact considered subsumption by the judges (LogMap, MaasMatch, YAM++), so that P_3 and P_4 decrease. Moreover, precision is low if we compare the results when the same systems are matching domain ontologies[13].

[13] http://oaei.ontologymatching.org/2015/conference/eval.html

Table 1. Union of the correspondences found by the tools.

			Manual			
Conference Ontology	**Top-Level Ontology**	**Ontologies**	$\equiv$	$\sqsupseteq$	**None**	**Tools**
Conference_document/Conference_ contribution/Written_contribution/Regular_ contribution/Extended_abstract/Abstract	particular/abstract	DOLCE Lite			3	5
	Entity/abstract	SUMO			3	3
	Individual/Abstract	GFO Basic			3	5
Person/Committee_member/Chair	Entity/object/artifact/ furniture/seat/chair	SUMO			3	3
Person/Conference_applicant	particular/spatio-temporal-particular/endurant/non-physical-endurant/ non-physical-object	DOLCE Lite			3	1
Conference_document	particular	DOLCE Lite			3	1
Conference_part	particular/spatio-temporal-particular/endurant/physical-endurant/feature/relevant-part	DOLCE Lite		1	2	1
	particular/abstract/region	DOLCE Lite			3	1
Conference_proceedings	particular/spatio-temporal-particular/perdurant/stative/ process	DOLCE Lite		1	2	1
	Individual/Concrete/ Processual Structure/ Process	GFO Basic		1	2	1
Conference/Conference_volume	particular	DOLCE Lite		2	1	1
Conference_document/Conference_ contribution/Written_contribution/ Regular_contribution/Extended_abstract	particular/abstract/region/ abstract-region	DOLCE Lite			3	1
Organization	Entity/physical/object/agent/ group/organization	SUMO	3			2
Organizer	Entity/physical/object/agent/ group/organization	SUMO			3	1
	Entity/physical/object/agent/ organism	SUMO		3		1
Conference_document/Conference_ contribution/Written_contribution/ Regular_contribution/Paper	Entity/physical/object/artifact/ paper	SUMO			3	3
Person	Individual	GFO Basic		1	2	1
Conference_document/Conference_ contribution/Poster	Entity/physical/content bearing physical/VisualContentBearing Object/PrintedSheet/Poster	SUMO	3			3
	Individual/Property	GFO Basic			3	1
Conference_document/Conference_ contribution/Presentation	particular/abstract/proposition	DOLCE Lite		2	1	1
	Individual/Concrete/Processual Structure/Occurrent/Event	GFO Basic		3		1
Publisher	Entity/physical/agent/ commercial-agent/publisher	SUMO	3			3
Person/Conference_applicant/ Registeered_applicant	particular/spatio-temporal-particular/endurant/physical-endurant/physical-object	DOLCE Lite		3		1
	Entity	GFO Basic		3		1
Topic	particular/abstract/region/ temporal-region/time-interval	DOLCE Lite			3	1
	Category/Concept	GFO Basic	1		2	1
Conference_part/Workshop	particular/spatio-temporal-particular/quality/physical-quality/spatial-location_q	DOLCE Lite			3	1
	Entity/physical/object/region/ geographic- area/LocalizablePlace/ stationary artifact/workshop	SUMO			3	3
Total of correspondences found by the tools:						49

Table 2. Precision of each system considering their complete set of alignments.

System	P_1	P_2	P_3	P_4
Aroma	0/2 0	1/3 .33	0/2 0	0/3 0
Falcon-AO	0/1 0	0/1 0	0/1 0	0/1 0
Lily	-	-	-	-
LopMap	3/9 .33	5/11 .55	3/9 .33	3/11 .27
LogMapLite	3/9 .33	3/9 .33	3/9 .33	3/9 .33
MaasMatch	3/10 .30	5/12 .42	0/10 0	1/12 .08
WeSeE-Match	-	-	-	-
WikiMatch	-	-	-	-
YAM++	3/11 .27	5/13 .38	2/11 .18	2/13 .15
Total	12/42 .29	19/49 .39	8/42 .19	9/49 .18

Table 3 shows the overall precision of aligned concepts for each pair of ontologies (based on the union of generated alignments). As expected, the best precision is achieved for P_2 (for the pairs involving GFO). However, if we consider only equivalences (P_3 and P_4), the best precision was achieved with SUMO. We also observe that more correspondences have been generated involving DOLCE concepts (12 pairs), but it corresponds to the lower precision across the different sets.

Table 3. Precision of the alignment union (considering all systems).

Pair of Ontologies	P_1	P_2	P_3	P_4
Conference - DOLCE Lite	1/8 .13	5/12 .42	0/8 0	0/12 0
Conference - GFO Basic	2/4 .50	5/7 .71	0/4 0	1/7 .14
Conference - SUMO	4/9 .44	4/9 .44	3/9 .33	3/9 .33
Total	7/21 .33	14/28 .50	3/21 .14	4/28 .14

Another simpler way to look at the quality of the alignments generated by the tools is presented in Table 4. It summarizes the correspondences considered correct by at least 1, 2 or by all 3 evaluators. The table also indicates the number of times a relation of equivalence found by the matchers were considered equivalence or subsumption by the evaluators. It shows that 14 out 28 correspondences made by the tools were considered as equivalent or subsumed by at least 1 evaluator. Total agreement happened in 7 of these cases (after discussion on the cases of total disagreement).

Table 4. Number of correct correspondences according to the evaluators analysis.

	at least 1 judge	**at least two judges**	**three judges**
$\sqsupseteq$	10	6	4
$\equiv$	4	3	3
$\sqsupseteq + \equiv$	14	9	7

Discussion. Regarding the qualitative analysis of the alignments, we observe that the systems found various correspondences between concepts with the same term (“Abstract”, “Chair”, “Paper”, “Workshop”, “Organization”, “Poster” and “Publisher”). This is quite expected as all tools are based on some string-based matching strategy. However, many of them were considered as having no correspondence by the evaluators

(“Abstract”, “Chair”, “Paper”, “Workshop”). Among these concepts, the most common aligned one was ‘Abstract’ involving DOLCE and GFO (5 tools) and SUMO (3 tools). Some other concepts were aligned by three or two different tools, but most concepts were aligned just by one. The other correspondences provided by the tools which were considered no correspondent by all the evaluators can be found in Table 1.

There were correspondences with the same term which were considered equivalent by the evaluators :

- “Organization” in the top-level ontology is defined as: a group of people with a common purpose or function in a corporate or similar institution, the same as in the conference domain.
- “Poster” is defined as: a printed sheet intended to be posted on a horizontal surface, so as to make the information it displays visible to passers by.
- “Publisher” in the top-level ontology refers to: some service that includes the publication of texts, so as in the conference domain.

The 3 cases above were SUMO concepts. The concept “Organization” was aligned by 2 systems and the others by 3. For some concepts, all evaluators considered that there was a correspondence but selected subsumption instead of equivalence :

- Organizer and organism: The first concept refers to people who organizes conferences and the second refers to a living individual, then, the concept “Organizer” was considered as subsumed by “Organism”.
- Presentation and event: The first concept refers to the action of explaining about some topic for a group of people. The second refers to processual structures comprising a process. “Presentation” was considered as subsumed by “Event” by the judges.
- Registered_applicant and physical-object: The first concept refers to people who apply and is able to participate in the conference. The main characteristic of the second concept is that they are endurants with unity and most physical objects change some of their parts while keeping their identity, they can have therefore temporary parts. In this case, one “Registered_applicant” is a person who in some specific time interval assumes this role, but keeping their identity as person, then, it was considered as subsumed by “Physical-object”.
- Registered_applicant and entity: The first concept was interpreted in the same way as above. The second concept refers to everything that exists in the broadest sense. In this case, one “Registered_applicant” is something that exists, than, the first concept was considered as subsumed by “Entity”.

An important aspect is that finding subsumption correspondences is in fact highly desirable when matching domain and top-level ontologies. Ideally, such a matcher should try to find the closest super concept. However the matchers we tested in this experiment were not able to generate subsumption, even if some of them (Aroma, for instance) are supposed to do so. They generated only equivalences, even when they were in fact subsumptions. This is however an important distinction. Finally, our analysis does not take into account the inconsistencies introduced in the merging alignments from all tools. In fact, it is contradictory that *Conference_applicant* aligns to *Non_physical_object* and *Registered_applicant* to *Physical_object*, considered that the

latter is a subclass of *Conference_applicant* in the domain ontology. This could be exploited for further filtering out correspondences. We could as well enrich the set of manually validated correspondences by introducing simple hierarchical reasoning.

To sum up, although the number of evaluators is relatively small, it allowed us to establish a first evaluation of available tools on the task. Our study was useful to observe various questions in the task of matching ontologies of different levels of abstraction :

- there was a small quantity of aligned concepts by the tools in general (in total, 18 of 60 concepts), even considering all concepts provided by the top ontologies;
- there were many produced correspondences which were not considered as correspondences by the specialists, many string matching cases which are usually safe in same domain correspondences did not apply, according to our study;
- there is a lack of comprehensive evaluation data sets (regarding domain vs. top-level ontologies) to evaluate the systems, and to overcome that we presented an analysis of the output generated by current systems;
- knowledge on top level ontologies is highly specialized, it is important that such evaluation considers an overview of experts in this area;
- both domain and top ontologies may lack further context or documentation that is appropriate to help identifying the right correspondences;
- manual analysis or correspondences generation by specialists is a hard and expensive work, in this work we ran experiments on a small set of concepts and this problem has been reduced; bigger data sets would require more efforts;
- matching strategies for dealing with this task should take advantage of structural features of the ontologies, background knowledge from external resources targeting subsumption correspondences, and logical reasoning techniques for guarantee the consistency of the generated alignments;
- at last, but not least, current tools do not distinguish between subsumption and equivalence correspondences, which in this kind of task is a crucial point, finding the closest super-concept is quite desirable when aligning to a top-level ontology.

4 Concluding remarks and future work

This paper presented an analysis of the alignments between three top-level ontologies with one domain ontology as produced by a set of matching tools. Our goal was to analyse the behaviour of these tools, which apply diverse matching techniques, with respect to this task. We could observe that matching top-level and domain ontologies automatically is an interesting and challenging task. Top-level ontologies focus on the standardisation of more general concepts to be easily reused in a large amount of domains. On the other hand, there are a lot of domain ontologies available in different fields. Therefore, we claim that it is important to reuse the well-founded knowledge available in the top-level ontologies together with the domain ontologies to reduce the time of ontology modeling, the heterogeneity problem of the knowledge representation, and the complexity of ontology modeling. Hence the automatic matching process should be an alternative. Furthermore, top-level ontologies are semantic bridges for helping solving the heterogeneity between domain ontologies that have to be integrated.

As future work, we plan to run experiments exploiting the whole space of possible alignments (regarding a data set) and to extend the evaluation taking into account

matching tools participating in more recent OAEI campaigns. We plan as well to involve evaluators experimented in top-level ontologies and with different backgrounds (Computer Scientists, Philosophers) in the manual evaluation process. We intend also to exploit background knowledge from external resources (like BabelNet) in order to improve the results reported here, paying special attention to subsumption relations. Combining it with logical reasoning is another aim. Finally, we intend to exploit other data sets such as the ones available on the BioPortal, which contain manually validated alignments between biomedical ontologies and the top level ontologies GFO and BFO.

References

1. Brodaric, B., Probst, F.: DOLCE ROCKS: Integrating Geoscience Ontologies with DOLCE. In: Semantic Scientific Knowledge Integration. pp. 3–8 (2008)
2. Cheatham, M., Hitzler, P.: Conference v2.0: An Uncertain Version of the OAEI Conference Benchmark. In: Proc. of the 13th Intern. Semantic Web Conference. pp. 33–48 (2014)
3. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A.: Sweetening WORDNET with DOLCE. AI Magazine 24(3), 13–24 (2003)
4. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., Schneider, L.: Sweetening Ontologies with DOLCE. In: Proc. of the 13th Intern. Conf. on Knowledge Engineering and Knowledge Management. pp. 166–181 (2002)
5. Guarino, N.: Formal Ontology in Information Systems: Proceedings of the 1st Intern. Conference. IOS Press, 1st edn. (1998)
6. Herre, H., Heller, B., Burek, P., Hoehndorf, R., Loebe, F., Michalek, H.: General Formal Ontology (GFO): A Foundational Ontology Integrating Objects and Processes. In: Basic Principles, Research Group Ontologies in Medicine) (2007)
7. Khan, Z., Keet, C.M.: The Foundational Ontology Library ROMULUS. In: Proc. of the 3rd Conf. on Model and Data Engineering. pp. 200–211 (2013)
8. Khan, Z., Keet, C.M.: Toward semantic interoperability with aligned foundational ontologies in ROMULUS. In: Proc. of the 7th Conf. on Knowledge Capture. pp. 23–26 (2013)
9. Mascardi, V., Cordì, V., Rosso, P.: A Comparison of Upper Ontologies. In: Proc. of the 8th AI*IA/TABOO Joint Workshop on Agents and Industry. pp. 55–64 (2007)
10. Mascardi, V., Locoro, A., Rosso, P.: Automatic Ontology Matching via Upper Ontologies: A Systematic Evaluation. IEEE Trans. on Knowl. and Data Eng. 22(5), 609–623 (2010)
11. Mika, P., Oberle, D., Gangemi, A., Sabou, M.: Foundations for Service Ontologies: Aligning OWL-S to Dolce. In: Proc. of the 13th Conf. on World Wide Web. pp. 563–572 (2004)
12. Mika, P., Oberle, D., Gangemi, A., Sabou, M.: Foundations for Service Ontologies: Aligning OWL-S to DOLCE. In: Proc. of the 13th Conf. on World Wide Web. pp. 563–572 (2004)
13. Niles, I., Pease, A.: Towards a Standard Upper Ontology. In: Proc. of the Intern. Conf. on Formal Ontology in Information Systems. pp. 2–9 (2001)
14. Paulheim, H., Gangemi, A.: Serving DBpedia with DOLCE - More than Just Adding a Cherry on Top. In: The Semantic Web, pp. 180–196 (2015)
15. Schadd, F.C., Roos, N.: Coupling of Wordnet Entries for Ontology Mapping Using Virtual Documents. In: Proc. of the 7th Conf. on Ontology Matching. pp. 25–36 (2012)
16. Semy, S., Pulvermacher, M., Obrst, L.: Toward the use of an upper ontology for U.S. government and U.S. military domains: An evaluation. Tech. rep., Sub. to Workshop on Information Integration on the Web, in conjunction with VLDB-2004 (2004)
17. Silva, V., Campos, M., Silva, J., Cavalcanti, M.: An Approach for the Alignment of Biomedical Ontologies based on Foundational Ontologies. Information and Data Management 2(3), 557–572 (2011)

Ontology Alignment Evaluation in the Context of Multi-Agent Interactions

Paula Chocron and Marco Schorlemmer

Artificial Intelligence Research Institute, IIIA-CSIC
Bellaterra (Barcelona), Catalonia, Spain **

Abstract The most prominent way to assess the quality of an ontology alignment is to compute its precision and recall with respect to another alignment taken as reference. These measures determine, respectively, the proportion of found mappings that belong to the reference alignment and the proportion of the reference alignment that was found. The use of these values has been criticised arguing that they fail to reflect important semantic aspects. In addition, they rely on the existence of a reference alignment. In this work we discuss the evaluation of alignments when they are used to facilitate communication between heterogeneous agents. We introduce the notion of pragmatic alignment to refer to the mappings that let agents understand each other, and we propose new versions of precision and recall that measure how useful mappings are for a particular interaction. We then discuss practical applications of these new measures and how they can be estimated dynamically by interacting agents.

1 Introduction

Communication between heterogeneous agents has been identified as one important application for ontology alignments [9]. In dynamic and open environments such as multi-agent systems, agents with multiple backgrounds may not share their vocabularies or representations of meaning. Even when a common vocabulary is established, maintaining it over time can be a difficult task, particularly in dynamic domains [4]. To achieve meaningful communication it is therefore necessary to develop techniques that align the vocabularies that agents use, obtaining a translation that allows them to interpret the messages they receive correctly. If agents organise their vocabularies in some kind of taxonomy or ontology, a very reasonable approach is to take advantage of the diverse ontology alignment tools that were developed in the last decades [9]. However, language used in agent communication has its own particularities that should be taken into account when using alignments for this purpose; mainly, language is contextualised in the concrete interaction agents are performing. General purpose

** pchocron,marco@iiia.csic.es. This research has been funded by the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 607062 /ESSENCE: Evolution of Shared Semantics in Computational Environments/.

ontology matchers do not take this into account, and despite being an important application, there is little research on the creation and use of alignments for agent interaction.

In this paper we focus on the problem of using ontology alignments as translators for agent communication, and particularly on their evaluation for that application. We are interested in developing measures to decide whether an alignment is useful for a particular interaction, that is, if using it will help agents communicate. Traditionally, ontology alignments are evaluated with respect to a human-crafted reference alignment, and accuracy measures count the elements in the intersection between the evaluated alignment and the reference. In this way, the *precision* of an alignment is defined as the proportion of found mappings that belong to the reference alignment, while the *recall* is the proportion of the reference alignment that was found. We propose an application-dependent evaluation technique that does not require the (possibly arbitrary) construction of a gold standard. In this way, we make a step towards considering the problem of "in situ evaluation", based on the idea that *"the relative quality or usefulness of a generated alignment also depends on its intended use"* [8].

Our approach considers agents taking part in task-oriented interactions, and defines a mapping as correct if it allows agents to finish the joint task successfully. This leads to the notion of *useful* and *misleading* mappings, which are, respectively, those that lead to the success or failure of an interaction. This new classification allows us to redefine the traditional precision and recall measures that are used for alignment evaluation, comparing an alignment against the specification of an interaction, thus providing a method for evaluating alignments that does not rely on a human-built alignment. We then show how these newly defined measures can be used by agents to improve the quality of their understanding, and sketch a method in which agents can estimate them online using their experience from interaction, making evaluation automatic.

2 Related Work

The use of the standard precision and recall notions from information retrieval for the evaluation of ontology alignments has been criticised by different authors, all of whom argue that these measures ignore important aspects of the problem that should also be taken into account to decide how good a solution is. The main approach to creating measures that are more appropriated for the nature of semantic mappings is the one of *semantic* precision and recall [6]. Here, Euzenat tackles the problem of the binary nature of traditional precision and recall (if a mapping is not found by the alignment, it is missing), by considering the relation between the consequences of the alignments instead of between the alignments themselves. In [11], Holling et al. propose new evaluation measures that take into account the frequency of use of the mappings found, as well as the semantic distance to an alignment. In [13], the authors introduce the notion of *relevance* of a mapping, that measures how often the mapped words appear in a particular context.

Also relevant are approaches that consider the use and evolution of alignments in a multi-agent environment. Both [10] and [2] propose methods to create alignments from scratch that are learned from the agent's interaction experience. In [7] and in [5] the authors propose techniques to repair alignments with information that is learned directly from observations made while interacting. A similar idea is proposed in [12], but in this case agents repair their ontologies instead of alignments.

3 A Pragmatic Approach to Alignment

We consider the problem of achieving meaningful communication between two agents a_1 and a_2 that need to interact to perform some task, but use potentially different vocabularies V_1 and V_2 respectively. Each agent can organise its vocabulary in its own way, using structures that go from simple lists of words to fully fledged ontologies. We only suppose that they can be matched with one of the existing tools to obtain an alignment between them.

Definition 1. *An* alignment $\mathcal{A}$ *between two vocabularies* V_1 *and* V_2 *is a finite set of* mappings *between words in* V_1 *and* V_2*. A mapping is defined as a quadruple* $\langle v_1, v_2, n, r\rangle$*, where* $v_1 \in V_1$*,* $v_2 \in V_2$*,* $n \in (0, 1]$ *is the degree of confidence on the mapping, and* r *is the kind of relation that holds between words. An alignment must contain at most one tuple for each pair* $v_1 \in V_1$*,* $v_2 \in V_2$*. [3]*

When working with an alignment $\mathcal{A}$, if a mapping $\langle v_1, v_2, n, r\rangle$ belongs to $\mathcal{A}$ we will write $v_1\, r\, v_2$ (for example, $v_1 \equiv v_2$).

In general, techniques to build alignments between different vocabularies make use of the structure or additional information in the ontologies in which such vocabularies are organised. Other techniques use external resources, such as text corpora or the web. Still others have a completely syntactic approach. Extending the ideas in [2], we propose a different kind of alignments, that we call *pragmatic*. This kind of alignments are produced by only taking into account the interactions in which agents use their vocabularies. Let us first define the specifications of interactions, and then move to formalise the alignments.

3.1 Interaction Specifications

We specify interactions performed jointly by agents by means of *interaction protocols* that define all possible sequences of message exchanges. The multi-agent systems community has extensively discussed possible formalisms to describe these kind of protocols; in this work we stick to a generic approach that uses Finite State Automata. Since we focus on agents that communicate to perform a task together (for example, *ordering drinks*), the interaction can end successfully (if the task is completed) or can fail (if it is not). To decide this outcome, we introduce the notion of *state properties*, which are Boolean predicates assigned to final states to represent observations. Interactions are successful only if agents reach together final states with the same properties.

Definition 2. *Given two agents a_1 and a_2, a vocabulary V, and a set of state properties SP, an* interaction model *IM is defined as a tuple $\langle Q, q_0, \delta, F, \rho, speaks\rangle$ where Q is a finite set of states, $q_0 \in Q$ is the initial state, $F \subseteq Q$ is the set of final states, $\rho : F \to \mathcal{P}(SP)$ assigns a subset of state properties to each final state, $speaks : Q \to \{a_1, a_2\}$ assigns to each state its sender agent, and $\delta : Q \times V \to Q$ is a partial function called the* transition function*.*

Note that while we do not specify any particular turn-taking pattern, we do require that, for each state, all messages labelling transitions from this state share the same sender agent, who is determined with the *speaks* function. For simplicity reasons, we will consider that δ is undefined for the final states F.

In the rest of this paper, including all the definitions, we consider interactions between two agents a_1 and a_2 with interaction models $IM_i = \langle Q_i, q_i^0, F_i, \delta_i, \rho_i, speaks_i\rangle$, $i = 1, 2$. While IM_1 and IM_2 have the same set of agents ($\{a_1, a_2\}$), their vocabularies and state properties can differ; we will call them V_1, V_2 and SP_1, SP_2 respectively.

3.2 Pragmatic Alignments

Alignments between the vocabularies of two interaction models, that we will call *pragmatic alignments*, capture relations between the ways in which words are used in a conversation. In this way, a word v_1 from IM_1 matches with a word v_2 from IM_2 if an agent can interpret v_1 as v_2 in an interaction and finish the task successfully.

Definition 3. *Consider IM_1 and IM_2 such that $speaks(q_1^0) = speaks(q_2^0)$. Extending [1], the* communication product *of IM_1 and IM_2 ($IM_1 \otimes IM_2$) is an interaction model $\langle Q, q^0, F, \delta, \rho, speaks\rangle$ over a language V that is the Cartesian product between V_1 and V_2, a set of agents $\{a_1, a_2\}$, and $SP = \{success, failure\}$, and such that:*

- *Q is a subset of the Cartesian product of Q_1 and Q_2 in which both states have the same senders, in other words, the states in Q are all possible ordered pairs $\langle q_1, q_2\rangle$ with $q_1 \in Q_1$, $q_2 \in Q_2$, and $speaks_1(q_1) = speaks_2(q_2)$*
- *$speaks$ is the speaker in q_1 or q_2: $speaks(\langle q_1, q_2\rangle) = speaks_1(q_1)(= speaks_2(q_2))$*
- *the initial state q^0 is the pair $\langle q_1^0, q_2^0\rangle$*
- *δ is defined as follows: $\langle q_1', q_2'\rangle = \delta(\langle q_1, q_2\rangle, \langle v_1, v_2\rangle)$ if $\delta_i(q_i, v_i) = q_i'$ for $i \in \{1, 2\}$*
- *F are all states in Q for which δ is not defined*
- *For $\langle q_1, q_2\rangle \in F$, $\rho(\langle q_1, q_2\rangle) = \{success\}$ if $q_1 \in F_1, q_2 \in F_2$, and $\rho_1(q_1) = \rho_2(q_2)$. It is $\{failure\}$ otherwise.*

With this construction, we can easily obtain all possible interactions between agents with two interaction models.

Definition 4. *An* interaction *between two interaction models IM_1, IM_2 is an accepted string in the communication product IM between IM_1 and IM_2. An interaction is* successful *if it ends in a state q such that $\rho(q) = \{success\}$, it is* unsuccessful *if $\rho(q) = \{failure\}$.*

These interactions can be seen as all possible combinations of uttered messages and their interpretations; our objective is to use them to define pragmatic alignments. An immediate approach consists in considering two words as equivalent if they belong to a successful interaction. In an alignment of this kind, one word in V_1 could be mapped to many words in V_2 if they have different interpretations in different states. Instead, agents will be interested in knowing which mapping is correct for each state. This information can be obtained from successful interactions if we consider deterministic FSAs in which any accepted string can be assigned to an unique sequence of states. In the following definition, mappings are parametrised by states in the communication product.

Definition 5. *A* pragmatic alignment *between interaction models* IM_1, IM_2 *is a set of tuples* $\langle q, v_1, v_2, r\rangle$*, where* $q \in Q, v_1 \in V_1, v_2 \in V_2$*, and* $r \in \{\equiv, \between\}$*.*

The relation between two words ($\equiv$ or $\between$) depends on whether finishing the interaction successfully is always possible after mapping them. To define formally their semantics, we will refer to each state in one of these accepted strings as $\langle q, v\rangle$, representing the state and the message.

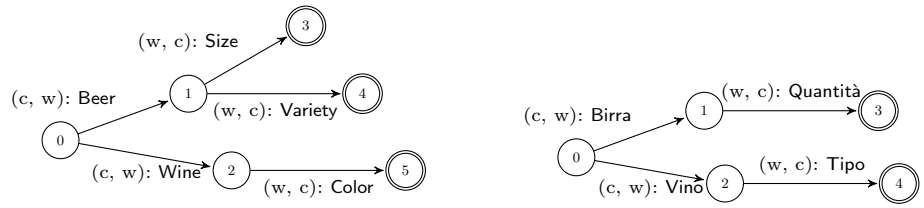
- $IM_1, IM_2 \models \langle\langle q_1, q_2\rangle, v_1, v_2, \equiv\rangle$ if there are interactions between IM_1 and IM_2 that include $\langle\langle q_1, q_2\rangle, \langle v_1, v_2\rangle\rangle$, and all strings accepted by IM_1 or IM_2 that include $\langle q_1, v_1\rangle$ or $\langle q_2, v_2\rangle$ are the projection of one of these interactions (the interaction can always end successfully after mapping v_1 with v_2).
- $IM_1, IM_2 \models \langle\langle q_1, q_2\rangle, v_1, v_2, \between\rangle$ if there exists at least one successful interaction between IM_1 and IM_2 that includes $\langle\langle q_1, q_2\rangle, \langle v_1, v_2\rangle\rangle$ (the interaction can end successfully at least for some cases after mapping v_1 with v_2).

As an example, consider the interaction models in Figure 1, which represent fragments of interactions between a waiter (w) and a customer (c) to order drinks in English and Italian (state transitions should be read as $(sender, receiver) : message$.). Let IM_1 have $SP : \{size\_beer, kind\_beer, kind\_wine\}$, and IM_2 have $SP : \{kind\_beer, kind\_wine\}$, and $\rho_1(3) = size\_beer$,$\rho_1(4) = \rho_2(3) = kind\_beer$, $\rho_1(5) = \rho_2(4) = kind\_wine$. The mapping $\mathsf{Wine} \equiv \mathsf{Vino}$ in $\langle 0, 0\rangle$ is satisfied by IM_1, IM_2, because the interaction $(a_1 : \langle \mathsf{Wine}, \mathsf{Vino}\rangle, a_2 : \langle \mathsf{Color}, \mathsf{Tipo}\rangle)$ is successful in the communication product, and all accepted strings in IM_1 and IM_2 that include $mathsfWine$ and $mathsfVino$ respectively are projections of it. The mapping $\mathsf{Beer} \equiv \mathsf{Birra}$ in $\langle 0, 0\rangle$ is not, because there is no interaction that projects $(\mathsf{Beer}, \mathsf{Size})$. However, $\mathsf{Beer} \between \mathsf{Birra}$ is satisfied, because $(\langle \mathsf{Beer}, \mathsf{Birra}\rangle, \langle \mathsf{Variety}, \mathsf{Tipo}\rangle)$ is successful.

Pragmatic alignments are everything agents need to communicate successfully, but they are only useful in a particular context. Notice, for example, that mapping Tipo with Color is not correct in a general English-Italian translation; however in the context of ordering drinks it yields to common understanding.

4 Pragmatic Evaluation of Alignments

The quality of a vocabulary alignment is typically measured in comparison with a *reference alignment*, for which values of *precision* and *recall* are computed. As



(a) $IM_1 = \langle Q_1, q_1^0, \delta_1, F_1, \rho_1 \rangle$ (b) $IM_2 = \langle Q_2, q_2^0, \delta_2, F_2, \rho_2 \rangle$

Figure 1: Fragments of interaction models for ordering drinks

it is commonly done, we do not take into account the confidence degrees in these measures.

Definition 6. *Given an alignment $\mathcal{A}$, let $\mathcal{A}'$ denote the set of mappings of $\mathcal{A}$ for which we have removed the confidence degree, i.e., $\mathcal{A}' = \{\langle v_1, v_2, r \rangle \mid \langle v_1, v_2, n, r \rangle \in \mathcal{A}$ for some $n\}$. The* precision *of an alignment $\mathcal{A}$ with respect to a reference alignment $\mathcal{B}$ is the fraction of the mappings in $\mathcal{A}'$ that are also in $\mathcal{B}'$:*

$$precision(\mathcal{A}, \mathcal{B}) = \frac{\mid \mathcal{A}' \cap \mathcal{B}' \mid}{\mid \mathcal{A}' \mid}$$

while its recall *is the fraction of the mappings in $\mathcal{B}$ that were found by $\mathcal{A}$:*

$$recall(\mathcal{A}, \mathcal{B}) = \frac{\mid \mathcal{A}' \cap \mathcal{B}' \mid}{\mid \mathcal{B}' \mid}$$

Two problems arise when using these measures to assess the quality of an alignment $\mathcal{A}$ used for agent interaction. First, a reference alignment between the vocabularies may not be available. Second, even if it is, the measures do not take into account the way in which terms are used in an interaction. To show this, we performed a small experiment, based on the ones in [5], and let agents with heterogeneous vocabularies interact using alignments of different qualities. In Figure 2, we can see that recall is more relevant than precision; this is because the alignment counts as correct many mappings that are not actually necessary for interacting.

In this section we propose adaptations of the traditional precision and recall measures that evaluate an alignment taking as reference, not a human-crafted standard, but a pragmatic alignment obtained from two interaction models. We introduce the notions of *useful* and *misleading* mappings for those that belong to successful and unsuccessful interactions respectively. In this first approach we will only consider alignments with $\equiv$ relations, the problem of analysing other relations is left for future work.

Definition 7. *Consider an alignment $\mathcal{A}$ between vocabularies V_1 and V_2 and the already defined interaction models IM_1 and IM_2. A mapping $\langle v_1, v_2, n, \equiv \rangle \in \mathcal{A}$ is* useful *with respect to IM_1, IM_2 if $\langle v_1, v_2 \rangle$ appears in a successful*

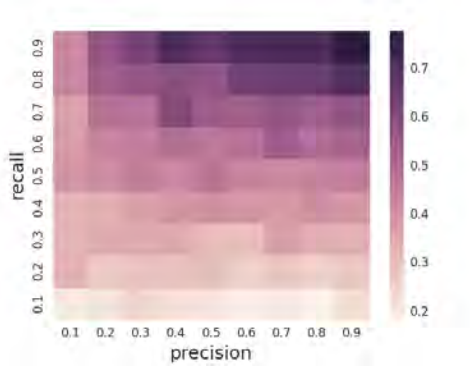


Figure 2: Success rates for different values of precision and recall

interaction between IM_1 and IM_2. It is misleading *if the same pair appears in an unsuccessful interaction between IM_1 and IM_2.*

Notice that there can be mappings in $\mathcal{A}$ that are neither useful or misleading. We will call *relevant* to the mappings that can be classified in useful or misleading, or equivalently, those between pairs that belong to an interaction between the models. More surprisingly, a mapping can be both useful and misleading at the same time, if the relation in the pragmatic alignment is $\wr$. This allows for different possibilities when computing precision and recall. In this paper we consider as correct all useful alignments.

To define precision and recall for $\mathcal{A}$ with respect to IM_1, IM_2, let $useful$ and $relevant$ be, respectively, the sets of useful and relevant mappings of $\mathcal{A}$ with respect to the interaction models. Let $\mathcal{A}_p$ be the pragmatic alignment between IM_1 and IM_2, and let us define $pragmatic = \{\langle v_1, v_2, r\rangle$ if $\langle q, v_1, v_2, r\rangle \in \mathcal{A}_p$ for some $q \in Q\}$. Pragmatic precision and recall are defined as follows:

$$recall = \frac{\mid useful \mid}{\mid pragmatic \mid}$$

$$precision = \frac{\mid useful \mid}{\mid relevant \mid}$$

As argued in [11], we may want to take into account not only how many, but also which of the mappings are found by the alignment. Finding a correct mapping for a very common word should have more impact in the precision than finding a mapping for a rarely used one. This can be taken into account in the pragmatic precision and recall measures we just defined, by simply considering *useful* and *relevant* as multi-sets:

- *useful:* for each state $q \in Q$, all mappings in $\mathcal{A}_p$ that are useful in q
- *relevant:* for each state $q \in Q$, all mappings in $\mathcal{A}_p$ that are relevant in q

Precision is defined in the same way, and recall as:

$$recall = \frac{\mid useful \mid}{\mid \mathcal{A}_p \mid}$$

It is worth noting that, with these definitions, possible values for pragmatic precision and recall are determined by the structure of interaction models. For example, consider a linear interaction model in which each state has only one outgoing arrow. There are no possible misleading matches with this protocol; therefore the minimum level of precision for alignments is necessarily 1.

4.1 An Example: ordering drinks

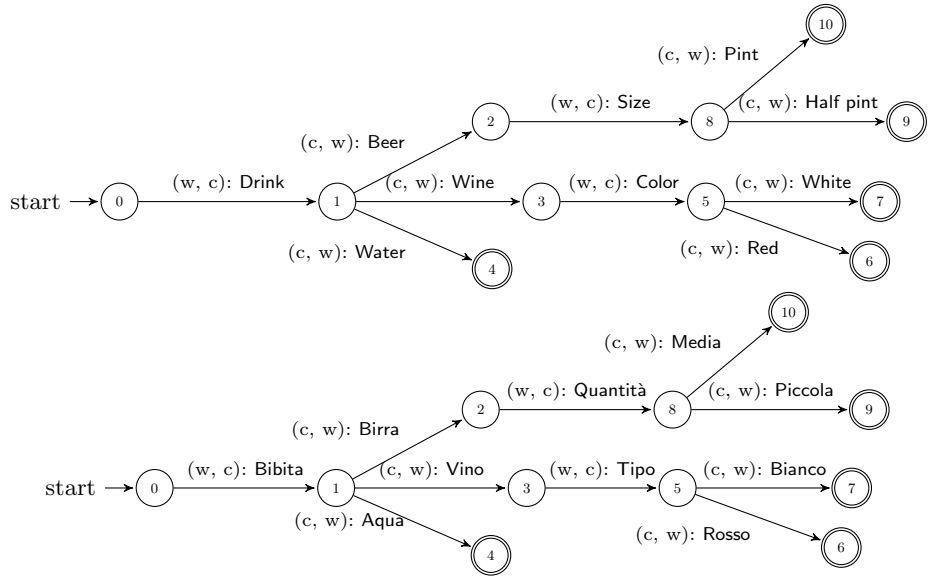


Figure 3: English and Italian interaction models for ordering drinks

Consider the alignments presented in Table 1 applied to the ordering drinks scenario represented by the protocols in Figure 3. According to an English-Italian dictionary, they would both have precision 0.5 (Wine ≡ Vino and Red ≡ Rosso are correct). Depending on the way of using the dictionary, Media ≡ Half Pint could also be considered correct, giving the second alignment a precision of 0.75. However, they are clearly not equally useful when used by agents interacting, because the second alignment has a misleading mapping Media ≡ Half Pint. Using our values, both alignments have a recall of 0.2 (Wine ≡ Vino, Red ≡ Rosso are the useful alignments found), but the first one has a precision of 1 and the second one of 0.66.

Alignment 1		Alignment 2	
$v_1 \in V_1$	$v_2 \in V_2$	$v_1 \in V_1$	$v_2 \in V_2$
Bibita	Water	Bibita	Water
Vino	Wine	Vino	Wine
Rosso	Red	Rosso	Red
Quantità	Pint	Media	Half Pint

Table1: Two alignments for the Ordering Drinks example

5 Pragmatic Precision and Recall in Practice

In their pragmatic version, precision and recall are not only indicators of how useful an alignment is for a particular interaction, but can also be actively used by semantically heterogeneous agents to improve their mutual understanding. Methods to learn pragmatic alignments and to transform traditional alignments into pragmatic ones can be obtained by adapting the techniques developed in [2] and [5] respectively. In this section we focus on the practical application of the evaluation of pragmatic alignments. We first analyse how pragmatic precision can be used to improve automatic matching techniques, and then sketch a method in which agents can estimate them from the experience of interaction.

5.1 Using Pragmatic Precision and Recall

Consider an agent that interacts with another one using an alignment that it does not trust completely. If the agent translates the messages it receives by always following the alignment, it would very frequently fail to communicate when the alignment has any misleading mapping. To avoid this situation, the following heuristic can be used to decide when to follow the alignment and when to explore.

Matching Criterion.
Consider an agent a_1 with interaction model IM_1 and an alignment $\mathcal{A}$. When receiving v_2 in state $q_1 \in Q_1$, a_1 needs to decide how to interpret it, or which outgoing arrow from q_1 to follow. Let $U(q_1)$ be the set of all these possible interpretations. For each $v_1 \in U(q_1)$, a_1 computes the value of the mapping as:

$$\mathcal{V}(v_1, v_2) = \begin{cases} n & \text{if } \langle v_1, v_2, n, \equiv \rangle \in \mathcal{A} \\ 0 & \text{otherwise} \end{cases}$$

let $\hat{\mathcal{V}}(v_1, v_2)$ be the normalized values for $v_1 \in U(q_1)$, and consider an exploration parameter $\alpha \in [0, 1]$. The criterion consists in choosing $v_1 \in U(q)$ with probability:

$$p(v_1) = \alpha\, \hat{\mathcal{V}}(v_1, v_2) + (1 - \alpha) \frac{1}{\mid U(q) \mid}$$

A reasonable question is how to choose a good value of α. It is easy to see that the values that give better results in terms of rate of successful interactions depend on the pragmatic precision of $\mathcal{A}$ with respect to IM_1 and the protocol IM_2 of the agent a_1 interacts with. If precision is high, agents should trust more on the alignment, if it is low they should rely more on the random exploration.

To show this, we performed a short experiment, in which we analyse the rate of success of interactions between agents that use different values of α and have alignments of different qualities. We used the customer and waiter agents from the example in Section 4.1 and let them interact for 150 times, measuring in how many cases they succeeded. As a simplification, we used only alignments that had the same values of precision and recall; this should be extended in future work to consider more realistic values. We defined three alignment quality levels: low (precision and recall 0.2), medium (precision and recall 0.5) and a high (precision and recall 0.8) quality. Figure 4 shows the results. As expected, when the alignment is good with respect to the interaction, best results are obtained with a high α, while for bad alignments it is better to make random choices. For medium quality, there is almost no difference, since the probability of a mapping being correct is similar to the one of choosing randomly the right option.

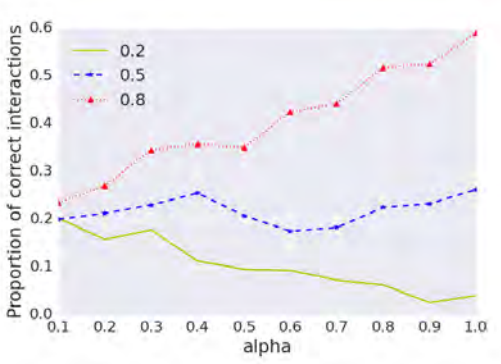


Figure 4: Success rates for different values of α

5.2 Estimating Pragmatic Precision and Recall

Although pragmatic precision and recall can be useful in practice, in most applications it is not realistic to expect agents to know them beforehand. In what follows we discuss how agents can use the experience of interaction to automatically estimate the values of precision and recall of an alignment. This would be useful not only to improve their behaviour as explained before, but also to evaluate alignments in a dynamic, distributed way.

Let us first focus on estimating recall. In this case, agents can simply use the proportion of the mappings they made in successful interactions that were already in $\mathcal{A}$.

$$recall_{est} = \frac{|\text{ mappings in successful interactions} \cap \mathcal{A} |}{|\text{ mappings in successful interactions }|}$$

Estimating precision is more complicated. A first attempt could be to consider:

$$precision_{est} = \frac{|\text{ mappings in successful interactions} \cap \mathcal{A} |}{|\text{ relevant mappings seen }|}$$

However, this considers as incorrect all the relevant mappings that were not part of successful interactions. This can sub-estimate the precision, particularly in the first steps, when an estimation is needed most.

Alternatively, we propose to use a learning strategy that estimates gradually the precision of $\mathcal{A}$ by analysing which of the mappings that were made are likely to be correct and which ones are not. A possibility is to use a technique proposed in [5], where all mappings start with a confidence equal to the one in $\mathcal{A}$ (or 0 if it is not a mapping in $\mathcal{A}$), and after an interaction they are updated as follows:

- After a successful interaction, the confidence in all mappings that were made is set to 1. These mappings are not updated in following interactions.
- After an unsuccessful interaction, a negative punishment is applied to the mappings made. At the same time, mappings are updated according to the quality of the aligning possibilities found later; if mappings with large confidence appeared as options after making one match, that match will increase its value.

To estimate precision, let *increased* be the set of all the mappings made that are in $\mathcal{A}$ and for which the calculated confidence is greater or equal to the one in $\mathcal{A}$. Precision can then be estimated as:

$$precision_{est} = \frac{|\text{ increased} \cap \mathcal{A} |}{|\text{ relevant mappings seen }|}$$

This can improve the precision estimation in early stages, since mappings that are likely to be correct (because many good mappings were found after them) would still increase their value. These are preliminary ideas, that we plan to further develop and evaluate experimentally in future work.

6 Conclusions

We consider the ideas presented in this paper to be a first step towards the development of ontology alignment tools that are particularly designed for agent interaction. These tools would require novel reasoning techniques that take into account contextual information about the tasks that are being performed to build

mappings of high pragmatic precision and recall. To this aim, a first technical requirement is the formalisation of a language that allows to express properties of the domain together with information about the interaction. To apply the ideas we propose here, it may be necessary to adapt them to more complex descriptions of interactions, or to incomplete ones.

References

1. Manuel Atencia and Marco Schorlemmer. Formalising interaction-situated semantic alignment: The communication product. In *Tenth International Symposium on Artificial Intelligence and Mathematics (ISAIM'08)*, Fort Lauderdale, Florida, USA, jan 2008.
2. Manuel Atencia and W. Marco Schorlemmer. An interaction-based approach to semantic alignment. *Journal of Web Semantics*, 12:131–147, 2012.
3. Paolo Bouquet, Jérôme Euzenat, Enrico Franconi, Luciano Serafini, Giorgos Stamou, and Sergio Tessaris. Specification of a common framework for characterizing alignment. Deliverable D2.2.1, Knowledge Web, 2004.
4. Alan Bundy and Fiona McNeill. Representation as a fluent: An ai challenge for the next half century. *IEEE Intelligent Systems*, 21(3):85–87, 2006.
5. Paula Chocron and Marco Schorlemmer. Attuning ontology alignments to semantically heterogeneous multi-agent interactions. In *Proceedings of the 22nd European Conference on Artificial Intellignce (ECAI)*, 2016 (to appear).
6. Jérôme Euzenat. Semantic precision and recall for ontology alignment evaluation. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, pages 348–353, San Francisco, CA, USA, 2007. Morgan Kaufmann Publishers Inc.
7. Jérôme Euzenat. First Experiments in Cultural Alignment Repair. In *Semantic Web: ESWC 2014 Satellite Events*, volume 8798, pages 115–130, 2014.
8. Jérôme Euzenat, Christian Meilicke, Heiner Stuckenschmidt, Pavel Shvaiko, and Cássia Trojahn. Ontology alignment evaluation initiative: Six years of experience. *Journal on Data Semantics XV*, pages 158–192, 2011.
9. Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.
10. Claudia V. Goldman, Martin Allen, and Shlomo Zilberstein. Learning to communicate in a decentralized environment. *Autonomous Agents and Multi-Agent Systems*, 15(1):47–90, August 2007.
11. Laura Hollink, Mark Van Assem, Shenghui Wang, Antoine Isaac, and Guus Schreiber. Two variations on ontology alignment evaluation: Methodological issues. In *Proceedings of the 5th European Semantic Web Conference on The Semantic Web: Research and Applications*, ESWC'08, pages 388–401, Berlin, Heidelberg, 2008. Springer-Verlag.
12. Fiona McNeill and Alan Bundy. Dynamic, automatic, first-order ontology repair by diagnosis of failed plan execution. *International Journal on Semantic Web and Information Systems*, 3(3), 2007.
13. Willem Robert Van Hage, Hap Kolb, and Guus Schreiber. Relevance-based evaluation of alignment approaches: The oaei 2007 food task revisited. In *Proceedings of the 3rd International Conference on Ontology Matching - Volume 431*, OM'08, pages 234–238, Aachen, Germany, Germany, 2008. CEUR-WS.org.

Tableau extensions for reasoning with link keys

Maroua Gmati[2,1], Manuel Atencia[1,2], and Jérôme Euzenat[2,1]

[1]Univ. Grenoble Alpes and [2]INRIA, France
gm.gmati.maroua@gmail.com,{Manuel.Atencia,Jerome.Euzenat}@inria.fr

Abstract. Link keys allow for generating links across data sets expressed in different ontologies. But they can also be thought of as axioms in a description logic. As such, they can contribute to infer ABox axioms, such as links, or terminological axioms and other link keys. Yet, no reasoning support exists for link keys. Here we extend the tableau method designed for $\mathcal{ALC}$ to take link keys into account. We show how this extension enables combining link keys with classical terminological reasoning with and without ABox and TBox and generate non trivial link keys.

1 Motivation

Part of the added value of linked data lies in the links between entities denoting the same individual in data sets issued by different sources as it allows for making inferences across data sets. For instance, links may identify the same books and articles in different bibliographical data sources. So finding the manifestation of the same entity across several data sets is an important task of linked data.

One way of identifying entities is to use link keys which generalise keys usually found in data bases to the case of different data sets. A link key [3] is a statement of the form:

$$\{\langle \mathsf{auteur}, \mathsf{creator}\rangle, \langle \mathsf{titre}, \mathsf{title}\rangle\}\ linkkey\ \langle \mathsf{Livre}, \mathsf{Book}\rangle$$

stating that whenever an instance of the class Livre has the same values for properties auteur and titre as an instance of class Book has for properties creator and title, then they denote the same entity. Such keys are slightly more complex than those of databases because, in RDF, properties are not necessarily functional (they may have several values) and their values may be other objects.

One further difference is that RDF data, together with ontologies expressed in the OWL or RDFS languages, are logic theories. In such a context, a link key is a statement as any other logical statement. As such, it may contribute deducing other statements. Indeed, the above link key entails:

$$\{\langle \mathsf{auteur}, \mathsf{creator}\rangle, \langle \mathsf{titre}, \mathsf{title}\rangle, \langle \mathsf{éditeur}, \mathsf{publisher}\rangle\}\ linkkey\ \langle \mathsf{Livre}, \mathsf{Book}\rangle$$

or

$$\{\langle \mathsf{auteur}, \mathsf{creator}\rangle, \langle \mathsf{titre}, \mathsf{title}\rangle\}\ linkkey\ \langle \mathsf{Livre}, \mathsf{Novel}\rangle$$

whenever Novel is subsumed by Book.

Hence, it is possible to reason on link keys in different ways:

– deducing link keys from OWL statements,
– deducing link keys from link keys,
– deducing OWL statements from link keys.

Our goal is to study reasoning procedures for link keys. For that purpose, we define a preliminary extension of the tableau method for $\mathcal{ALC}$ dealing with link keys and we provide examples for each of the inference types above.

In the following, we first discuss related work (§2) and define more precisely the problem (§3). Then we present a tableau extensions allowing for ABox reasoning with link keys (§4) and for reducing link key inference to that ABox reasoning (§5).

2 Related work

Data interlinking is a very active area [9]. Two main approaches are used for coping with this problem: numerical methods and logical methods. The numerical methods usually compute a similarity between resources based on their property values to establish links between those which are highly similar [11; 13]. Logical methods for data interlinking use an axiomatic characterisation of what makes two resources the same to find the links between different data sets [12; 1; 3].

This work belongs to the logic-based approach. It uses a generalisation of keys in relational databases, called link keys, for expressing the condition for identifying resources across different ontologies. Keys in databases indicate that a set of properties uniquely identifies individuals. Relational properties are functional (have only one value) and concrete (the value comes from a data type).

RDF data differ from relational data in their properties, which are not functional, and their values, which may be resources. Hence, keys have been generalised to cope with this problem [2]. RDF property values are considered the same if they are the same concrete value or are interpreted as the same individual. Coping with non functionality lead to define two different types of keys: in-keys and eq-keys. Eq-keys require that the properties of two objects have exactly the same values for them to be equal, while in-keys only require that each property shares at least one common value. In this work, we focus on in-keys.

Keys may be introduced in description logics either as global constraints in a specific KBox [7; 10], or as a new concept constructor [6]. [7] discusses the introduction of keys in the $\mathcal{DLR}$ logic but does not provide any reasoning method. Keys based on features (functional roles whose value is from a concrete domain) have been introduced within the $\mathcal{ALCOK}(\mathcal{D})$ and $\mathcal{SHROIK}(\mathcal{D})$ logics [10] and an extension of the tableau method has been provided to deal with these logics.

Keys identify objects within a single data source with a single schema. Link keys have been designed for coping with heterogeneous data sources [8]. They can be seen either as a generalisation of keys across two data sets or as a merge between keys and alignments. They express conditions by which two individuals, from two different classes, must be considered the same by comparing values of properties.

Link keys raise two distinct problems: the first one is to extract link keys from data sets [3]; the second one is to take advantage of link keys to generate links. These two

problems may be thought of as two steps of a link generation procedure: first extract link keys, then generate links from them.

Here we tackle a third problem (not unrelated to the second one): reasoning with link keys, i.e., infering links, ontological and assertional statements as well as other link keys. We define this problem more precisely below.

3 Preliminaries

Data interlinking is the process of generating links across data sets that can help finding equivalent resources representing the same entity on the web for linked data. These links are usually owl:sameAs statements between two resources across different RDF data sets. We will consider that these data sets are description logic knowledge bases ($KB = \langle T, A\rangle$) made of a TBox T and an ABox A. Description logics [4] are at the basis of OWL, so this is quite natural.

We decided to extend the tableau method used for checking entailment in the $\mathcal{ALC}$ family of description logics for several reasons:

- $\mathcal{ALC}$ is a subset of OWL;
- The tableau method is extensible, so it is possible to add rules for dealing with more expressive logics. We could have started with procedure specific to less expressive logics ($\mathcal{EL}$, DL-Lite, OWL-RL), but we could barely extend them.

An $\mathcal{ALC}$ TBox is a set of general concept inclusion axioms of the form $C \sqsubseteq C'$. Concepts are defined by:

$$C = A|\bot|\top|C \sqcap C'|C \sqcup C'|\neg C|\forall R.C|\exists R.C$$

and roles are simply atomic roles ($R = r$).

The ABox is made of assertions of the form $C(a)$ and $r(a, b)$. We will use two specific statements $a = b$ and $\neq b$ which are interpreted as usual. These two predicates are the transcription of `owl:sameAs` and `owl:differentFrom`.

The semantics of such logics is defined by interpretations $\mathcal{I} = \langle \Delta^\mathcal{I}, \cdot^\mathcal{I}\rangle$ such that $\Delta^\mathcal{I}$ is a non empty set and $\cdot^\mathcal{I}$ is a function such that: $a^\mathcal{I} \in \Delta^\mathcal{I}$, $C^\mathcal{I} \subseteq \Delta^\mathcal{I}$, and $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$ with:

$$\begin{array}{ll} (\neg C)^\mathcal{I} = \Delta^\mathcal{I} \setminus C^\mathcal{I} & \bot^\mathcal{I} = \varnothing \qquad \top^\mathcal{I} = \Delta^\mathcal{I} \\ (C \sqcap C')^\mathcal{I} = C^\mathcal{I} \cap C'^\mathcal{I} & (\forall r.C)^\mathcal{I} = \{\delta \in \Delta^\mathcal{I} | \forall \delta'; \langle \delta, \delta'\rangle \in r^\mathcal{I} \Rightarrow \delta' \in C^\mathcal{I}\} \\ (C \sqcup C')^\mathcal{I} = C^\mathcal{I} \cup C'^\mathcal{I} & (\exists r.C)^\mathcal{I} = \{\delta \in \Delta^\mathcal{I} | \exists \delta' \in C^\mathcal{I}; \langle \delta, \delta'\rangle \in r^\mathcal{I}\} \end{array}$$

An interpretation satisfies an axiom (denoted by $\mathcal{I} \models \alpha$) in the following conditions:

$$\begin{array}{ll} \mathcal{I} \models C(a) \text{ iff } a^\mathcal{I} \in C^\mathcal{I} & \mathcal{I} \models r(a, b) \text{ iff } \langle a^\mathcal{I}, b^\mathcal{I}\rangle \in r^\mathcal{I} \\ \mathcal{I} \models a = b \text{ iff } a^\mathcal{I} = b^\mathcal{I} & \mathcal{I} \models a \neq b \text{ iff } a^\mathcal{I} \neq b^\mathcal{I} \\ \mathcal{I} \models C \sqsubseteq C' \text{ iff } C^\mathcal{I} \subseteq C'^\mathcal{I} & \end{array}$$

A model of a knowledge base KB is an interpretation satisfying all its axioms and an assertion α is entailed by a knowledge base (denoted by $KB \models \alpha$) if it is satisfied by all the models of KB.

We extend description logics with a KBox K which contains link keys instead of simple keys. The KBox is a set of link keys: $\{\langle p_i, q_i\rangle\}_{i\in I}\ linkkey^w_{in}\ \langle C, D\rangle$ with C and D two classes coming from different data sets and p_i and q_i roles, from the data sets of C and D respectively, indexed by a finite set of indices I. Since we concentrate specifically on weak in-link keys, we use the keyword $linkkey^w_{in}$.

The semantics of description logics is extended to cover link keys: An interpretation $\mathcal{I}$ satisfies $(\{\langle p_i, q_i\rangle\}_{i\in I}\ linkkey^w_{in}\ \langle C, D\rangle)$ iff, for any $\delta \in C^{\mathcal{I}}$ and $\eta \in D^{\mathcal{I}}$,

$$\bigwedge_{i\in I}(\exists z_i \in \Delta^{\mathcal{I}}; \langle \delta, z_i\rangle \in p_i^{\mathcal{I}} \wedge \langle \eta, z_i\rangle \in q_i^{\mathcal{I}}) \Rightarrow \delta = \eta$$

Any key $\{p_i\}_{i\in I}\ keyFor\ C$ is equivalent to the link key $\{\langle p_i, p_i\rangle\}_{i\in I}\ linkkey\ \langle C, C\rangle$. In this paper, we only consider hierarchical KBoxes, i.e., KBoxes in which there cannot be circular dependencies between link keys.

It is possible, to establish entailment rules for link keys considered as assertions:

$$\{\langle p_i, q_i\rangle\}_{i\in I}\ linkkey^w_{in}\ \langle C, D\rangle \models \{\langle p_i, q_i\rangle\}_{i\in I\cup J}\ linkkey^w_{in}\ \langle C, D\rangle$$
$$\{\langle p_i, q_i\rangle\}_{i\in I}\ linkkey^w_{in}\ \langle C, D\rangle, C' \sqsubseteq C \models \{\langle p_i, q_i\rangle\}_{i\in I}\ linkkey^w_{in}\ \langle C', D\rangle$$
$$\{\langle p_i, q_i\rangle\}_{i\in I}\ linkkey^w_{in}\ \langle C \sqcup C', D\rangle \models \{\langle p_i, q_i\rangle\}_{i\in I}\ linkkey^w_{in}\ \langle C, D \sqcap D'\rangle$$

Proving all such rules one by one is tedious, so an inference procedure for doing this would be useful.

4 Links and Abox entailments with link keys

The basic way of applying link keys is to start with two datasets A and A' described by two ontologies T and T' and a set K of link keys across these ontologies and to generate links, i.e., statements of the form $a = b$ with a and b from each data set.

We consider this problem more widely as that of reasoning in a knowledge base[1] $KB = \langle T \cup T', K, A \cup A'\rangle$. We will consider more precisely the decision problem of checking the entailment of any ABox axiom α from such a knowledge base.

Problem: ABOX AXIOM ENTAILMENT
INSTANCE:
– A knowledge base $KB = \langle T, K, A\rangle$
– An ABox assertion α.
QUESTION: Does $KB \models \alpha$?

4.1 Tableau rule for applying link keys

The tableau method is the classical technique to reason with $\mathcal{ALC}$. Explaining the method is out of the scope of this paper (see [4; 5]). To summarise, this method attempts to find a model of a knowledge base $KB = \langle T, A\rangle$ in negation normal form.

[1] We assume no unwanted name conflicts, i.e., the same name or URI in both data sets must have the same interpretation.

For that purpose, it starts with a representation of the ABox A and applies rules (see Appendix) guided by T until no rule is applicable [5]. In such a case, there exists a model of KB. However, there are special constraints, called clashes, which express the impossibility to build a model: if such a clash is satisfied, then the current representation cannot be turned into a model and the algorithm must explore eventual alternative representations. Finally, for guaranteeing the termination of the process due to infinitely expanding rules, provisions are taken for detecting this and blocking some parts of the representation to be expanded. We rely here on the classical tableau method for $\mathcal{ALC}$ and use a graphical representation of partial models in which nodes (x) represent individuals labeled ($L(x)$) by sets of class descriptions and edges ($\langle x, y\rangle$) represent relations labeled ($L(\langle x, y\rangle)$ by role descriptions. The tableau method may be used for finding a model or for proving that there exist no model of a knowledge base.

In order to tackle the ABox Axiom entailment problem within the tableau method we introduce the Linkkey-rule:

Linkkey-rule

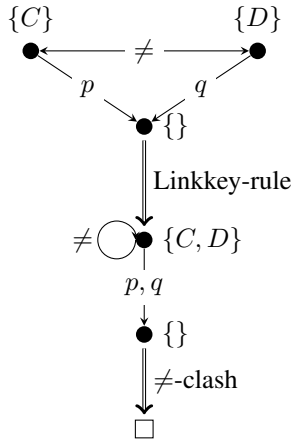
Condition: $\{\langle p_i, q_i\rangle\}_{i\in I}\ linkkey^{w}_{in}\ \langle C, D\rangle \in K$,
$\exists x, y$, not blocked, such that $C \in L(x)$, $D \in L(y)$, and
$\forall i \in I$, $\exists z_i$, such that $p_i \in L(\langle x, z_i\rangle)$ and $q_i \in L(\langle y, z_i\rangle)$

Action: $L(x) := L(x) \cup L(y)$
Replace y by x in all edges starting from or ending at y
Suppress node y

This rule is sound, i.e., any model has to satisfy it, as it strictly follows the semantics of link keys. It generalises rule T14 in [10] to link keys.

The use of this rule for checking a link $a = b$ can be illustrated on the straightforward Example 1: For proving the entailment of $a = b$, we proceed by refutation, i.e., we prove that it is not possible to create a model satisfying the antecedents and the negation of the consequence ($a \neq b$). A representation of such a model is created and the rules are applied on it. The Linkkey-rule merges the two nodes satisfying the link key condition which makes them fall under the $\neq$-clash.

Example 1 (Simple link generation).



Problem:
$\langle p, q\rangle\ linkkey^{w}_{in}\ \langle C, D\rangle$,
$C(a), D(b), p(a, v), q(b, v)$
$\models a = b$?

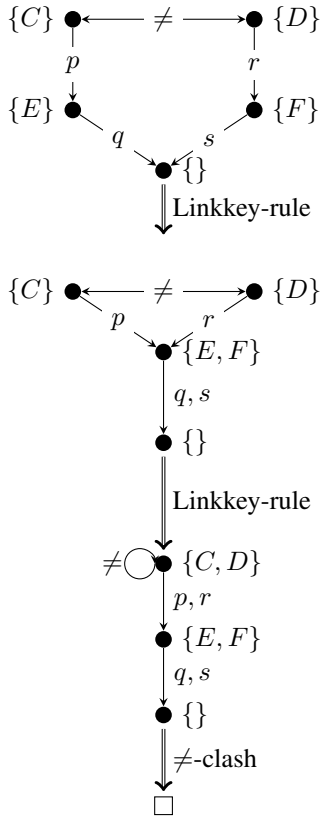
Knowledge base:

$$T = \{\}$$
$$K = \{\langle p, q\rangle\ linkkey^{w}_{in}\ \langle C, D\rangle\}$$
$$A = \{C(a), D(b), p(a, v), q(b, v), a \neq b\}$$

4.2 Combining link key reasoning and ABox reasoning

Example 2 shows the use of these rules for chaining the use of two link keys. However, it may be used in any ABox reasoning development.

Example 2 (Chaining link generation).



Problem:
$\langle p, r\rangle\ linkkey_{in}^{w}\ \langle C, D\rangle$,
$\langle q, s\rangle\ linkkey_{in}^{w}\ \langle E, F\rangle$,
$C(a), p(a, c), E(c), q(c, v), D(b), r(b, d), F(d), s(d, v)$
$\models a = b$?

Knowledge base:

$T = \{\}$
$K = \{\langle p, r\rangle\ linkkey_{in}^{w}\ \langle C, D\rangle, \langle q, s\rangle\ linkkey_{in}^{w}\ \langle E, F\rangle\}$
$A = \{C(a), p(a, c), E(c), q(c, v),$
$\quad D(b), r(b, d), F(d), s(d, v), a \neq b\}$

Solving the ABox entailment problem may not be the most efficient way to generate links from RDF especially if the size of the considered ABox is very large. A more interesting use of such reasoning is for checking link key entailment.

5 Link key entailment

The link key entailment problem aims at checking if a link key is entailed by a knowledge base. Because this resorts to the terminological level, i.e., without regard to a particular ABox, it is defined only on a knowledge base made of a TBox and a KBox. Indeed, some link keys may be entailed from terminological axioms, some others from other link keys of a mix of this.

Problem: LINK KEY ENTAILMENT
INSTANCE:
– A knowledge base $KB = \langle T, K \rangle$
– A link key λ.
QUESTION: Does $KB \models \lambda$?

5.1 Reducing link key entailment to knowledge base satisfiability

The tableau method cannot be directly used for refuting a link key axiom because there is no negation for link keys: a link key is an axiom of our logic, the negation of a link key is not.

Other authors have considered expressing keys as simple concept constructors [6]:

$$C \sqsubseteq \mathsf{key}(\{p_i\}_{i \in I})$$

This could be transposed for link keys as:

$$\langle C, D \rangle \sqsubseteq \ linkkey^{w}_{in}\ (\{\langle p_i, q_i \rangle\}_{i \in I})$$

such statements would solve half of the problem as it is possible to negate the subsumption statements, but this would lead to strange statements as they concern pairs of classes. They would also be stronger than, and not equivalent to, our actual link key statements.

Adding the negation of a link key to the logic is another solution to this problem. However, since its only use would be for the decision procedure, we preferred to avoid this solution.

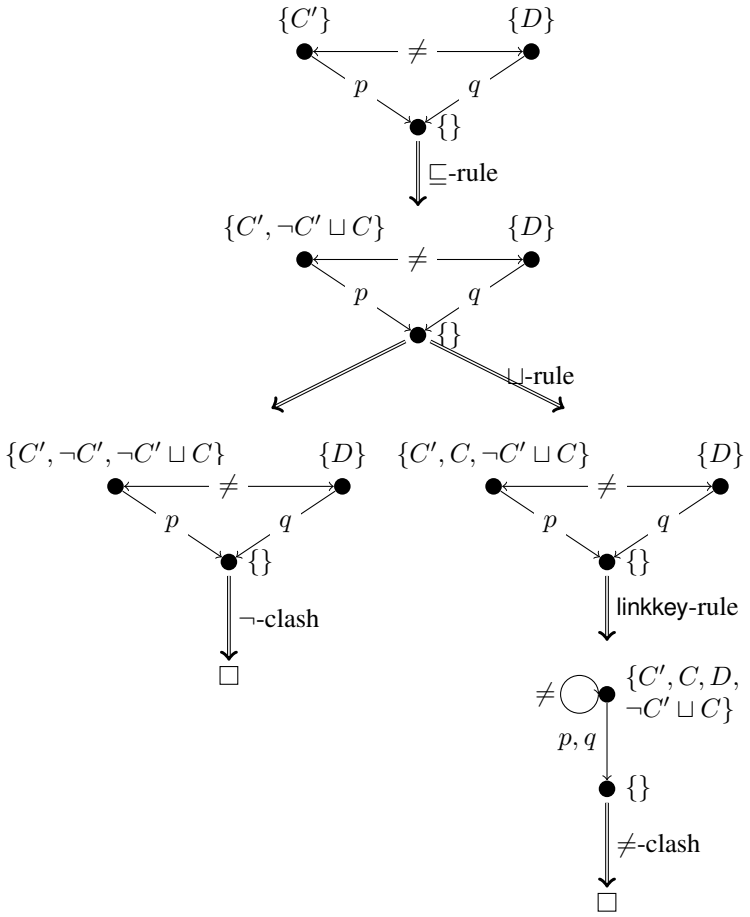
We choose a simpler method given that our goal is simply to have negated link keys as the statement to refute: we use a set of ABox statements as witness of the unsatisfiability of a link key. This set is given by the function ρ:

$$\rho(\{\langle p_i, q_i \rangle\}_{i \in I}\ linkkey^{w}_{in}\ \langle C, D \rangle) = \{C(x), D(y), x \neq y\} \cup \{p_i(x, v_i), q_i(y, v_i)\}_{i \in I}$$

Checking the entailment of a link key λ by a knowledge base $\langle \varnothing, T, K \rangle$ can be reduced to checking the satisfiability of the knowledge base $KB = \langle T, K, \rho(\lambda) \rangle$. Any model in which the link key λ is not valid satisfies $\rho(\lambda)$. Hence, if KB is satisfiable, then λ is not entailed.

Example 3 (Link key inference from other link keys and TBox).
Problem: $\langle p, q\rangle\ linkkey_{in}^{w}\ \langle C, D\rangle, C' \sqsubseteq C \models \langle p, q\rangle\ linkkey_{in}^{w}\ \langle C', D\rangle$?
Knowledge base:

$$T = \{C' \sqsubseteq C\}$$
$$K = \{\langle p, q\rangle\ linkkey_{in}^{w}\ \langle C, D\rangle\}$$
$$A = \{C'(a), D(b), p(a, v), q(b, v), a \neq b\}$$



For instance, one of the example given in Section 3 is a link key entailed from another link key and terminological axioms. Example 3 shows how this is performed without introducing any new rule or clash in the tableau procedure.

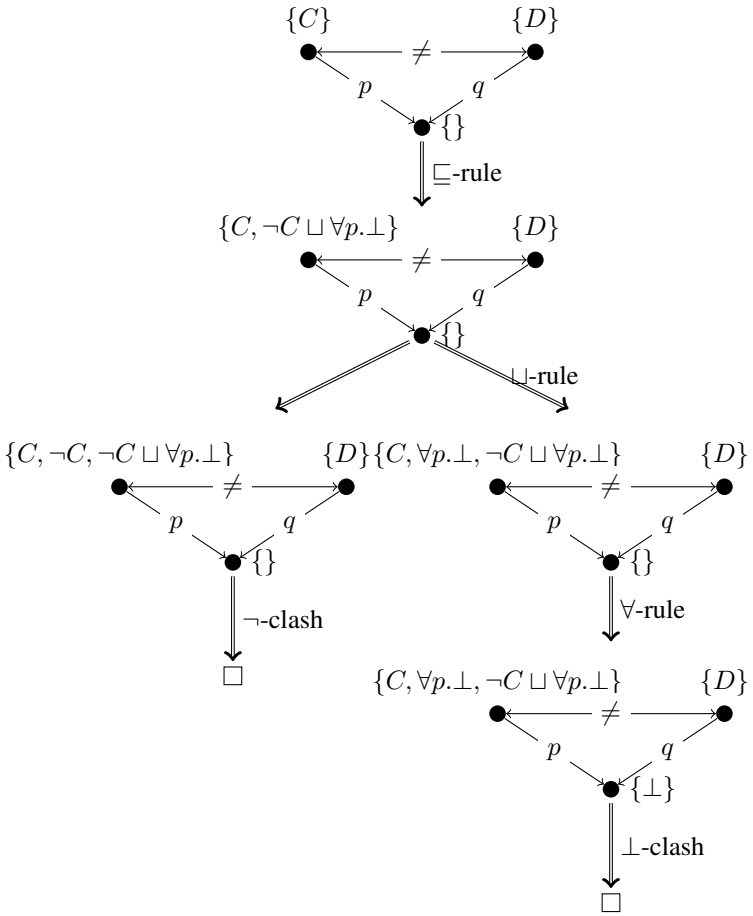
This shows the importance of being able to reason with the ABox, since the refutation of the KB is mostly carried out by reasoning in the ABox even if the problem does not have an ABox. It also shows that link key rules can be adequately interleaved with $\mathcal{ALC}$ rules. This suggests that extensions can properly work in the same way.

5.2 Link key entailed from terminological axioms

Some other link keys may only be entailed by terminological axioms. We illustrate this by the counter-intuitive Example 4. This inference is of little use, but it shows that the method indeed proves this valid link key.

Example 4 (Link key inference from TBox alone).
Problem:
$C \sqsubseteq \forall p.\bot \models \langle p, q\rangle\ linkkey^{w}_{in}\ \langle C, D\rangle$?
Knowledge base:

$$T = \{C \sqsubseteq \forall p.\bot\}$$
$$K = \{\}$$
$$A = \{C(a), p(a, v), D(b), q(b, v), a \neq b\}$$



It is noteworthy that Example 4 does not use the Linkkey-rule; it only relies on the encoding of the problem and classical $\mathcal{ALC}$ reasoning.

The use of the tableau method allows both to check inference rules and to determine minimal logics in which they hold. Example 4 shows that the given entailment holds in any description logic, with $\mathcal{ALC}$-style models, which accepts subsumption axioms ($\sqsubseteq$), universal quantification ($\forall$) and the empty concept ($\bot$).

6 Conclusion and future work

Link keys are very useful for generating links from data sources, but they can be studied independently from data sources as axioms. In order to prove when a particular knowledge base, eventually with link keys, entails a particular link key, we proposed extensions of the tableau method for $\mathcal{ALC}$ enabling the interpretation of link keys. We showed that these extensions also allow for checking link key entailment.

We considered the tableau method because it is well-adapted to $\mathcal{ALC}$ and thus to OWL as a whole. Weaker fragments of OWL ($\mathcal{EL}$, DL-Lite, OWL-RL) are supported efficiently by other reasoning methods. It would be interesting to investigate the opportunity to reason with and about link keys in this context.

This work is preliminary and many developments may be undertaken from here. We discuss a few of them.

First, we need to determine the properties of the proposed extension. We have yet no formal proof to offer, but basic arguments for these. Although correctness of rules and clash independently seems to be straightforward, proving the completeness of the designed procedure with various logics must be considered. Termination can be guaranteed with a blocking mechanisms and because no rule erases any other rule condition (the Linkkey-rule merges nodes, but preserves the constraints on these nodes). Finally, the current link key rule should not increase the complexity of existing tableau methods since the rule does not introduce branches. The Linkkey-rule may offer new development opportunities by merging nodes but (*i*) this process is bounded, and (*ii*) new tableau developments should not go beyond current complexity.

Then, we want to implement these extensions. This would allow us to check automatically the link key inference rules that we designed. It would also be interesting, in a further step, to develop techniques to generate (specific) entailed assertions in a forward deduction style.

Finally, it would be worth considering the other type of link key conditions (eq-link keys). However, this may not be easy to integrate with the open world aspect of description logic semantics.

A $\mathcal{ALC}$+Linkkey rules

We provide the full set of rules for helping the reader to read the examples.

A.1 Completion rules

$\sqcap$-rule
Condition: $C \sqcap D \in L(x)$, x is not blocked; $\{C, D\} \not\subseteq L(x)$
Action: $L(x) := L(x) \cup \{C, D\}$

$\sqcup$-rule
Condition: $C \sqcup D \in L(x)$, x is not blocked; $C \notin L(x)$, $D \notin L(x)$
Action: $L(x) := L(x) \cup \{C\}$, or $L(x) := L(x) \cup \{D\}$

$\exists$-rule
Condition: $\exists r.C \in L(x)$, x is not blocked; $\nexists y; r(x, y) \wedge C \in L(y)$
Action: create a new node y with $L(\langle x, y\rangle) = \{r\}$ and $L(y) = \{C\}$

$\forall$-rule
Condition: $\forall r.C \in L(x)$, x is not blocked; $\exists y; r(x, y) \wedge C \notin L(y)$
Action: $L(y) := L(y) \cup \{C\}$

$\sqsubseteq$-rule
Condition: $C \sqsubseteq D \in T$, x is not blocked, $\neg C \sqcup D \notin L(x)$
Action: $L(x) := L(x) \cup \{\neg C \sqcup D\}$

Linkkey-rule
Condition: $\{\langle p_i, q_i\rangle\}_{i \in I}\ linkkey^{w}_{in}\ \langle C, D\rangle \in K$,
$\exists x, y$, not blocked, such that $C \in L(x)$, $D \in L(y)$, and
$\forall i \in I$, $\exists z_i$, such that $p_i \in L(\langle x, z_i\rangle)$ and $q_i \in L(\langle y, z_i\rangle)$
Action: $L(x) := L(x) \cup L(y)$
Replace y by x in all edges starting from or ending at y
Suppress node y

A.2 Clash conditions

$$\begin{aligned}
&\neg\textbf{-clash}: && \exists x; \{C, \neg C\} \subseteq L(x)\\
&\bot\textbf{-clash}: && \exists x; \bot \in L(x)\\
&\neq\textbf{-clash}: && \exists \langle x, x\rangle; \neq \in L(\langle x, x\rangle)
\end{aligned}$$

References

1. Mustafa Al-Bakri, Manuel Atencia, Steffen Lalande, and Marie-Christine Rousset. Inferring same-as facts from linked data: an iterative import-by-query approach. In Blai Bonet and Sven Koenig, editors, *Proc. 29th Conference on Artificial Intelligence (AAAI), Austin (TX US)*, pages 9–15, 2015.
2. Manuel Atencia, Michel Chein, Madalina Croitoru, Jérôme David, Michel Leclère, Nathalie Pernelle, Fatiha Saïs, François Scharffe, and Danai Symeonidou. Defining key semantics for the RDF datasets: experiments and evaluations. In *Proc. 21st International Conference on Conceptual Structures (ICCS), Iasi (RO)*, pages 65–78, 2014.
3. Manuel Atencia, Jérôme David, and Jérôme Euzenat. Data interlinking through robust linkkey extraction. In *Proc. 21st european conference on artificial intelligence (ECAI), Praha (CZ)*, pages 15–20, 2014.
4. Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter Patel-Schneider, editors. *The description logic handbook: theory, implementations and applications*. Cambridge University Press, 2003.
5. Franz Baader, Ian Horrocks, and Ulrike Sattler. Description logics. In Frank van Harmelen, Vladimir Lifschitz, and Bruce Porter, editors, *Handbook of Knowledge Representation*, chapter 3, pages 135–179. Elsevier, Amsterdam (NL), 2008.
6. Alexander Borgida and Grant Weddell. Adding uniqueness constraints to description logics (preliminary report). In *Proc. 5th Deductive and Object-Oriented Databases conference (DOOD)*, volume 1341 of *LNCS*, pages 85–102, Montreux (CH), 1997.
7. Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Moshe Vardi. View-based query processing for regular path queries with inverse. In *Proceedings of the 19th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS)*, pages 58–66, 2000.
8. Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer, Heidelberg (DE), 2 edition, 2013.
9. Alfio Ferrara, Andriy Nikolov, and François Scharffe. Data linking for the semantic web. *International Journal of Semantic Web and Information Systems*, 7(3):46–76, 2011.
10. Carsten Lutz, Carlos Areces, Ian Horrocks, and Ulrike Sattler. Keys, nominals, and concrete domains. *Journal of Artificial Intelligence Research*, 23:667–726, 2005.
11. Axel-Cyrille Ngonga Ngomo and Sören Auer. LIMES: A time-efficient approach for large-scale link discovery on the web of data. In *Proc. 22nd International Joint Conference on Artificial Intelligence (IJCAI)*, pages 2312–2317, Barcelona (ES), 2011.
12. Fatiha Saïs, Nathalie Pernelle, and Marie-Christine Rousset. Combining a logical and a numerical method for data reconciliation. *Journal on Data Semantics*, 12:66–94, 2009.
13. Julius Volz, Christian Bizer, Martin Gaedke, and Georgi Kobilarov. Discovering and maintaining links on the web of data. In *Proc. 8th International Semantic Web Conference (ISWC)*, volume 5823 of *Lecture notes in computer science*, pages 650–665, Chantilly (VA US), 2009.

Rewriting SELECT SPARQL queries from 1:n complex correspondences

Élodie Thiéblin, Fabien Amarger, Ollivier Haemmerlé, Nathalie Hernandez, Cassia Trojahn

IRIT & Université de Toulouse 2 Jean Jaurès, Toulouse, France
elodie.thieblin@gmail.com,{fabien.amarger, ollivier.haemmerle, nathalie.hernandez,cassia.trojahn}@irit.fr

Abstract. This paper presents a mechanism for rewriting SPARQL queries based on complex ontology correspondences. While the usefulness of simple correspondences, involving single entities from both source and target ontologies, has long been recognized, query rewriting requires more expressive links between ontology entities expressing the true relationships between them. Here, complex correspondences, in the format 1:n, between overlapping ontologies are exploited for rewriting `SELECT` SPARQL queries, so that they can be expressed over different RDF data sets in the Linked Open Data. Our approach has been evaluated using two data sets, one from the agriculture domain and another based on a reduced set involving the ontologies from the OAEI Conference track.

1 Introduction

A SPARQL query is intrinsically related to the ontological model that describes the RDF source. To federate knowledge from different sources described by various ontologies, a SPARQL query must be adapted to each of the knowledge bases. In order to use the Linked Open Data potential at its best, it is important to bridge the gap of semantic heterogeneity between knowledge bases. Ontology matching [5] is a solution for finding correspondences (i.e., an alignment) between two ontologies. There are two types of correspondences : simple correspondences and complex correspondences. A simple correspondence matches an element from the first ontology to its semantically related ontological element in the second ontology. Nevertheless, simple correspondences cannot cover every case of use because of the model differences between ontological sources. Complex correspondences palliate the lack of expressiveness of simple alignments. They extend simple correspondences to correspondences between complex constructions of ontological entities of the two ontologies.

Simple correspondences can be easily used to transform SPARQL queries. The usual approach (integrated in the Alignment API [3]) is to replace the IRI of an ontological entity in the initial query by the IRI of its corresponding entity. This approach considers that the correspondence stands for an equivalence relation. However by using simple correspondences, not all SPARQL queries can be transformed. In this paper, an approach that exploits complex correspondences

for SPARQL query transformation is proposed. Even though there are only a few systems able to automatically generate them, manually drawing correspondences used in the proposed mechanism is likely a less fastidious task than manually rewriting every SPARQL query for each new RDF-triple store.

Our approach is based on a set of rules for rewriting a subset of `SELECT` SPARQL queries from complex correspondences involving an equivalence relation between ontology entities. These correspondences are expressed in EDOAL, a language proposed for representing complex correspondences. The approach has been validated on two data sets. The first one was built to meet the needs of agriculture experts willing to find cross knowledge about agronomic taxons between DBpedia and a dedicated knowledge base. The second data set was inspired from a subset of queries from the OAEI oa4qa[1] task data set and could be further developed in order to enrich this track.

The rest of the paper is structured as follows. §2 introduces ontology matching and the EDOAL syntax. §3 discusses related work and §4 presents the rewriting rules on which our approach is based. §5 discusses the validation of the approach and §6 concludes the paper and presents perspectives for future work.

2 Ontology matching

Matching two ontologies is the process of generating an alignment A between two ontologies $\mathcal{O}$ and $\mathcal{O}'$. A is directional, denoted $A_{\mathcal{O}\rightarrow\mathcal{O}'}$:

Definition 1 (Alignment). *An alignment $A_{\mathcal{O}\rightarrow\mathcal{O}'}$ is a set of correspondences $A_{\mathcal{O}\rightarrow\mathcal{O}'} = \{c_1, c_2, ..., c_n\}$, where each c_i is a triple $\langle e_{\mathcal{O}}, e'_{\mathcal{O}}, r\rangle$:*

- *if the correspondence c_i is **simple**, both $e_{\mathcal{O}}$ and $e_{\mathcal{O}'}$ stand for one and only one entity (i.e., a class or a property) (1:1);*
- *if the correspondence is **complex**, at least one of $e_{\mathcal{O}}$ or $e_{\mathcal{O}'}$ involves one or more entities in a logical formulation. The correspondence is therefore (1:n), (m:1) or (m:n), where $e_{\mathcal{O}}$ refers to a subset of elements $\in \mathcal{O}$, and $e_{\mathcal{O}'}$ refers to a subset of elements $\in \mathcal{O}'$. The elements of $e_{\mathcal{O}}$, resp. $e_{\mathcal{O}'}$ form a logical construction using the constructors of a formal language (First-Order Logic or Description Logics);*
- *r is a relation, e.g., equivalence ($\equiv$), more general($\sqsupseteq$), more specific ($\sqsubseteq$), holding between $e_{\mathcal{O}}$ and $e_{\mathcal{O}'}$;*

The alignment $A_{\mathcal{O}\rightarrow\mathcal{O}'}$ is said *complex* if it contains at least one complex correspondence. A correspondence c_i, can also be noted $e_{\mathcal{O}}\ r\ e_{\mathcal{O}'}$, as for the complex correspondences in the following.

$$Chairman \equiv Demo\_Chair \sqcup OC\_Chair \sqcup PC\_Chair \sqcup Session\_Chair \sqcup Tutorial\_Chair \sqcup Workshop\_Chair \quad (1)$$

$$Accepted\_Paper \equiv Paper \sqcap \exists hasDecision.Acceptance \quad (2)$$

[1] `http://oaei.ontologymatching.org/2015/oa4qa/index.html`

Example 1 expresses a complex correspondence between entities from Cmt[2] and Ekaw[3] ontologies. It states that the concept *Chairman* of Cmt is equivalent to the union of the concepts *Demo_Chair*, *OC_Chair*, *PC_Chair*, *Session_Chair*, *Tutorial_Chair* and *Workshop_Chair* of Ekaw. Example 2 expresses a complex correspondence, where the concept *Accepted_Paper* of Ekaw is equivalent to the concept *Paper* of Cmt for which the domain of the object property *hasDecision* is restricted to an individual of the type *Acceptance*.

For representing complex correspondences, the EDOAL language has been proposed [6, 3]. It is fit to express simple and complex matching of cardinality (1:1), (1:n), (n:1) and (n:m). The entities $e_{\mathcal{O}}$ and $e_{\mathcal{O}'}$ are represented by expressions (class, relation or property expressions) that can be an ID (or IRI), a construction or a restriction. For a detailed description of EDOAL syntax the reader can refer to [6]. We illustrate this syntax with the following examples of 1:n complex correspondences given above. We use the prefixes `ekaw:<http://ekaw#>`, `cmt:<http://cmt#>`. Example 1 presents a *class expression* involving a class construction built with a union of concepts, while Example 2 expresses a *class expression* involving an attribute domain restriction.

Example 1

```
<entity1>
  <edoal:Class rdf:about="&cmt;Chairman"/>
</entity1>
<entity2>
<edoal:Class>
  <edoal:or rdf:parseType="Collection">
    <edoal:Class rdf:about="&ekaw;Demo_Chair"/>
    <edoal:Class rdf:about="&ekaw;OC_Chair"/>
     <edoal:Class rdf:about="&ekaw;PC_Chair"/>
     <edoal:Class rdf:about="&ekaw;Session_Chair"/>
    <edoal:Class rdf:about="&ekaw;Tutorial_Chair"/>
    <edoal:Class rdf:about="&ekaw;Workshop_Chair"/>
  </edoal:or>
</entity2>
<measure rdf:datatype="&xsd;float">1.0</measure>
<relation>Equivalence</relation>
```

Example 2

```
<entity1>
  <edoal:Class rdf:about="&ekaw;Accepted_Paper"/>
</entity1>
<entity2>
  <edoal:Class>
    <edoal:and rdf:parseType="Collection">
      <edoal:Class rdf:about="&cmt;Paper"/>
        <edoal:AttributeDomainRestriction>
          <edoal:onAttribute>
            <edoal:Relation rdf:about="&cmt;hasDecision"/>
          </edoal:onAttribute>
          <edoal:class>
            <edoal:Class rdf:about="&cmt;Acceptance"/>
          </edoal:class>
        </edoal:AttributeDomainRestriction>
    </edoal:and>
  </edoal:Class>
</entity2>
<measure rdf:datatype="&xsd;float">1.0</measure>
<relation>Equivalence</relation>
```

Additional examples of correspondences expressed in EDOAL are presented in examples 3 and 4. Example 3 shows a correspondence where the relation *writtenBy* of Ekaw is equivalent to *a relation expression* constructed with the inverse of the relation *writePaper* in Cmt. Example 4, involving the ConfOf[4] ontology (prefix `confOf:<http://confOf#>`), gives an example of a *class expression* constructed with an attribute value restriction stating that in Ekaw an *Early-Registered_Participant* is a *Participant* for which the value of the data property *earlyRegistration* is equal to *true* in ConfOf.

[2] `http://oaei.ontologymatching.org/2015/conference/data/cmt.owl`
[3] `http://oaei.ontologymatching.org/2015/conference/data/ekaw.owl`
[4] `http://oaei.ontologymatching.org/2015/conference/data/confof.owl`

Example 3

```
<entity1>
  <edoal:Relation rdf:about="&ekaw;writtenBy"/>
</entity1>
<entity2>
  <edoal:Relation>
    <edoal:inverse>
      <edoal:Relation rdf:about="&cmt;writePaper"/>
    </edoal:inverse>
  </edoal:Relation>
</entity2>
<measure rdf:datatype="&xsd;float">1.0</measure>
<relation>Equivalence</relation>
```

Example 4

```
<entity1>
  <edoal:Class rdf:about="&ekaw;Early-Registered_Participant"/>
</entity1>
<entity2>
  <edoal:Class>
    <edoal:and rdf:parseType="Collection">
      <edoal:Class rdf:about="&confOf;Participant"/>
      <edoal:AttributeValueRestriction>
        <edoal:onAttribute>
          <edoal:Property rdf:about="&confOf;earlyRegistration"/>
        <edoal:onAttribute>
        <edoal:comparator rdf:resource="&edoal;equals"/>
        <edoal:value>
          <edoal:Literal edoal:type="xsd;boolean" edoal:string="true"/>
        </edoal:value>
      </edoal:AttributeValueRestriction>
    </edoal:and>
  </edoal:Class>
</entity2>
<measure rdf:datatype="&xsd;float">1.0</measure>
<relation>Equivalence</relation>
```

3 Related Work

A naive approach for rewriting SPARQL queries consists in replacing the IRI of an entity of the initial query by the corresponding IRI in the alignment, using simple correspondences. This approach is integrated in the Alignment API [3]. However, it does not take into account the specific kind of relation expressed in the correspondence (e.g., generalisation or specialization). The approach in Euzenat *et al.* [4] aims at writing `CONSTRUCT` SPARQL queries from complex alignments. A new knowledge base expressed with the source ontology vocabulary is populated with the instances of the target knowledge base. A rewriting approach not limited to queries of type `CONSTRUCT` and that takes advantage of complex (1:n) alignments has been proposed by Correndo *et al.* [1]. It applies a declarative formalism for expressing alignments between RDF graphs. In [2], a subset of EDOAL expressions are transformed into a set of rewriting rules. The expressions involving the restrictions on concepts and properties and the restrictions on property occurrences and values are not featured in the rewriting rules. Makris *et al.* [9, 8] present the SPARQL-RW rewriting framework that applies a set of predefined rules for (complex) correspondences. They define a set of correspondence types on which the rewriting process is based (i.e., *Class Expression*, *Object Property Expression*, *Datatype Property*, and *Individual*). Zheng *et al.* [14] propose a rewriting algorithm that serves the purpose of context (i.e, units of measure) interchange for interoperability. Finally, Gillet *et al.* [7] propose an approach for rewriting query patterns that describe query families, using complex alignments. In this paper, we propose a set of rules for automatically rewriting SPARQL queries based on complex alignments. Differently from [4], our approach rewrites SPARQL queries instead of writing them from a complex alignment. Unlike [2], the proposed mechanism can handle restrictions on concepts and relations. In comparison with the [9] approach, EDOAL is an alternative to represent alignments in a more expressive (thus complete) way. For

instance, we propose occurrence and property datatype restrictions translation rules. However, our approach is limited to (1:n) complex alignments and does not handle initial SPARQL queries containing filters, unions, or other SPARQL options. The approach is based on the assumption that the queries to be transformed aim at retrieving new instances to meet a certain need. This is why only $\mathcal{T}box$ elements are taken into account. [14] focuses on context correspondences while our approach intends to translate all $\mathcal{T}box$ elements of a query. Finally, the proposal of [7] rewrites query patterns while we are interested in rewriting SPARQL queries, in a different level of abstraction. Although our approach relies on complex correspondences, their generation is out of the scope of this paper. The reader can refer to [13, 11] on the generation of complex correspondences based on patterns, linguistic approaches [12], or query mining [10].

4 SPARQL queries reformulation approach

Our approach focuses on the reformulation of a subset of SELECT SPARQL queries. We consider initial queries, which are to be rewritten, of the type:

$$Q_{\mathcal{O}} \; = \; SELECT \; DISTINCT? \; (\; Var+ \; | \; '*' \;) \; WHERE \; \{ \; T_{Q_{\mathcal{O}}} \; \}$$

where Var corresponds to the set of variables used as projection attributes and T_{Q_O} stands for the query pattern made of triples expressed using the source ontology $\mathcal{O}$. A triple t of T_{Q_O} is composed of a subject s, a predicate p and an object o. $\forall t \in T_{Q_{\mathcal{O}}}, \quad t = \langle s, p, o \rangle$. We only consider triples where s is a variable.

The purpose of our approach is to produce the set $T_{Q_{O'}}$ that contains the triples expressed according to entities of the ontology O', from T_{Q_O}, by using the complex alignment $A_{\mathcal{O} \rightarrow \mathcal{O}'}$. The approach is limited to complex correspondences (1:n) establishing an equivalence relation between entities of same nature. Such correspondences involve EDOAL expressions, as follows :

$$\langle ClassID, ClassID | ClassConstruction | ClassConstruction, \equiv \rangle$$
$$\langle RelationID, RelationID | RelationConstruction | RelationRestriction, \equiv \rangle$$
$$\langle PropertyID, PropertyID | PropertyConstruction | PropertyRestriction, \equiv \rangle$$

We also make the assumption that the alignment is complete and covers all the correspondences required to transform the entities of $T_{Q_{\mathcal{O}}}$. We define rules that take the set of triples $T_{Q_{\mathcal{O}}}$ as input and generate a SPARQL query. Three types of triples in $T_{Q_{\mathcal{O}}}$ are considered : *Class Object Triples*, *Predicate Triples* and *Other Triples*.

Algorithm 1 depicts the SPARQL query rewriting process. The **rewriteClassObject** and **rewritePredicate** functions apply the rules described in the following sections. These functions are recursive and can call each other. If a triple is not a *Class Object Triples* or a *Predicate Triples*, it means that its subject s is a variable, its predicate is an object property or a data property for which no correspondence is needed and its object is either a literal or a variable. This kind of triple does not need any transformation and is directly added to the final query. An example of such triple is `?s rdfs:label "a literal"`.

Algorithm 2 Rewriting mechanism process

```
new_query ← " "
for all triple t = ⟨s, p, o⟩ in query do
   if t is a Class Object Triple then
      new_query ← new_query + rewriteClassObject(s,p, o_{O'})
   else if t is a Predicate Triple then
      new_query ← new_query + rewritePredicate(s,p_{O'}, o)
   else
      new_query ← new_query + t
   end if
end for
return new_query
```

4.1 Class Object Triples

Class object triples, denoted $T_{Q_\mathcal{O}}^{Class}$, are structured as

$$\forall t \in T_{Q_\mathcal{O}}^{Class}, t = \langle s, p, o_\mathcal{O} \rangle \quad \text{, where} \begin{cases} s \text{ is a variable} \\ p \text{ is rdf:type} \\ o_\mathcal{O} \text{ is a } \textit{ClassID} \\ \exists < o_\mathcal{O}, o_{\mathcal{O}'}, \equiv > \ \in A_{\mathcal{O} \to \mathcal{O}'} \end{cases}$$

A class triple is identified if its object $o_\mathcal{O}$ is a *ClassID* and if there is a correspondence linking $o_\mathcal{O}$ to a class expression $o_{\mathcal{O}'}$ in $A_{\mathcal{O} \to \mathcal{O}'}$. In the transformation of a class triple t, its subject s and its predicate p remain the same. Only its object $o_\mathcal{O}$ is transformed according to its equivalent element $o_{\mathcal{O}'}$ in the alignment. The transformation rules of the **rewriteClassObject** function depend on the nature of the expression $o_{\mathcal{O}'}$, as follows:

1. *ClassID*: The expression $o_{\mathcal{O}'}$ is a*ClassID*. The transformed triple return by the function is: s p `IRI`$(o_{\mathcal{O}'})$.
2. *ClassConstruction*: $o_{\mathcal{O}'}$ is a class construction between two or more class expressions denoted by: $e^1_{\mathcal{O}'}$, $e^2_{\mathcal{O}'}$, $e^n_{\mathcal{O}'}$. The transformation rule depends on the construction operator.
   (a) AND: transforming an intersection consists in rewriting each triplet having as subject s, as predicate p and as object a distinct $e^i_{\mathcal{O}'}$ expression:
   rewriteClassObject$(s, p, e^1_{\mathcal{O}'})$ `+` **rewriteClassObject**$(s, p, e^2_{\mathcal{O}'})$ `+` `...` `+` **rewriteClassObject**$(s, p, e^n_{\mathcal{O}'})$
   (b) OR: transforming a union consists in using the SPARQL keyword "UNION" between the rewriting of each triplet having as subject s, as predicate p and as object a distinct $e^i_{\mathcal{O}'}$:[5]
   `{ +` **rewriteClassObject**$(s, p, e^1_{\mathcal{O}'})$ `+ } UNION { +` **rewriteClassObject**$(s, p, e^2_{\mathcal{O}'})$ `+ } +...+ UNION { +` **rewriteClassObject**$(s, p, e^n_{\mathcal{O}'})$ `+}`
   Table 1 shows an example of query rewriting based on the complex correspondence of Example 1, involving a *class construction with OR*.

[5]For sake of clarity and simplicity, we do not represent string delimiters.

Query for Cmt	Transformed query for Ekaw

SELECT ?z WHERE {
?z rdf:type cmt:Chairman.
}
	SELECT ?z WHERE {
	{?z rdf:type ekaw:Demo_Chair. }
	UNION {?z rdf:type ekaw:OC_Chair. }
	UNION {?z rdf:type ekaw:PC_Chair. }
	UNION {?z rdf:type ekaw:Session_Chair. }
	UNION {?z rdf:type ekaw:Tutorial_Chair. }
	UNION {?z rdf:type ekaw:Workshop_Chair. }}

Table 1. Transformation of a class triple based on the correspondence of Example 1 between a *classID* and a *class construction* using the OR rule.

(c) NOT: finding the negation of a class expression consists in finding the set of triples $\langle s, p, v\rangle$, where v is an intermediate variable, and from which the triples $\langle s, p, e^1_{\mathcal{O}'}\rangle$ are removed :

s p v . MINUS { + **rewriteClassObject**$(s, p, e^1_{\mathcal{O}'})$ + }

3. *ClassRestriction* : Restriction on class expressions takes into account relation or property expressions noted $relation(o_{\mathcal{O}'})$ or $property(o_{\mathcal{O}'})$. The transformation of the class triple depends on the nature of the restriction:
   (a) *TypeRestriction*: this restriction applies to a property expression stated in $o_{\mathcal{O}'}$ that limits the datatype of the property to a given *type*. The transformation rule consists in using an intermediate variable v that becomes the object of a new triple (that will keep on being rewritten according to the nature of $property(o_{\mathcal{O}'})$). The type restriction is applied to v with the use of a SPARQL `FILTER` and the `datatype(`v`)` function:
   rewritePredicate$(s, property(o_{\mathcal{O}'}), v)$ + FILTER (datatype(v) = *type*)
   (b) *DomainRestriction*: this restriction limits the range of a relation expression stated in $o_{\mathcal{O}'}$ to a class expression $range(o_{\mathcal{O}'})$ also stated in $o_{\mathcal{O}'}$. The **rewritePredicate** function is called with the relation $relation(o_{\mathcal{O}'})$ between the subject s and an intermediate variable v. The **rewriteClassObject** function is called to assert that v is an instance of the $range(o_{\mathcal{O}'})$ class expression : **rewritePredicate**$(s, relation(o_{\mathcal{O}'}), v)$ + **rewriteClassObject**$(v, \texttt{rdf:type}, range(o_{\mathcal{O}'}))$
   Table 2 presents a query transformation example based on the correspondence of Example 2 involving this kind of restriction.

Query for Ekaw	Transformed query for Cmt

SELECT ?z WHERE {
?z rdf:type ekaw:Accepted_Paper.
}
	SELECT ?z WHERE {
	?z rdf:type cmt:Paper.
	?z cmt:hasDecision ?var_temp.
	?var_temp rdf:type cmt:Acceptance. }

Table 2. Transformation of a class triple based on the correspondence on Example 2 between a *classID* and a *class expression* using the *DomainRestriction* rule.

(c) *ValueRestriction*: this restriction applies to a relation or property expression. The **rewritePredicate** function is called between the subject s, the $relation(o_{\mathcal{O}'})$ or $property(o_{\mathcal{O}'})$ and an intermediate variable v.

To restrain the values that can be taken by v, a SPARQL "FILTER" is used to compare v to a *value* given in the class expression. In the actual implementation, the stated value *value* can only be a literal or an instance. The comparator *cp* used in the SPARQL `FILTER` is one of the comparators provided by the EDOAL syntax : "=", ">" and "<". **rewritePredicate**$(s, relation/property(o_{\mathcal{O}'}), v)$ `+ FILTER (`v *cp value*`)` where $cp \in \{=, <, >\}$. For a "=" comparator, the resulting query is not optimal in terms of performance. The rewriting rule exception could be : **rewritePredicate**$(s, relation/property(o_{\mathcal{O}'}), value)$ instead of using an intermediate variable and a `FILTER` that applies to it. Table 3 presents a transformation example based on the correspondence of Example 4.

Query for Ekaw	Transformed query for ConfOf

`SELECT ?z WHERE {`
`?z rdf:type ekaw:Early-Registered_Participant.`
`}`
	`SELECT ?z WHERE {`
	`?z rdf:type confOf:Participant.`
	`?z confOf:earlyRegistration ?var_temp.`
	`FILTER(?var_temp="true"^^ xsd:boolean).}`

Table 3. Transformation of a triple using the correspondence of Example 4 between a class ID and a class expression using the *ValueRestriction* rule.

(d) *AttributeOccurrenceRestriction*: this restriction restrains the number of occurrences of a relation or a property expression. In order to count this number of occurrences, a SPARQL `SELECT` is imbricated to link the subject s to the count $count_v$ of an intermediate variable v. The value of $count_v$ is calculated thanks to the SPARQL `COUNT` function. The graph pattern in the imbricated `SELECT` is represented by the call of **rewritePredicate**$(s, relation/property(o_{\mathcal{O}'}), v)$. After the imbricated `SELECT`, a `FILTER` limits the value of $count_v$ to the restriction value val_{rest} with the comparator *cp* (both stated in the class restriction).

`{{SELECT` s `(COUNT(`v`) AS` $count_v$`) WHERE`
`{ +` **rewritePredicate**$(s, relation/property(o_{\mathcal{O}'}), v)$ `+ }`
`GROUP BY` s`.}`
`FILTER (`$compte_v$ *cp* val_{rest}`)}` , where $cp \in \{=, <, >\}$

Here, the resulting query could be optimized for a relation or a property occurring at least once ($count > 0$) . Instead of having an imbricated `SELECT`, the rewriting rule could be: **rewritePredicate**$(s, relation/property(o_{\mathcal{O}'}), v)$ with v a temporary variable.

4.2 Predicate Triples

Predicate triples, denoted by $T_{Q_\mathcal{O}}^{Predicate}$ have the following structure :

$$\forall t \in T_{Q_\mathcal{O}}^{Predicate} t = \langle s, p_\mathcal{O}, o \rangle \quad \text{, where} \begin{cases} s \text{ is a variable} \\ p_\mathcal{O} = \text{ a RelationId or PropertyId} \\ o \text{ is a variable, an instance or a literal} \\ \exists < p_\mathcal{O}, p_{\mathcal{O}'}, \equiv > \in A_{\mathcal{O} \to \mathcal{O}'} \end{cases}$$

In predicate triples, $p_{\mathcal{O}}$ is either a *RelationId* or a *PropertyId* and $p_{\mathcal{O}'}$ is respectively a relation expression or a property expression. A relation triple is transformed according to the nature of the expression $p_{\mathcal{O}'}$.

1. *RelationId* or *PropertyId*: the following triple is added to $T_{\mathcal{Q}_{\mathcal{O}'}}$: s `IRI`$(p_{\mathcal{O}'})$ o.
2. *RelationConstruction* or *PropertyConstruction*: $p_{\mathcal{O}'}$ is a construction between relation or property expressions designated by $p^1_{\mathcal{O}'}$, $p^2_{\mathcal{O}'}$, $p^n_{\mathcal{O}'}$. The transformation of the relation triple depends on the operator of the construction.
   (a) AND: this construction can be between two or more expressions (relation expressions resp. property expressions). The **rewritePredicate** function is called as follows:
   rewritePredicate$(s, p^1_{\mathcal{O}'}, o)$ `+` **rewritePredicate**$(s, p^2_{\mathcal{O}'}, o)$ `+ ...` `+` **rewritePredicate**$(s, p^n_{\mathcal{O}'}, o)$
   (b) OR: this construction can be between two or more expressions (relation expressions resp. property expressions). A SPARQL `UNION` links the calls to **rewritePredicate**:
   `{ +` **rewritePredicate**$(s, p^1_{\mathcal{O}'}, o)$ `+} UNION { +` **rewritePredicate**$(s, p^2_{\mathcal{O}'}, o)$ `+ } + ... + UNION { +` **rewritePredicate**$(s, p^n_{\mathcal{O}'}, o)$ `+ }`
   (c) NOT: the negation of a relation is the subset of all relations minus this relation. To represent all relations an intermediate variable v is introduced. The negation will be done using a SPARQL `MINUS`:
   s v o `. MINUS { +` **rewritePredicate**$(s, p^1_{\mathcal{O}'}, o)$ `+ }`
   (d) COMPOSE: a relation composition is a relation chain. Intermediate variables v_1, v_2, etc. are introduced to complete the chain between the subject s and the object o. If the relation expression $p_{\mathcal{O}'}$ is a *RelationExpression*, all the expressions of the chain will be relation expressions. If $p_{\mathcal{O}'}$ is a *PropertyExpression*, all the expressions of the chain will be relation expressions except the last one that will be a property expression. We assume that a composition imbrication or the use of a negation inside a composition is a modeling problem in the alignment itself.
   rewritePredicate$(s, p^1_{\mathcal{O}'}, v_1)$ `+` **rewritePredicate**$(v_1, p^2_{\mathcal{O}'}, v_2)$ `+ ... +` **rewritePredicate**$(v_{n-1}, p^n_{\mathcal{O}'}, o)$
   (e) INVERSE : this construction only applies to a *RelationExpression*. Inverting a relation consists in switching its subject and its object in a triple. **rewritePredicate**$(o, p^1_{\mathcal{O}'}, s)$ Table 4 gives an example of a triple transformation based on the correspondence of Example 3.
   (f) REFLEXIVE : this construction only applies to a *RelationExpression*. This operator is used to specify that a relation links its subject s to itself.
   rewritePredicate $(s, p^1_{\mathcal{O}'}, s)$
   (g) SYMMETRIC : this construction only applies to a *RelationExpression*. This operator is used to specify that a relation is used both ways : it is the intersection of a relation and its inverse.
   rewritePredicate$(s, p^1_{\mathcal{O}'}, o)$ + **rewritePredicate**$(o, p^1_{\mathcal{O}'}, s)$
3. *RelationDomainRestriction* or *PropertyDomainRestriction*: these restrictions limit the domain of a relation or property to a class expression $domain(p^1_{\mathcal{O}'})$ stated in the relation or property expression.

rewriteClassObject$(s, \texttt{rdf:type}, domain(p_{\mathcal{O}'}))$

4. *RelationCoDomainRestriction*: this restriction restrains the range of a *RelationExpression* to a class expression $range(p_{\mathcal{O}'})$.
   rewriteClassObject$(o, \texttt{rdf:type}, range(p_{\mathcal{O}'}))$
5. *PropertyTypeRestriction*: this restriction limits the datatype of a property to a given type *type* in the property expression $_{\mathcal{O}'}$. A SPARQL `FILTER` with the `datatype(`o`)` function is used. `FILTER (datatype(`o`) =` *type*`)`

Query for Ekaw	Transformed query for Cmt
SELECT ?z WHERE {	SELECT ?z WHERE{
?paper :writtenBy ?author. }	?author `cmt:writePaper` ?paper. }

Table 4. Transformed of a triple using the correspondence of Example 3 between a relation ID and a relation construction with the INVERSE constructor.

5 Validation

As far as we know, there is no available data set consisting of two knowledge bases, a complex and complete alignment between two ontologies and corresponding SPARQL queries for both bases. In the context of the OAEI oa4qa[6], a data set involving simple alignments is available. In order to fill this gap, we have manually created two data sets, following the principle of the oa4qa task. The validation of our mechanism checks that the translated query retrieves the same results as the reference query. Although these data sets only contain a small number of queries, it serves as a basis for a first validation of our approach.

Knowledge bases and SPARQL queries. The first data set was built during a project aiming at collecting knowledge about plant taxonomy. To meet this need, the knowledge bases Agronomic Taxon[7] and DBpedia have been considered. The task consists of retrieving answers to the following queries:

- $qa1$: which are the taxa of type species ?
- $qa2$: which are the taxa having for higher taxonomic rank a family taxon ?
- $qa3$: which are the taxa of taxonomical rank kingdom ?
- $qa4$: which are the taxa of taxonomical rank order ?
- $qa5$: which are the taxa of taxonomical rank genus ?

In order to build this data set, reference SPARQL queries corresponding to the natural language description above have been written manually for each knowledge base. The same approach was followed to construct the second data set. It aims at interrogating a subset of the OAEI 2015 ontologies about conference organization[8]. Three ontologies of this data set were considered (Cmt, ConfOf and Ekaw). We have defined the SPARQL queries answering the following queries :

- $qb1$: which are the reviewers of accepted papers ? (Ekaw to Cmt)

[6] http://oaei.ontologymatching.org/2015/
[7] http://ontology.irstea.fr/AgronomicTaxon
[8] http://oaei.ontologymatching.org/2015/conference/index.html

– $qb2$: which are authors of long submissions ? (Ekaw to Cmt)
– $qb3$: which are the chairmen who have submitted a paper ? (Cmt to Ekaw)
– $qc1$: which are the early registered participants who authored a submitted paper ? (Ekaw to ConfOf)
– $qc2$: which are the late registered participants who wrote a poster ?(Ekaw to ConfOf)

The three ontologies were populated with instances meeting these needs. Simultaneously, the queries were transformed into SPARQL queries specifically written for each of the knowledge bases.

Complex alignments. 10 complex correspondences (and 1 simple) have been manually produced between Agronomic Taxon and DBpedia. 8 simple and 6 complex correspondences have been manually produced between the three ontologies of the Conference data set. The alignments are available online[9].

Discussion. Our validation is based on the manual comparison of the set of results returned from the automatically rewritten query with respect to the results of the reference query. Even though the reference query and the rewritten one differ in terms of syntax, they retrieve the same set of instances. For example, Table 5 shows the queries considered for the need $qc1$ described above. The initial SPARQL query for Ekaw was transformed by using the complex correspondences of Example 4 and the simple correspondence $ekaw : authorOf \equiv confOf : writes$. As stated above, although the generated query is not syntactically equivalent to the reference query for ConfOf, they retrieve the same set of instances. The whole set of rewritten queries is available online[10].

Initial query for Ekaw (a)	Reference query for ConfOf	Generated query for ConfOf

`SELECT ?person WHERE {`
`?person :authorOf ?paper.`
`?paper a :Paper.`
`?person rdf:type`
`:Early-Registered_Participant.}`
	`SELECT ?person WHERE{`
	`?person`
	`:earlyRegistration true.`
	`?person :writes ?papier.`
	`?papier a :Paper.}`
		`SELECT ?person WHERE {`
		`?person :writes ?paper.`
		`?paper a :Paper.`
		`?person rdf:type :Participant.`
		`?person :earlyRegistration`
		`?v_temp0.`
		`FILTER(?v_temp0 =`
		`"true"^^xsd:boolean).}`

Table 5. Transformation of an initial query in comparison to its reference.

6 Conclusion and perspectives

In this paper, we have presented an approach to rewrite SELECT SPARQL queries formulated for a particular ontology to interrogate a knowledge base based on a second ontology using (1:n) complex correspondences. The proposed approach has been validated on two data sets manually created. There are many improvements to make to this mechanism. Indeed, the approach is limited to

[9]https://www.irit.fr/recherches/MELODI/telechargements/alignements.zip
[10]https://www.irit.fr/recherches/MELODI/telechargements/requetesgenerees.zip

formatted queries composed of triples whose subject is a variable. Instance alignments are not considered yet. We do not consider as well (n:m) correspondences. Proposals on complex graph pattern recognition in SPARQL queries would be interesting to take into account in order to address that matter. Another point is that we do not distinguish the kind of relation of a correspondence (subsumption and equivalence). Moreover, some EDOAL syntax of concepts have not been implemented, such as functions on literal (string concatenation, arithmetic operations, etc. that could be used in particular for value restrictions). Finally, property value restrictions is another EDOAL expression that was not implemented because it is more likely to be found in (n:m) correspondences. We plan to address all these points in future work.

References

1. Correndo, G., Salvadores, M., Millard, I., Glaser, H., Shadbolt, N.: SPARQL Query Rewriting for Implementing Data Integration over Linked Data. In: 1st International Workshop on Data Semantics (DataSem 2010) (March 2010)
2. Correndo, G., Shadbolt, N.: Translating expressive ontology mappings into rewriting rules to implement query rewriting. In: 6th Workshop on Ontology Matching (2011)
3. David, J., Euzenat, J., Scharffe, F., Trojahn, C.: The Alignment API 4.0. Semantic Web 2(1), 3–10 (2011)
4. Euzenat, J., Polleres, A., Scharffe, F.: Processing ontology alignments with sparql. In: International Conference on Complex, Intelligent and Software Intensive Systems. pp. 913–917 (2008)
5. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer-Verlag, Berlin, Heidelberg (2007)
6. Euzenat, J., Scharffe, F., Zimmermann, A.: Expressive alignment language and implementation. Tech. rep., INRIA (2007), `http://hal.inria.fr/hal-00822892/`
7. Gillet, P., Trojahn, C., Haemmerlé, O., Pradel, C.: Complex correspondences for query patterns rewriting. In: 8th Workshop on Ontology Matching (2013)
8. Makris, K., Bikakis, N., Gioldasis, N., Christodoulakis, S.: SPARQL-RW: transparent query access over mapped RDF data sources. In: 15th International Conference on Extending Database Technology. pp. 610–613. ACM (2012)
9. Makris, K., Gioldasis, N., Bikakis, N., Christodoulakis, S.: Ontology Mapping and SPARQL Rewriting for Querying Federated RDF Data Sources. In: OTM Confederated International Conferences. pp. 1108–1117 (2010)
10. Qin, H., Dou, D., LePendu, P.: Discovering executable semantic mappings between ontologies. In: OTM International Conference. pp. 832–849 (2007)
11. Ritze, D., Meilicke, C., Sváb-Zamazal, O., Stuckenschmidt, H.: A pattern-based ontology matching approach for detecting complex correspondences. In: 4th Workshop on Ontology Matching (2009)
12. Ritze, D., Völker, J., Meilicke, C., Sváb-Zamazal, O.: Linguistic analysis for complex ontology matching. In: 5th Workshop on Ontology Matching (2010)
13. Scharffe, F., Fensel, D.: Correspondence patterns for ontology alignment. In: Knowledge Engineering: Practice and Patterns, pp. 83–92. Springer (2008)
14. Zheng, X., Madnick, S.E., Li, X.: SPARQL Query Mediation over RDF Data Sources with Disparate Contexts. In: WWW Workshop on Linked Data on the Web (2012)

Identifying and Validating Ontology Mappings by Formal Concept Analysis

Mengyi Zhao[1] and Songmao Zhang[2]

[1,2]Institute of Mathematics, Academy of Mathematics and Systems Science,
Chinese Academy of Sciences, Beijing, P. R. China
[1]myzhao@amss.ac.cn, [2]smzhang@math.ac.cn

Abstract. As a well developed mathematical model for analyzing individuals and structuring concepts, Formal Concept Analysis (FCA) has been applied to ontology matching (OM) tasks since the beginning of OM research, whereas ontological knowledge exploited in FCA-based methods is limited. The study in this paper aims to empowering FCA with as much as ontological knowledge as possible for identifying and validating mappings across ontologies. Our method, called FCA-Map, constructs three types of formal contexts and extracts mappings from the lattices derived. Firstly, the token-based formal context describes how class names, labels and synonyms share lexical tokens, leading to lexical mappings (anchors) across ontologies. Secondly, the relation-based formal context describes how classes are in taxonomic, partonomic and disjoint relationships with the anchors, leading to positive and negative structural evidence for validating the lexical matching. Lastly, after incoherence repair, the positive relation-based context can be used to discover additional structural mappings. Evaluation on anatomy track and large biomedical ontologies track of the 2015 Ontology Alignment Evaluation Initiative (OAEI) campaign demonstrates the effectiveness of FCA-Map and its competitiveness with 2015 OAEI top-ranked OM systems.
Keywords: ontology matching, Formal Concept Analysis, concept lattice.

1 Introduction

In the Semantic Web, ontologies model domain conceptualizations so that applications built upon them can interoperate with each other by sharing the same meanings. Such knowledge sharing and reuse can be severely hindered by the fact that ontologies for the same domain are often developed for various purposes, differing in coverage, granularity, naming, structure and many other aspects. Ontology matching (OM) techniques aim to alleviate the heterogeneity by identifying correspondences across ontologies. Ontology matching can be performed at the element level and the structure level [4]. The former considers ontology classes and their instances independently, such as string-based and language-based techniques, whereas the latter exploits relations among entities, including graph-based and taxonomy-based techniques. Most ontology matching systems [2,3,5,9,11] adopt both element and structure level techniques to achieve better performance.

Among the first batch of OM algorithms and tools proposed in the early 2000s, FCA-Merge [13] distinguished in using Formal Concept Analysis (FCA) formalism to

derive mappings from classes sharing textual documents as their individuals. Proposed by Rudolf Wille [14], FCA is a well developed mathematical model for analyzing individuals and structuring concepts. FCA starts with a formal context consisting of a set of objects, a set of attributes, and their binary relations. Concept lattice, or Galois lattice, can be computed based on formal context, where each node represents a formal concept composed of a subset of objects (extent) with their common attributes (intent). The extent and the intent of a formal concept uniquely determine each other in the lattice. Moreover, the lattice represents a concept hierarchy where one formal concept becomes sub-concept of the other if its objects are contained in the latter. FCA can naturally be applied to ontology construction [12], and is also widely used in data analysis, information retrieval, and knowledge discovery.

Following the steps of FCA-Merge, several OM systems continued to use FCA as well as its alternative formalisms, exploiting different entities as the sets of objects and attributes for constructing formal contexts [1, 8, 15]. FCA-OntMerge [8], for example, utilizes the classes of ontologies and their attributes to form its formal context, whereas in [1] the formal context is composed of ontology classes as objects and terms of a domain-specific thesaurus as attributes. Different types of formal contexts decide the information used for ontology matching, and we observed that some intrinsic and essential knowledge of ontology has not been involved yet, including both textual information within classes (e.g., class names, labels, and synonyms) and relationships among classes (e.g., ISA, sibling, and disjointedness relations).

This motivated the study in this paper, i.e., empowering FCA with as much as ontological information as possible for identifying and validating mappings across ontologies. Our method, called FCA-Map, generates three types of formal contexts and extracts mappings from the lattices derived. Firstly, the token-based formal context describes how class names, labels and synonyms share lexical tokens, leading to lexical mappings (anchors) across ontologies. Secondly, the relation-based formal context describes how classes are in taxonomic, partonomic and disjoint relationships with the anchors, leading to positive and negative structural evidence for validating the lexical matching. Lastly, after incoherence repair, the positive relation-based context can be used to discover additional structural mappings. Evaluation on anatomy track and large biomedical ontologies track of the 2015 Ontology Alignment Evaluation Initiative (OAEI) campaign demonstrates the effectiveness of FCA-Map and its competitiveness with 2015 OAEI top-ranked OM systems.

2 Preliminaries

Formal Concept Analysis (FCA) is a mathematical theory of data analysis using formal contexts and concept lattices. Formal context is defined as a triple $\mathbb{K} := (G, M, I)$, where G is a set of objects, M a set of attributes, and I a binary relation between G and M in which gIm holds, i.e., $(g, m) \in I$, reads: object g has attribute m [6]. Formal contexts are often illustrated in binary tables, as exemplified by Table 1, where rows correspond to objects, columns to attributes, and a cell is marked with "$\times$" if the object in its row has the attribute in its column.

Definition 1. *[6] For subsets of objects and attributes $A \in G$ and $B \in M$, derivation operators are defined as follows:*

$$A' = \{m \in M \mid gIm \; for \; all \; g \in A\}$$
$$B' = \{g \in G \mid gIm \; for \; all \; m \in B\}$$

A' denotes the set of attributes common to the objects in A; B' denotes the set of objects which have all the attributes in B.

A formal concept of context $\mathbb{K}$ is a pair (A, B) consisting of extent $A \in G$ and intent $B \in M$ such that $A = B'$ and $B = A'$. $\mathfrak{B}(\mathbb{K})$ denotes the set of all formal concepts of context $\mathbb{K}$. The partial order relation, namely subconcept-superconcept-relation, is defined as:

$$(A_1, B_1) \leq (A_2, B_2) :\Leftrightarrow A_1 \subseteq A_2 (\Leftrightarrow B_1 \supseteq B_2)$$

Relation $\leq$ is called a hierarchical order of formal concepts. $\mathfrak{B}(\mathbb{K})$ ordered in this way is exactly a complete lattice, called the concept lattice and denoted by $\underline{\mathfrak{B}}(\mathbb{K})$.

	vertebrate	mammal	flying	aquatic	carnivorous
elephant	×	×			
dolphin	×	×		×	×
porpoise	×	×		×	×
hawk	×		×		×
octopus				×	×

Table 1: An example formal context $\mathbb{K}_e$.

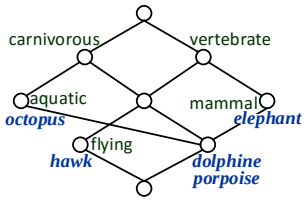


Fig. 1: Concept lattice $\underline{\mathfrak{B}}(\mathbb{K}_e)$ with simplified labelling.

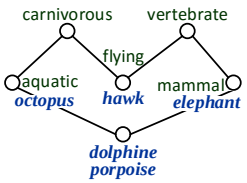


Fig. 2: GSH of concept lattice $\underline{\mathfrak{B}}(\mathbb{K}_e)$.

For an object $g \in G$, its *object concept* $\gamma g := (\{g\}'', \{g\}')$ is the smallest concept in $\underline{\mathfrak{B}}(\mathbb{K})$ whose extent contains g. In other words, object g can generate formal concept γg. Symmetrically, for an attribute $m \in M$, its *attribute concept* $\mu m := (\{m\}', \{m\}'')$ is the greatest concept in $\underline{\mathfrak{B}}(\mathbb{K})$ whose intent contains m. In other words, object m can generate formal concept μm. For a formal concept (A, B), its *simplified extent* (*simplified intent*), denoted by K_{ex} (K_{in}), is a minimal description of the concept. Each object (attribute) in K_{ex} (K_{in}) can generate the formal concept (A, B). As a matter of fact, K_{ex} dose not appear in any descendant of (A, B) and K_{in} dose not appear in any ancestor of (A, B). Figure 1 shows the concept lattice of context $\mathbb{K}_e$ in Table 1, where each formal concept is labeled by its simplified extent and intent.

Galois Sub-hierarchy (GSH) introduced by [7] is a sub-structure of concept lattice. Only concepts carrying information are retained in GSH, meaning that GSH solely contains formal concepts that introduce new objects or new attributes and excludes formal concepts whose K_{ex} and K_{in} are both empty. The ordering of formal concepts in GSH is the same as in the original concept lattice. Removing the formal concepts without labels in Figure 1 leads to the GSH shown in Figure 2.

3 The FCA-Map Method

Given two ontologies, FCA-Map builds formal contexts and uses the derived concept lattices to cluster the commonalities among ontology classes, at lexical level and structural level, respectively. Concretely, FCA-Map performs step-by-step as follows.

1. **Acquiring anchors lexically.** The token-based formal context is constructed, and from its derived concept lattice, a group of lexical anchors $\mathcal{A}$ across ontologies can be extracted.
2. **Validating anchors structurally.** Based on $\mathcal{A}$, the relation-based formal context is constructed, and from its derived concept lattice, positive and negative structural evidence of anchors can be extracted. Moreover, an enhanced alignment $\mathcal{A}'$ without incoherences among anchors is obtained.
3. **Discovering additional matches.** Based on $\mathcal{A}'$, the positive relation-based formal context is constructed, and from its derived concept lattice, additional matches across ontologies can be identified.

We take two anatomical ontologies, Adult Mouse Anatomy[1] (MA) and the anatomy subset of National Cancer Institute Thesaurus[2] (NCI), to demonstrate our method. MA is a structured controlled vocabulary describing the anatomical structure of the adult mouse, whereas NCI describes the human anatomy for the purpose of cancer research. The versions used are the OWL files of these two ontologies provided by the 2015 OAEI. For MA and NCI, the token-based and relation-based formal contexts are of large-size, resulting in complex structures of the concept lattices derived. In order to avoid generating redundant information, GSH, a polynomial-sized representation of concept lattice that preserves the most pertinent information, is utilized in FCA-Map.

3.1 Constructing the token-based formal context to acquire lexical anchors

Most OM systems rely on lexical matching as initiation due to the fact that classes sharing names across ontologies quite likely represent the same entity in the domain of interest. FCA-Map, rather than using lexical and linguistic analysis, generates a formal context at the lexical level and obtains mappings from the lattice derived from the context.

The token-based formal context $\mathbb{K}_{lex} := (G_{lex}, M_{lex}, I_{lex})$ is described as follows. Names of ontology classes as well as their labels and synonyms, when available, are exploited after normalization that includes inflection, tokenization, stop word elimination[3], and punctuation elimination. In $\mathbb{K}_{lex}$, G_{lex} is the set of strings each corresponding to a name, label, or synonym of classes in two ontologies, M_{lex} is the set of tokens in these strings, and binary relation $(g, m) \in I_{lex}$ holds when string g contains token m,

[1] http://www.informatics.jax.org/glossary/adult_ma_dictionary

[2] https://ncit.nci.nih.gov/ncitbrowser/

[3] Although eliminating the stop words carrying logical meanings may affect the precision, its benefit in recall is more advantageous according to our experiments.

or a synonym[4] or lexical variation[5] of m. Table 2 shows $\mathbb{K}_{lex}$ of a small part of MA and NCI, and its derived concept lattice in GSH form is displayed in Figure 3. For each formal concept derived, in addition to strings in its extent, we are also interested in the classes that these strings come from, called *class-origin extent*. For example, in Figure 3, the *class-origin extent* of formal concept by node 7 is {MA:*mammary gland fluid/secretion*, NCI:*Breast Fluid or Secretion*} since in NCI, "*Mammary Gland Fluids and Secretions*" is a synonym of class NCI:*Breast Fluid or Secretion*.

	gland	adrenal	zona	zone	fasciculata	reticularis	salivary	palatine	mammary	secretion	fluid
MA:*palatine gland*	×							×			
MA:*adrenal gland zona fasciculata*	×	×	×		×						
MA:*adrenal gland zona reticularis*	×	×	×			×					
MA:*mammary gland fluid/secretion*	×								×	×	×
NCI:*Palatine Salivary Gland*	×						×	×			
NCI:*Fasciculata Zone*				×	×						
NCI:*Reticularis Zone*				×		×					
NCI:*Mammary Gland Fluids and Secretions*	×								×	×	×

Table 2: Token-based formal context $\mathbb{K}_{lex}$ of a small part of MA and NCI.

An essential property of FCA is the duality between a set of objects and their attributes. The more attributes demanded, the fewer objects can meet the requirements. In the case of the token-based formal concept, the more common tokens appearing in its intent, the fewer strings the extent contains, and the more possibly for the classes in *class-origin extent* to be matched. This is to say that cardinality of the extent can reflect how similar the strings are, thus classes from different source ontologies in a smaller-sized *class-origin extent* can be considered as a mapping with higher confidence. Practically, we restrict our attention to formal concepts whose *simplified extent* or *class-origin extent* contains exactly two strings or classes across ontologies, and extract two types of lexical anchors, namely ***Type I anchor*** for the exact match, and ***Type II anchor*** for the partial match, respectively. Of note, on the other hand, cardinality of the intent cannot be used to measure the similarity of strings. For example, MA:*nerve* and NCI:*Nerve*, which is a match, only share one token, whereas MA:*left lung respiratory bronchiole* and NCI:*Right Lung Respiratory Bronchiole*, not a match, share three tokens.

Type I anchor. *Simplified extent* K_{ex} of the formal concept contains exactly two strings from classes across ontologies. This indicates that the two strings are composed of the same or synonymous tokens, thus the corresponding classes are extracted to be a match, as exemplified by $(MA$: *mammary gland fluid/secretion*, NCI : *Breast Fluid or Secretion*$)$ through formal concept of node 7 in Figure 3 whose K_{ex} has two strings, one from MA and the other NCI.

Type II anchor. The *class-origin extent* of the formal concept contains exactly two classes across ontologies and *simplified extent* K_{ex} contains strings from at most

[4] Sub-Term Mapping Tools (https://lsg2.nlm.nih.gov/LexSysGroup/Projects/stmt/2013+/web/index.html) are used to access synonyms.

[5] SPECIALIST Lexicon (https://lexsrv3.nlm.nih.gov/LexSysGroup/Projects/lexicon/current/web/index.html) of UMLS is used to access lexical variations.

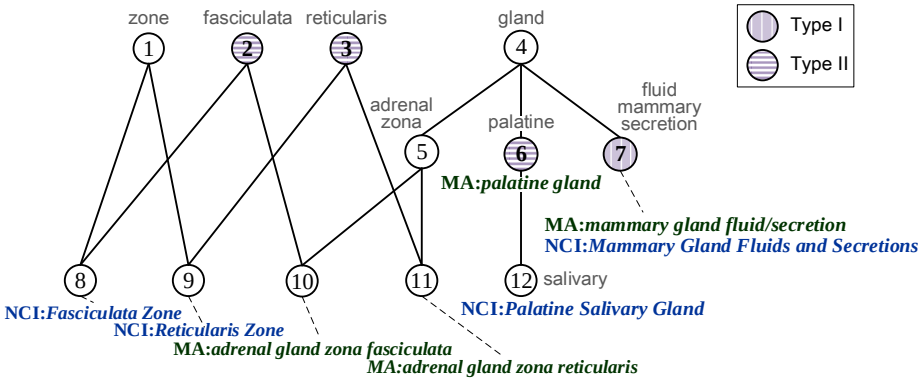


Fig. 3: Concept lattice in GSH with simplified labeling derived from $\mathbb{K}_{lex}$ in Table 2.

one source ontology. Here the strings share tokens in the intent rather than composed of the same or synonymous tokens. For example, (MA:*adrenal gland zona fasciculata*, NCI:*Fasciculata Zone*) is extracted from node 2 in Figure 3, due to the common token "fasciculata" which exists solely in these two classes. And (MA:*palatine gland*, NCI:*Palatine Salivary Gland*) is identified as an anchor from node 6, due to the common tokens "palatine" and "gland" which co-exist solely in these two classes.

3.2 Constructing the relation-based formal context to validate lexical anchors

Structural relationships of ontologies are exploited to validate the matches obtained at the lexical level. One of our previous studies [16] proposed using positive and negative structural evidence among anchors for the purpose of validation. More precisely, classes of one anchor sharing relationships to classes in another anchor can be seen as their respective positive evidence. On the other hand, negative structural evidence refers to the conflict based on the disjointedness relationships between classes. In FCA-Map, we build the relation-based formal context to obtain both positive and negative structural evidence for lexical anchors. Both explicitly represented and inferred semantic relations are used in our method.

	(ISA) (MA:*ligament*, NCI:*Ligament*)	(I-D) (MA:*organ system*, NCI:*Organ System*)	(SIB) (MA:*adipose tissue*, NCI:*Adipose Tissue*)	(SIB) (MA:*larynx ligament*, NCI:*Laryngeal Ligament*)	(PAT) (MA:*larynx*, NCI:*Larynx*)
MA:*ligament*		×	×		
MA:*periodontal ligament*	×	×		×	
MA:*auricular ligament*	×	×		×	
MA:*adipose tissue*		×			
MA:*larynx ligament*	×	×			×
NCI:*Ligament*		×			
NCI:*Periodontium*	×	×		×	
NCI:*Broad Ligament*	×	×		×	
NCI:*Adipose Tissue*		×			
NCI:*Laryngeal Ligament*	×	×			×

Table 3: Relation-based formal context $\mathbb{K}_{rel}$ of a small part of MA and NCI.

The relation-based formal context $\mathbb{K}_{rel} := (G_{rel}, M_{rel}, I_{rel})$ is described as follows. Classes in two source ontologies are taken as object set G_{rel}, and lexical anchors prefixed with different relational labels are taken as attribute set M_{rel}. In the case of MA and NCI, four kinds of relationships are considered, *ISA*, *SIBLING-WITH*, *PART-OF*, and *DISJOINT-WITH*, labeled by "(ISA)", "(SIB)", "(PAT)", and "(I-D)" (or "(D-I)"), respectively. Binary relation $(g, m) \in I_{rel}$ holds if g has the corresponding relationship (as in the prefix of m) with the class from the same source ontology as g in the anchor of m. The relation-based formal context $\mathbb{K}_{rel}$ of a small part of MA and NCI is displayed in Table 3. For instance, MA:*periodontal ligament* and NCI:*Periodontium* are subclasses of MA:*ligament* and NCI:*Ligament*, respectively, thus (MA:*periodontal ligament*, (ISA)(MA:*ligament*, NCI:*Ligament*)) $\in I_{rel}$ and (NCI:*Periodontium*, (ISA)(MA: *ligament*, NCI:*Ligament*)) $\in I_{rel}$ hold. Moreover, MA:*adipose tissue* is a subclass of MA:*organ system* whereas NCI:*Adipose Tissue* is disjoint with NCI:*Organ System*, thus (MA:*adipose tissue*, (I-D)(MA:*organ system*, NCI:*Organ system*)) $\in I_{rel}$ and (NCI:*Adipose Tissue*, (I-D)(MA:*organ system*, NCI:*Organ system*)) $\in I_{rel}$ hold.

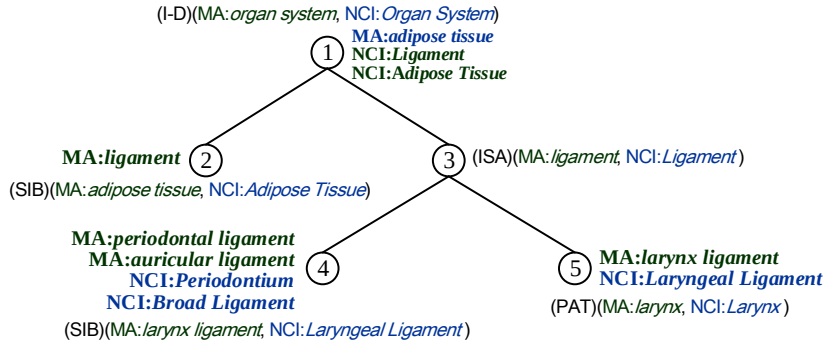


Fig. 4: GSH of $\mathbb{K}_{rel}$ with simplified labeling.

The derived concept lattice in GSH form of $\mathbb{K}_{rel}$ of a small part of MA and NCI is illustrated in Figure 4. Formal concepts whose extents include both classes in some anchors indicate structural evidence. Such anchors are positive evidence to anchors with label"(ISA)", "(SIB)" or "(PAT)" in the intent, and vice versa. Conversely, they are negative evidence to anchors with label "(I-D)" or "(D-I)" in the intent, and vice versa. In this way, positive and negative structural evidence set of each anchor a can be obtained, denoted by $P(a)$ and $N(a)$, respectively. For example, in the extent of node 3 in Figure 4, (MA:*periodontal ligament*, NCI:*Periodontium*) and (MA:*larynx ligament*, NCI:*Laryngeal Ligament*), two anchors acquired lexically, are positive evidences to anchor (MA:*ligament*, NCI:*Ligament*) with label "(ISA)" in the intent, and negative evidences to anchor (MA:*organ system*, NCI:*Organ System*) with label "(I-D)". The *support degree* and *incoherence degree* of each anchor are the cardinality of its positive and negative evidence set, respectively.

Now we can utilize all the positive evidence sets $\mathcal{P}$ and negative evidence sets $\mathcal{N}$ to eliminate incorrect lexical anchors and retain the correct ones. There are two steps conducted one-by-one as follows.

Incoherence repairing. The negative evidence leads to incoherency among anchors, for which FCA-Map repairs in a greedy way, i.e., eliminating the incoherence-causing

anchors iteratively until $\mathcal{N}$ becomes empty. At each iteration, anchor a having the least negative evidence set, i.e., the smallest *incoherence degree*, is selected. For every anchor a' in $N(a)$, if *incoherence degree* of a' is greater than a, eliminate a'; otherwise, compare the *support degree* of a and a', and eliminate the one with smaller *support degree*.

Anchor screening. Anchors having no positive structural evidence according to the updated $\mathcal{P}$ are either caused by the structural isolatedness of classes, or simply incorrect mismatches. FCA-Map screens anchors based on both lexical and structural evidence, where ***Type II anchors*** without positive evidence are eliminated.

3.3 Constructing the positive relation-based formal context to discover additional matches

After incoherence repair and screening, anchors retained are those supported both lexically and structurally. Based on the enhanced alignment, FCA-Map goes further to build the positive relation-based formal context aiming to identify new, structural mappings. The way positive relation-based formal context $\mathbb{K}'_{rel}$ constructed is similar to $\mathbb{K}_{rel}$, i,e., using classes in two source ontologies as object set and anchors prefixed with relationship labels as attribute set. In the case of MA and NCI, five kinds of relationships are considered, *ISA*, *SUPERCLASS-OF*, *SIBLING-WITH*, *PART-OF*, and *HAS-PART*, where disjointedness relationship is no longer necessary. For the derived formal concepts, we restrict our attention to those with exactly two classes across ontologies in the *simplified extent*. Although most of the mappings extracted this way have already been identified at the lexical level, new additional matches emerge, as exemplified by (MA: *hindlimb bone*, NCI: *Bone of the Lower Extremity*).

4 Evaluation

To demonstrate the effectiveness of FCA-Map, evaluation is performed on two pairs of real-world ontologies, Adult Mouse Anatomy (2,744 classes) and the anatomy subset of NCI Thesaurus (3,304 classes); and the Foundational Model of Anatomy (3,696 classes) and NCI (6,488 classes), respectively, from anatomy track and large biomedical ontologies track of OAEI 2015. FCAlib[6] is used to derive concept lattices (GSH) from formal contexts. It is an open-source, extensible library for FCA tool developers. FCA-Map is implemented in Java and the experiments were conducted in a PC with Intel i7 (3.60GHz) and 8GB RAM. It took 166 seconds and 425 seconds, respectively, for FCA-Map to finish the MA-NCI$_{Anat.}$ and the FMA-NCI matching.

4.1 Anchors obtained

The results of lexical matching by FCA-Map are summarized in Table 4, and structural matching is presented in Table 5 where the upper part is about structural validation and the lower part about extra discovered structural mappings. Columns "Corr.", "Incor.", and "Unkn." indicate the number of correct, incorrect, and unknown mappings, respectively, as categorized by OAEI where "unknown" mappings will neither be considered as correct nor incorrect when evaluating the alignment, but will simply be ignored.

	MA-NCI$_{Anat.}$				FMA-NCI				
Types of anchors	**Total**	Corr.	Incor.	P	**Total**	Corr.	Unkn.	Incor.	P
Type I	1,223	1,163	60	95.1%	2,759	2,416	248	95	96.2%
Type II	172	113	59	65.7%	131	60	4	67	47.2%
Total	1,395	1,276	119	91.5%	2,890	2,476	252	162	93.9%

Table 4: Results of lexical anchors.

	MA-NCI$_{Anat.}$				FMA-NCI				
Types of anchors	**Total**	Corr.	Incor.	P	**Total**	Corr.	Unkn.	Incor.	P
Type I	1,220	1,161	59	95.2%	2,703	2,414	208	81	96.8%
Type II	125	98	27	78.4%	63	46	2	15	75.4%
Total	1,345	1,259	86	93.6%	2,766	2,460	210	96	96.2%
Additional	16	10	6	62.5%	25	3	0	22	12%
Total	1,361	1,269	92	93.2%	2,791	2,463	210	118	95.4%

Table 5: Results of enhanced alignment.

One can see that most of the lexical anchors are of ***Type I***, i.e., the name, synonym or label of one class is the same as another class. For example, MA:*cortical layer II* and NCI:*External Granular Layer* are extracted as an anchor because in MA, "*external granular layer*" is a synonym of MA:*cortical layer II*. Incorrect ***Type I anchors*** mainly come from three cases. (1) Although having the same name, classes in anchor do not represent equivalent entity. For example, MA:*organ system* and NCI:*Organ System*, although sharing matched subclasses, have respective additional different subclasses. (2) Mismatched classes may be considered to be a mapping based on their synonyms or labels. For example, anchor (MA:*cerebellum lobule I*, NCI:*Lingula*) (through synonym "*lingula*" in MA) is a mismatch because the former is a part of cerebellar vermis and the latter a part of left lung. (3) Using external lexicon may introduce incorrect anchors. For example, MA:*back* matches NCI:*Dorsum* because "back" and "dorsum" are synonymous according to the lexicon used in FCA-Map. This is a mismatch because in MA back is a part of trunk, while in NCI dorsum refers to outer surface of scapula.

Type II lexical anchors have lower precisions, reflecting the unstable performance of relying on names sharing tokens to derive commonalities of classes. Nevertheless, many incorrect anchors can be eliminated in the validation process, causing the precision to increase, for instance from 47.2% to 75.4% for ***Type II*** anchors in FMA-NCI. Take ***Type II*** anchor (MA:*retina ganglion cell layer*, NCI: *Retinal Ganglion Cell*) for example. It is eliminated in incoherence repair because of its conflict with (MA:retina layer, NCI: Retina Layer), of which the *support degree* is 0 and 8, respectively. The structural validation based on the relation-based concept lattice in FCA-Map can ensure to improve the precision of lexical mappings.

4.2 Comparing with other lexical matching methods

Among many lexical matching methods such as string equality, substring test, and edit distance, TFIDF-based methods [4] are of particular interest because similarly to FCA-Map they are based on tokens. Adopted in OM systems YAM++ [3] and GMap [10],

[6] https://julianmendez.github.io/fcalib/

TFIDF measures simultaneously how often the tokens appear in one class name and how much information the tokens bring across names of classes from different ontologies. We compare the performance of lexical matching of FCA-Map with TFIDF solely using the class names of MA and NCI without any external resources. The result is shown in Figure 5, where F-measure of FCA-Map is higher than TFIDF for any threshold.

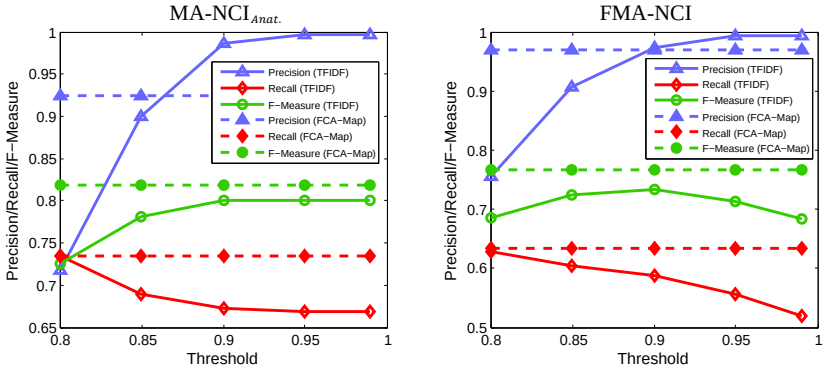


Fig. 5: Comparing with TFIDF.

Compared with the TFIDF-based methods, FCA-Map emphasizes on the particular commonality of two strings, and there is no need for setting thresholds which is required in TFIDF for selecting matches. This can be illustrated by MA: *tectum* and NCI: *tectum mesencephali*. They are not matched according to TFIDF because token "mesencephali" has a high inverse-document-frequency (it solely appears in this string) and token "tectum" is ignored (it solely appears in the two strings). On the other hand, this correspondence can be derived in our method since there is a formal concept with intent {"tectum"} and extent exactly containing these two strings. Moreover, our method can avoid the mistake of locally measuring frequency of tokens. For instance, MA: *common iliac artery* and NCI: *Right Common Iliac Artery* have a relatively high similarity (0.86) according to TFIDF, while this pair is not extracted by FCA-Map. There are many other class names share tokens "common", "iliac", and "artery", such as MA: *Left Common Iliac Artery* and NCI: *Right Common Iliac Artery Branch*, therefore what the two strings in comparison share are not unique enough for them to be chosen as a match. Indeed, our method features in detecting the particular commonality solely belongs to the names compared while ignoring the commonality shared by many other names.

4.3 Comparing with OAEI 2015 top-ranked systems

A comparison between FCA-Map and OAEI 2015 top-ranked systems is shown in Table 6. For MA-NCI$_{Anat.}$, the precision, recall and F-measure of FCA-Map ranks second, fifth, and forth, respectively. Results of FMA-NCI are encouraging, with both recall and F-measure tie for first. Moreover, FCA-Map is capable of extracting mappings that cannot be identified by other systems, as exemplified by ***Type II*** anchors (MA:*adrenal gland zona reticularis*, NCI:*Reticularis Zone*), (MA:*ileocaecal junction*, NCI:*Ileocecal Valve*). These mappings are identified in the token-based concept lattice and validated

in the relation-based concept lattice. The tokens shared by two classes in these mappings are unique to their names. The lexical matching method of FCA-Map is suitable for domain ontologies having class names, labels, or synonyms from domain-specific vocabulary, whereas its performance can be relatively poor for general-purpose ontologies whose terminologies are more varied and ambiguous, like those in the conference track of OAEI where FCA-Map ranked at the average level. Additionally, for negative evidence to be identified, our method requires that at least one source ontology declares disjointedness relationships between classes.

	MA-NCI$_{Anat.}$			FMA-NCI		
Systems	P	R	F	P	R	F
XMAP-BK	-	-	-	0.971	0.902	0.935
AML	0.956	0.931	0.944	0.960	0.899	0.928
LogMap	0.918	0.846	0.88	0.949	0.901	0.924
LogMapBio	0.882	0.901	0.891	0.926	0.917	0.921
XMAP	0.928	0.865	0.896	0.970	0.784	0.867
FCA-Map	0.932	0.837	0.882	0.954	0.917	0.935

Table 6: Comparing with OAEI 2015 top-ranked systems.

5 Discussion and Conclusions

Discovering complex mappings structurally. As shown in Table 5, structural mappings identified by the positive relation-based concept lattice are limited. Nevertheless, in the lattice we noticed that the *simplified extents* of some formal concepts contain more than two classes from different source ontologies, meaning these classes share the same structural relationships to anchors in the intent. Such classes may compose a complex mapping, as elaborated in the following.

1. *One-to-group mappings*. The *simplified extent* contains only one class from one source ontology and multiple classes from the other source ontology. For example, MA:*inferior suprarenal vein* can be mapped to the group of concepts {NCI:*Left Suprarenal Vein*, NCI:*Right Suprarenal Vein*} as the three concepts are contained within one *simplified extent* that has no more classes. This one-to-group mapping comes from the difference in granularity between MA and NCI.
2. *Group-to-group mappings*. The *simplified extent* contains multiple classes from different source ontologies, respectively. For example, two groups of concepts {MA:*sacral vertebra 1*, MA:*sacral vertebra 2*, MA:*sacral vertebra 3*, MA:*sacral vertebra 4*} and {NCI:*S1 Vertebra*, NCI:*S2 Vertebra*, NCI:*S3 Vertebra*, NCI:*S4 Vertebra*, NCI:*S5 Vertebra*} can be mapped as these classes are contained in one *simplified extent* that has no more classes. This group-to-group mapping represents the difference between mouse and human anatomy.

Compared with other FCA-based OM systems, the study in this paper is more comprehensive as an attempt to push the envelope of the Formal Concept Analysis formalism in ontology matching tasks. Three types of formal contexts are constructed one-by-one, and their derived concept lattices are used to cluster the commonalities among

classes at lexical and structural level, respectively. Experiments on large, real-world domain ontologies show promising results and reveal the power of FCA. Our future work would introduce more elements of ontology into FCA-Map including properties, individuals, and logical constructors and axioms. Optimization techniques for handling large-scale FCA contexts will also be worth exploring.

Acknowledgements. This work has been supported by the National Key Research and Development Program of China under grant 2016YFB1000902, the Natural Science Foundation of China under No. 61232015, the Knowledge Innovation Program of the Chinese Academy of Sciences (CAS), Key Lab of Management, Decision and Information Systems of CAS, and Institute of Computing Technology of CAS.

References

1. de Souza, K.X.S., Davis, J.: Aligning ontologies and evaluating concept similarities. In: OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer (2004) 1012–1029
2. Djeddi, W.E., Khadir, M.T.: Xmap: a novel structural approach for alignment of owl-full ontologies. In: Machine and Web Intelligence (ICMWI), 2010 International Conference on, IEEE (2010) 368–373
3. Duyhoa, N., Bellahsene, Z.: Yam++ results for oaei 2012. In: Seventh International Workshop on Ontology Matching. (2012) 226–233
4. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer Science & Business Media (2013)
5. Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I.F., Couto, F.M.: The agreementmakerlight ontology matching system. In: OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer (2013) 527–541
6. Ganter, B., Wille, R.: Formal concept analysis: mathematical foundations. Springer Science & Business Media (2012)
7. Godin, R., Mili, H.: Building and maintaining analysis-level class hierarchies using galois lattices. In: ACM SIGplan Notices. Volume 28., ACM (1993) 394–410
8. Guan-yu, L., Shu-peng, L., et al.: Formal concept analysis based ontology merging method. In: Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on. Volume 8., IEEE (2010) 279–282
9. Jiménez-Ruiz, E., Grau, B.C.: Logmap: Logic-based and scalable ontology matching. In: International Semantic Web Conference, Springer (2011) 273–288
10. Li, W.: Combining sum-product network and noisy-or model for ontology matching. Ontology Matching (2015) 35
11. Niepert, M., Meilicke, C., Stuckenschmidt, H.: A probabilistic-logical framework for ontology matching. In: AAAI, Citeseer (2010)
12. Obitko, M., Snsel, V., Smid, J.: Ontology design with formal concept analysis. CLA **128**(3) (2004) 1377–1390
13. Stumme, G., Maedche, A.: Fca-merge: Bottom-up merging of ontologies. In: IJCAI. Volume 1. (2001) 225–230
14. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Ordered sets. Springer (1982) 445–470
15. Xu, X., Wu, Y., Chen, J.: Fuzzy fca based ontology mapping. In: 2010 First International Conference on Networking and Distributed Computing, IEEE (2010) 181–185
16. Zhang, S., Bodenreider, O.: Experience in aligning anatomical ontologies. International journal on Semantic Web and information systems **3**(2) (2007) 1

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).

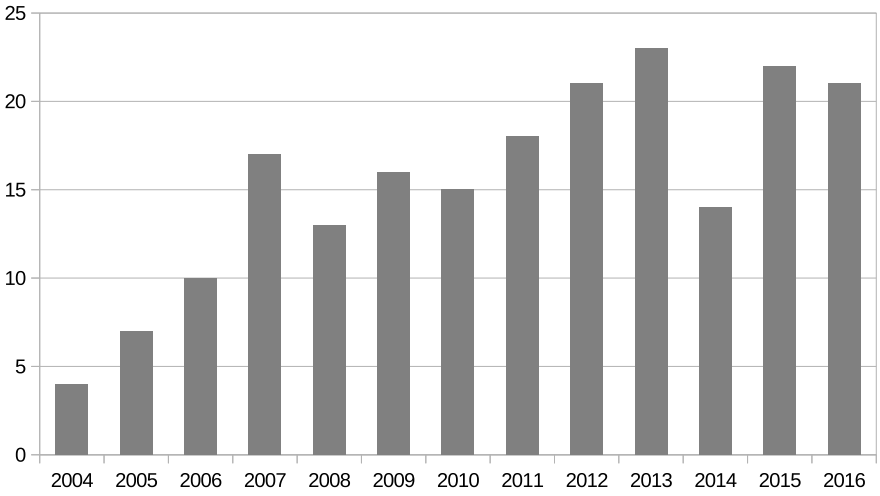


Fig. 1. Number of systems participating in OAEI per year.

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.

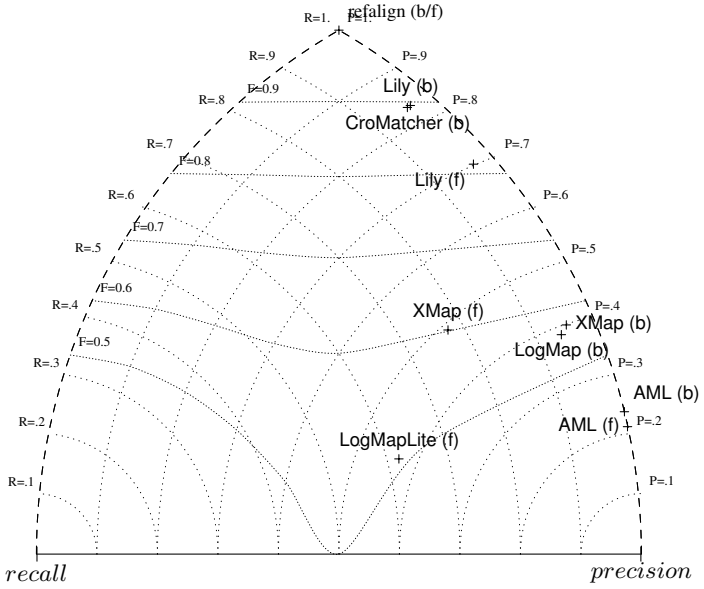


Fig. 2. Triangle view on the benchmark data sets (biblio=(b), film=(f), run 5, non present systems have too low F-measure, below .5).

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

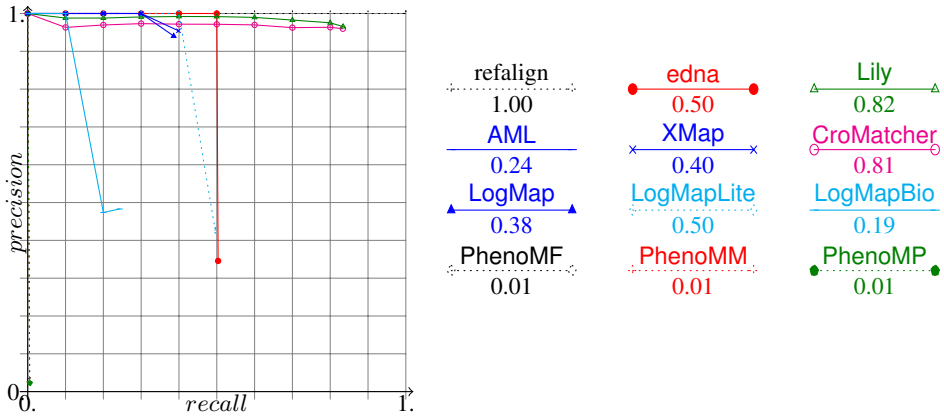


Fig. 3. Precision/recall plots on biblio.

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

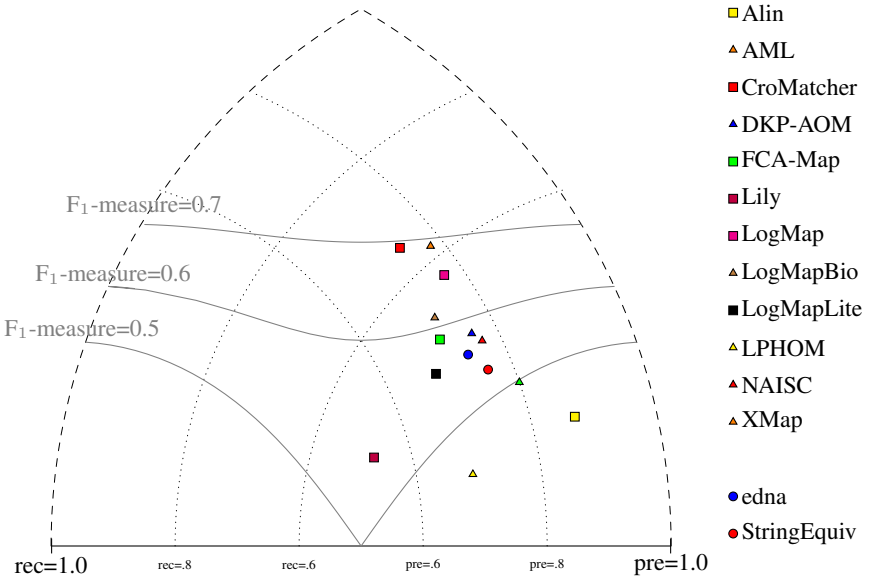


Fig. 4. Precision/recall triangular graph for the conference test case. Dotted lines depict level of precision/recall while values of F_1-measure are depicted by areas bordered by corresponding lines F_1-measure=0.[5|6|7].

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	194149	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.

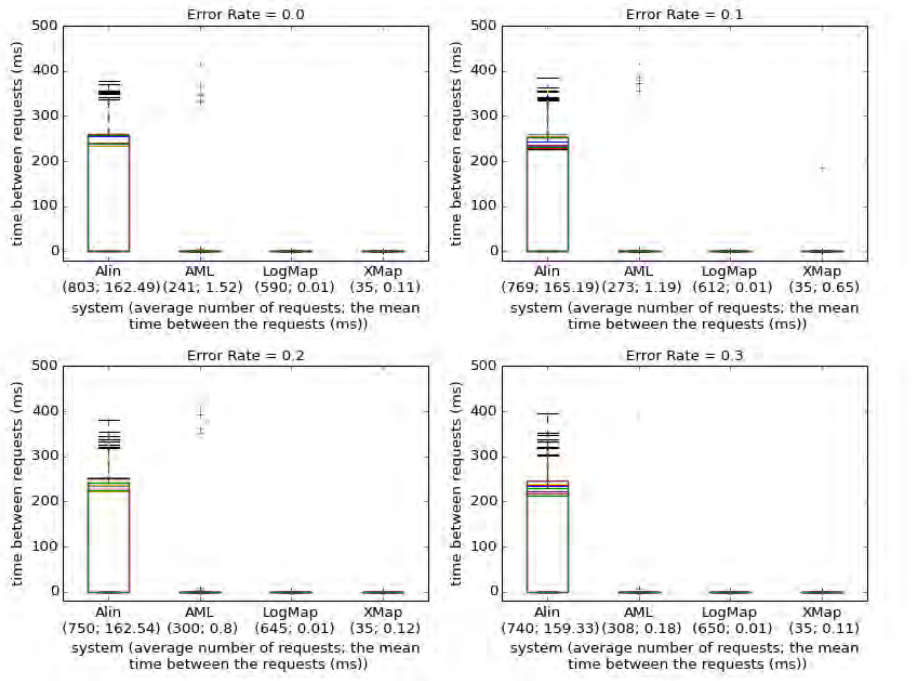


Fig. 5. Time intervals between requests to the user/oracle for the Anatomy data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the average number of requests and the mean time between the requests for the ten runs.

– 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

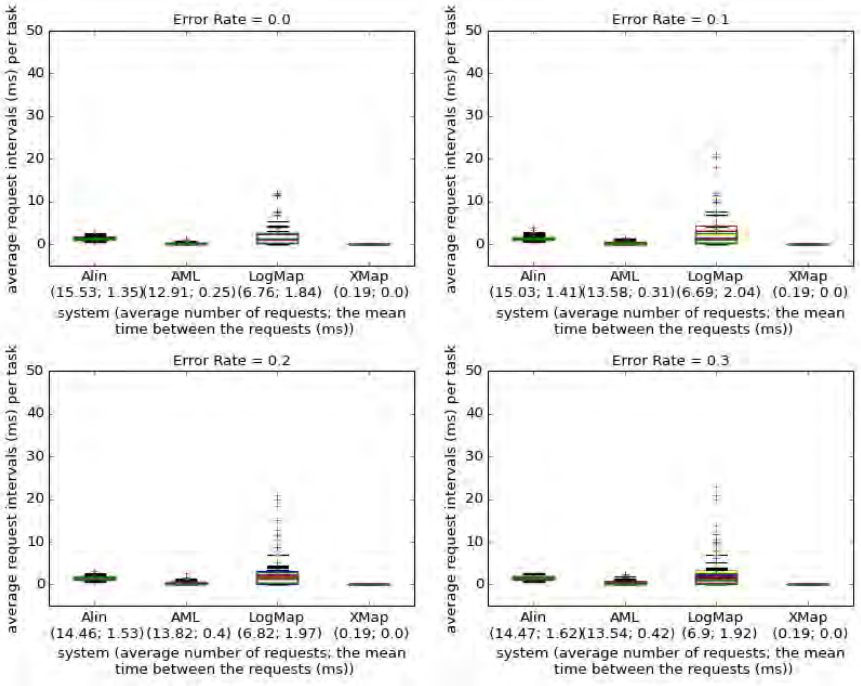


Fig. 6. Average time between requests per task in the Conference data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the average number of requests and the mean time between the requests (calculated by taking the average of the average request intervals per task)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

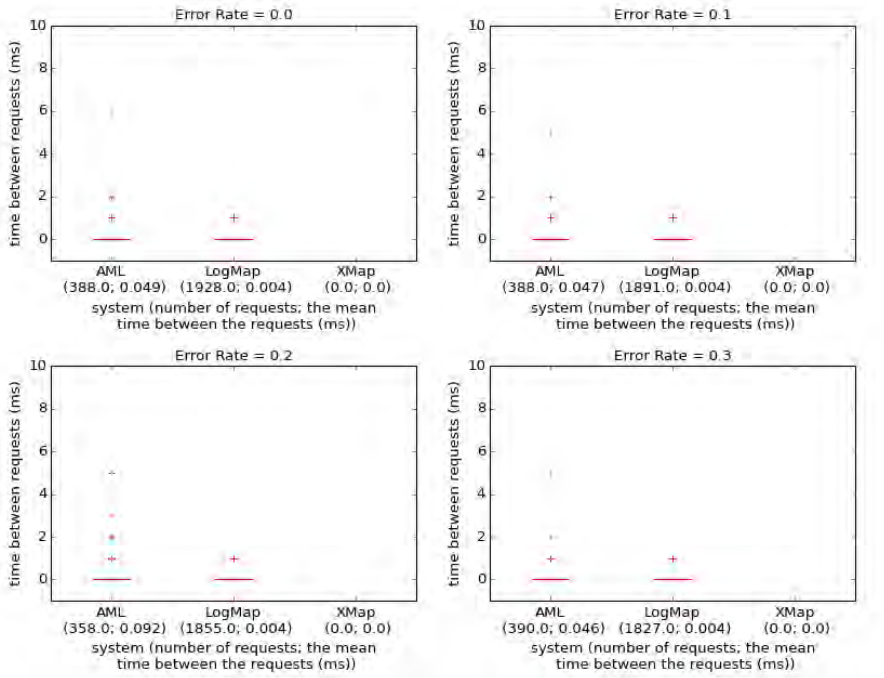


Fig. 7. Time between requests per task in the HP-MP data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the number of requests and the mean time between the requests.

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

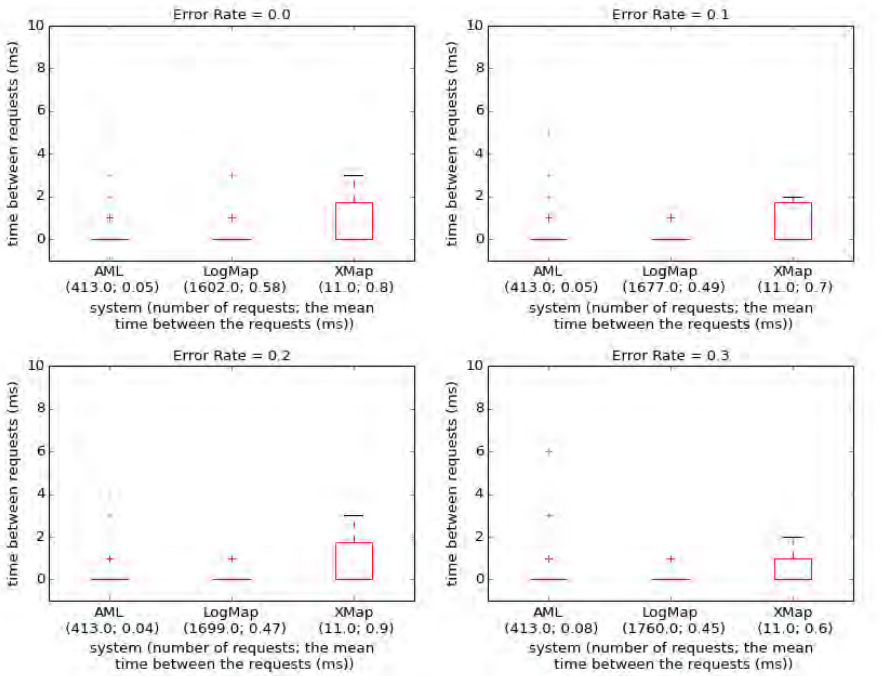


Fig. 8. Time between requests per task in the DOID-ORDO data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the number of requests and the mean time between the requests.

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.

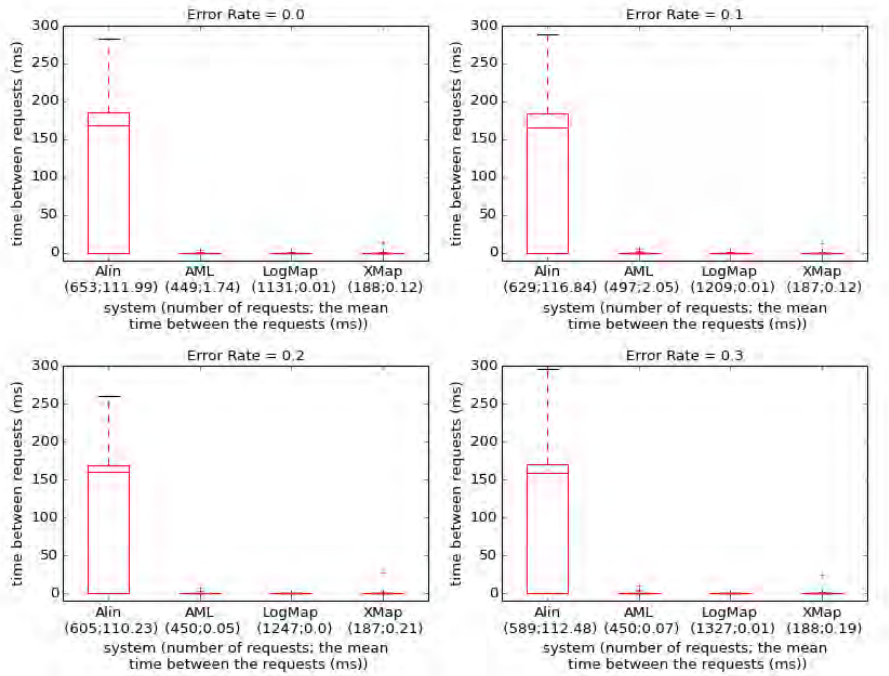


Fig. 9. Time between requests per task in the FMA-NCI data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the number of requests and the mean time between the requests.

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.

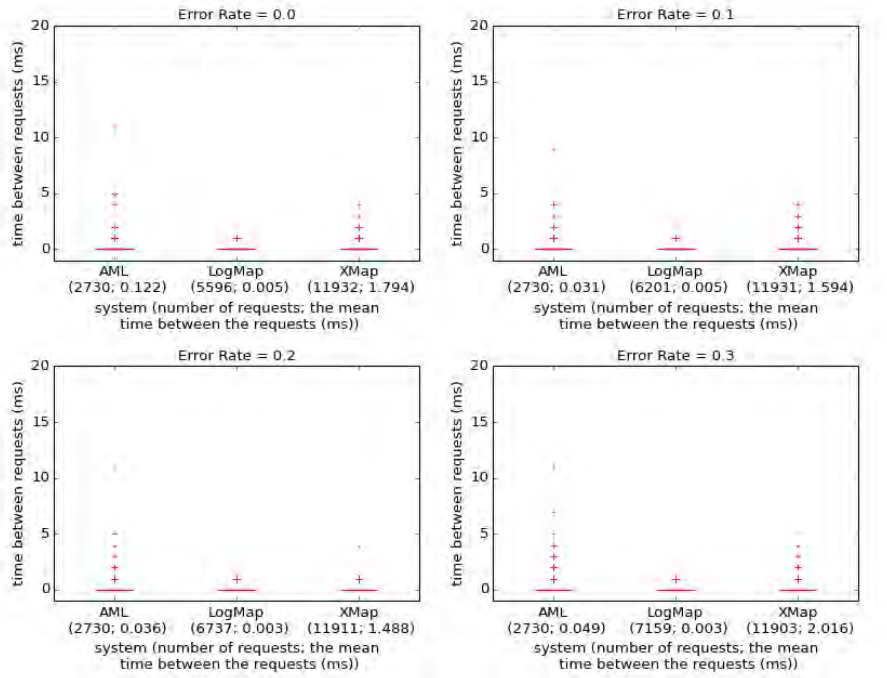


Fig. 10. Time between requests per task in the SNOMED-NCI data set (whiskers: Q1-1,5IQR, Q3+1,5IQR, IQR=Q3-Q1). The labels under the system names show the number of requests and the mean time between the requests.

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].

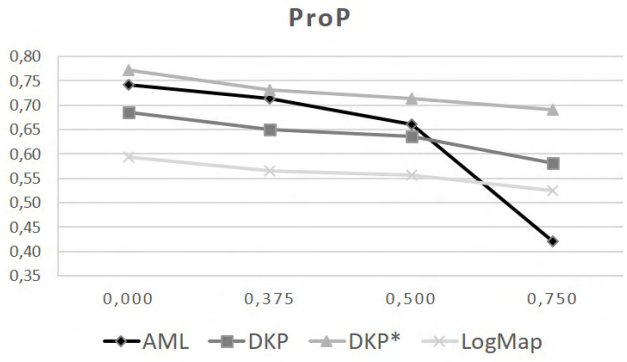
The results depicted in Table 47 and Figure 11 indicate that the progress made in ontology matching has also a positive impact on other related matching problems, like it is the case for process model matching. While it might require to reconfigure, adapt, and extend some parts of the ontology matching 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

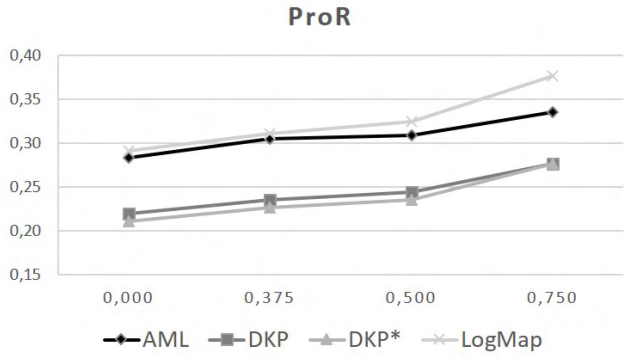
The lessons learned from running OAEI 2016 were the following:

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.

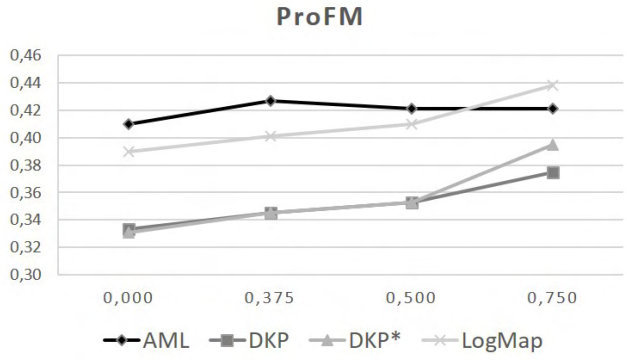
B) Thanks in part to the new submission schedule, this marked the first OAEI edition where all participants and all tracks were evaluated using the SEALS client. Nevertheless, some system developers still struggled to get their systems working with the client, mostly due to incompatible versions of libraries. 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

ALIN Results for OAEI 2016

Jomar da Silva, Fernanda Araujo Baião and Kate Revoredo

Department of Applied Informatics
Federal University of the State of Rio de Janeiro (UNIRIO), Rio de Janeiro, Brazil
{jomar.silva,fernanda.baiao,katerevoredo}@uniriotec.br

Abstract. ALIN is an ontology alignment system specialized in the interactive alignment of ontologies. Its main characteristic is the selection of correspondences to be shown to the expert, depending on the previous feedbacks given by the expert. This selection is based on semantic and structural characteristics. ALIN has obtained the alignment with the highest quality in the interactive tracking for Conference data set. This paper describes its configuration for the OAEI 2016 competition and discusses its results.

Keywords: Interactive Ontology Matching; Anti-patterns;

1 Presentation of the system

A large amount of data repositories became available due to the advances in information and communication technologies. Those repositories, however, are highly semantically heterogeneous, which hinders their integration. Ontology alignment has been successfully applied to solve this problem, by discovering correspondences between two distinct ontologies which, in turn, conceptually define the data stored in each repository. Among the various ontology alignment approaches that exist in the literature, interactive ontology alignment includes the participation of experts of the domain to improve the quality of the final alignment. This approach has proven more effective than non-interactive ontology alignment [1]. ALIN is an ontology alignment system specialized in interactive alignment. This is the first version of the system.

1.1 State, purpose, general statement

ALIN is an ontology alignment system, specialized in the ontology interactive alignment, based primarily on linguistic matching techniques, using the Wordnet as external resource. After generating an initial set of correspondences (called set of candidate correspondences, which are the correspondences selected to receive the feedback from the expert), interactions are made with the expert, and to each interaction, the set of candidate correspondences is modified. The modification of

the set of candidate correspondences is through the use of the structural analysis of ontologies and use of alignment anti-patterns. The interactions continue until there are no more candidate correspondences left. ALIN was built with a special focus on the interactive matching track of OAEI 2016.

1.2 Specific techniques used

The ALIN workflow is shown in figure 1.

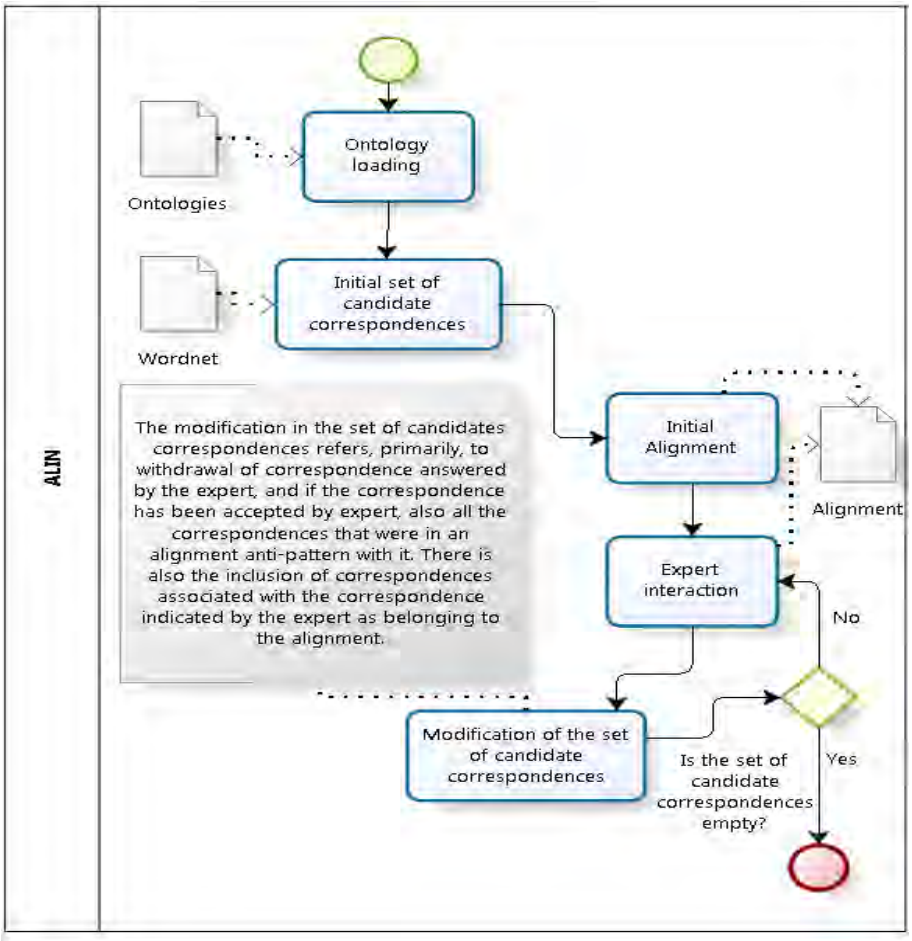


Fig. 1. – Workflow of ALIN

The steps of ALIN workflow are the following:

1. Load of the ontologies with load of classes, object properties and data properties through the Align API[1]. For each entity some data are stored such as name and label. In the case of classes, their superclasses and disjunctions are saved. In the case of object properties are saved the properties that are their hypernyms and their associated classes. The classes of property data are saved, too. ALIN does not use instances. After loading, the matching problem is profiled taking into account the size of the ontologies. The ALIN can only work with ontologies whose entity names are in English.
2. As an initial set of candidate correspondences a stable marriage algorithm with incomplete preference lists with maximum size of the list equals to 1, using linguistic metrics to sort the priority list was used [2]. The list is sorted in decreasing order. For this algorithm only the correspondence whose first entity is in the list of second entity and vice-versa is selected. The linguist metrics used are Jaccard, Jaro-Winkler and n-Gram [3] provided by Simmetrics API[2] and Wu-Palmer, Jiang-Conrath and Lin [3] provide by ws4j API[3] that use Wordnet. To use Wordnet the canonical form of the word is needed, therefore Stanford CoreNLP API[4] was considered. The algorithm is run six times, once by each metric, and the result set is the union of results of each metric.
3. The value of the similarity metrics (Wu-Palmer, Jiang-Conrath, Lin, Jaccard, Jaro-Winkler and n-Gram) vary from 0 to 1 (1 is the maximum value). When a correspondence in the set of candidate correspondences has all the six metrics with the maximum value, it is added to the final alignment and removed from the set of candidate correspondences. There are exceptions to this rule, some correspondences that fall into some structural patterns are not put on the final alignment and are not removed from the set of candidate correspondences.
4. The correspondences whose entities are not in the same synset of wordnet are removed from the set of candidate correspondences. These correspondences are put into a backup set, and can return to the set of candidate correspondences using structural analysis.
5. At this point the interactions with the expert begin. The correspondences in the set of candidate correspondences are sorted by the sum of similarity metric values, with the greatest sum first. The options are showed one by one to the expert. The first correspondence is showed and it is removed from the list after the answer of the expert. The set of candidate correspondences has, at first, only correspondences of classes. When the expert answer one question, the set of candidate correspondences is changed. Correspondences (besides the

1 "Alignment API". Available at http://alignapi.gforge.inria.fr/ Last accessed on Apr, 11, 2016.

2 "String Similarity Metrics for Information Integration". Available on http://www.coli.uni-saarland.de/courses/LT1/2011/slides/stringmetrics.pdf. Last accessed on Apr, 19, 2016.

3 "WS4J". Available at https://code.google.com/archive/p/ws4j/ Last accessed on Apr, 11, 2016.

4 "Stanford CoreNLP". Available at http://stanfordnlp.github.io/CoreNLP/ Last accessd on Sept, 15, 2016.

correspondence answered by expert) can be removed and included, depending on the answer of the expert. If the expert does not accept the correspondence it is removed from the set of candidate correspondences. But if the expert accepts the correspondence it is removed from the set of candidate correspondences and put in the final alignment.

At each interaction with the specialist we also:

- We remove from the set of candidate correspondences and disregard all the correspondences that are in anti-pattern of alignment [4]with the correspondence accepted by the expert;
- We insert into the set of candidate correspondences, correspondences of data properties and correspondences of object properties related to the correspondence of classes accepted by the expert.
- We insert into the set of candidate correspondences, correspondences of the backup set (step 4) whose both entities are subclasses of the classes of a correspondence accepted by expert.

This step continues until the set of candidate correspondences is empty.

1.3 Link to the system and parameters file

ALIN is available through Mediafire (https://www.mediafire.com/folder/726zohj792kod/ALIN) as a package for running through the SEALS client.

2 Results

The system ALIN has been developed with its focus on interactive ontology alignment. The approach performs better when the number of data and object properties is proportionately large. ALIN considers properties associated to correspondent classes when selecting entities for user feedback, thus allowing for increased recall. When the number of properties in the ontologies is small, the system still generates a very precise alignment, but its recall tends to decrease.

Another characteristic of ALIN is its reliance on an interactive phase. The non-interactive phase of the system is quite simple, mainly based on maximum string similarity, specializing in maintaining a high precision without worrying about recall, generating initially a low f-measure. The recall increases in the interactive phase. Finally, ALIN is also not robust to users errors. The system uses a number of techniques that take advantage of the expert response to reach other conclusions when the expert gives a wrong answer it is propagated generating other errors, thereby diminishing the f-measure.

2.1 Comments on the participation of the ALIN in non-interactive tracks

As expected the participation of ALIN in non-interactive alignment processes showed the following results: high precision and not so high recall, as can be seen in Table 1, where recall+ field refers to non-trivial correspondences found and Coherent field filled by + indicates that the generated alignment is consistent.

Matcher	Runtime	Size	Precision	F-Measure	Recall	Recall+	Coherent
Alin	306	510	0.996	0.501	0.335	0.0	+

Table 1. - Participation of ALIN in Anatomy track

Matcher	Threshold	Precision	F.5-measure	F1-measure	F2-measure	Recall
Alin	0	0.89	0.65	0.46	0.36	0.31

Table 2. - Participation of ALIN in Conference track taking into account only the classes (m1) , and the reference alignment publicly available (r1).

Matcher	Threshold	Precision	F.5-measure	F1-measure	F2-measure	Recall
Alin	0	0	0	0	0	0

Table 3. - Participation of ALIN in Conference track taking into account only the properties (m2) and the reference alignment publicly available (r1)

Matcher	Threshold	Precision	F.5-measure	F1-measure	F2-measure	Recall
Alin	0	0.89	0.6	0.4	0.3	0.26

Table 4. - Participation of ALIN in Conference track taking into account the classes and properties (m3), and the reference alignment publicly available (r1).

Regarding the Conference track, as ALIN evaluates only the properties associated with classes already evaluated as belonging to the alignment, the alignment of the M2 type (which take into account only the properties of ontologies) were with the f-measure = 0, as can be seen in Table 3. As properties are evaluated only in the interactive phase in the ALIN, alignments of type M1 (only classes) remained with a higher recall than M3 (classes and properties), as can be seen in Tables 2 and 4, because the reference alignments of type M3 contain properties besides classes.

2.2 Comments on the participation of ALIN in interactive tracks

Anatomy track.
In this track the program ALIN showed the highest precision among the four evaluated tools when the error rate is zero. When the error rate increases both the precision as the recall falls, reducing the f-measure. This is expected and explained earlier.

Error Rate 0.0

Tool	Run Time (sec)	Size	Precision	Recall	F-measure	Recall+	Precision Non Inter	Recall Non Inter	F-measure Non Inter	Recall+ Non Inter	Precision Oracle	Recall Oracle	F-measure Oracle	Total Requests	Distinct Mappings	True Positives	True Negatives	False Positives	False Negative	Precision	Negative Precision
⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩
Alin	505	1142.8	0.993	0.749	0.854	0.35	0.984	0.335	0.5	0	0.993	0.749	0.854	803	1221	626	594	0	0	1	1
AML	48	1494	0.968	0.948	0.958	0.862	0.95	0.936	0.943	0.832	0.968	0.948	0.958	241	240	51	189	0	0	1	1
LogMap	27	1306	0.982	0.846	0.909	0.595	0.911	0.846	0.877	0.593	0.982	0.846	0.909	590	590	287	303	0	0	1	1
XMap	49	1416	0.929	0.867	0.897	0.653	0.928	0.865	0.895	0.647	0.929	0.867	0.897	35	35	5	30	0	0	1	1

Error Rate 0.1

Tool	Run Time (sec)	Size	Precision	Recall	F-measure	Recall+	Precision Non Inter	Recall Non Inter	F-measure Non Inter	Recall+ Non Inter	Precision Oracle	Recall Oracle	F-measure Oracle	Total Requests	Distinct Mappings	True Positives	True Negatives	False Positives	False Negative	Precision	Negative Precision
⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩
Alin	489	1121.1	0.947	0.701	0.805	0.307	0.984	0.335	0.5	0	0.993	0.742	0.85	769	1123	554	451	51	67	0.916	0.87
AML	50	1503.7	0.953	0.945	0.949	0.856	0.95	0.936	0.943	0.832	0.969	0.95	0.959	273	272	47	194	23	6	0.67	0.967
LogMap	24	1312	0.961	0.832	0.892	0.565	0.911	0.846	0.877	0.593	0.964	0.831	0.893	612	612	258	290	35	28	0.881	0.912
XMap	46	1416.1	0.929	0.867	0.897	0.653	0.928	0.865	0.895	0.647	0.929	0.866	0.896	35	35	4.2	27.5	2.7	0.8	0.609	0.972

Error Rate 0.2

Tool	Run Time (sec)	Size	Precision	Recall	F-measure	Recall+	Precision Non Inter	Recall Non Inter	F-measure Non Inter	Recall+ Non Inter	Precision Oracle	Recall Oracle	F-measure Oracle	Total Requests	Distinct Mappings	True Positives	True Negatives	False Positives	False Negative	Precision	Negative Precision
⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩
Alin	481	1103.7	0.907	0.661	0.765	0.266	0.984	0.335	0.5	0	0.993	0.736	0.845	750	1077	493	368	94	121	0.84	0.753
AML	48	1521.9	0.939	0.943	0.941	0.85	0.95	0.936	0.943	0.832	0.969	0.952	0.96	300	299	46	193	46	13	0.503	0.936
LogMap	24	1314.2	0.944	0.819	0.877	0.543	0.911	0.846	0.877	0.593	0.944	0.814	0.874	645	645	225	287	70	61	0.761	0.823
XMap	47	1416.1	0.929	0.867	0.897	0.653	0.928	0.865	0.895	0.647	0.929	0.865	0.896	35	35	4.3	25.1	5.4	0.7	0.448	0.974

Error Rate 0.3

Tool	Run Time (sec)	Size	Precision	Recall	F-measure	Recall+	Precision Non Inter	Recall Non Inter	F-measure Non Inter	Recall+ Non Inter	Precision Oracle	Recall Oracle	F-measure Oracle	Total Requests	Distinct Mappings	True Positives	True Negatives	False Positives	False Negative	Precision	Negative Precision
⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩	⇧⇩
Alin	472	1080.1	0.869	0.619	0.723	0.231	0.984	0.335	0.5	0	0.993	0.73	0.841	740	1058	430	311	134	182	0.763	0.631
AML	47	1540.1	0.925	0.94	0.932	0.843	0.95	0.936	0.943	0.832	0.969	0.952	0.96	308	307	40	177	71	18	0.363	0.905
LogMap	24	1327.2	0.932	0.817	0.871	0.541	0.911	0.846	0.877	0.593	0.926	0.8	0.858	650	650	202	256	106	84	0.655	0.752
XMap	47	1416.4	0.929	0.867	0.897	0.653	0.928	0.865	0.895	0.647	0.929	0.864	0.895	35	35	3.1	21.8	8.9	1.9	0.265	0.92

Table 5. - Participation of ALIN in interactive alignment - Anatomy track.

Tool	Run Time (sec)	Precision	Recall	F-measure	Precision Non Inter	Recall Non Inter	F-measure Non Inter	Precision Oracle	Recall Oracle	F-measure Oracle	Total Requests	Distinct Mappings	True Positives	True Negatives	False Positives	False Negative	Precision	Negative Precision
⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩
Alin	101	0.957	0.735	0.831	0.888	0.259	0.401	0.957	0.735	0.831	326	574	144	429	0	0	1	1
AML	29	0.912	0.711	0.799	0.841	0.659	0.739	0.912	0.711	0.799	271	270	47	223	0	0	1	1
LogMap	26	0.886	0.61	0.723	0.818	0.59	0.686	0.886	0.61	0.723	142	142	49	93	0	0	1	1
XMap	21	0.837	0.574	0.681	0.837	0.574	0.681	0.837	0.574	0.681	4	4	0	4	0	0	0	1

Error Rate 0.1

Tool	Run Time (sec)	Precision	Recall	F-measure	Precision Non Inter	Recall Non Inter	F-measure Non Inter	Precision Oracle	Recall Oracle	F-measure Oracle	Total Requests	Distinct Mappings	True Positives	True Negatives	False Positives	False Negative	Precision	Negative Precision
⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩
Alin	101	0.794	0.67	0.727	0.888	0.259	0.401	0.961	0.743	0.838	315	557	124	375	42	15	0.747	0.962
AML	30	0.847	0.703	0.768	0.841	0.659	0.739	0.921	0.732	0.816	285	279	51	204	18	5	0.74	0.977
LogMap	26	0.847	0.6	0.702	0.818	0.59	0.686	0.855	0.593	0.701	140	140	45	81	10	3	0.819	0.965
XMap	22	0.837	0.574	0.681	0.837	0.574	0.681	0.837	0.573	0.68	4	4	0	3.6	0.4	0	0	1

Error Rate 0.2

Tool	Run Time (sec)	Precision	Recall	F-measure	Precision Non Inter	Recall Non Inter	F-measure Non Inter	Precision Oracle	Recall Oracle	F-measure Oracle	Total Requests	Distinct Mappings	True Positives	True Negatives	False Positives	False Negative	Precision	Negative Precision
⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩
Alin	100	0.672	0.615	0.642	0.888	0.259	0.401	0.964	0.748	0.843	303	538	108	321	81	27	0.572	0.923
AML	33	0.767	0.681	0.721	0.841	0.659	0.739	0.925	0.745	0.825	290	277	53	170	42	11	0.558	0.94
LogMap	26	0.822	0.588	0.686	0.818	0.59	0.686	0.831	0.579	0.682	143	143	38	75	18	10	0.679	0.883
XMap	21	0.837	0.574	0.681	0.837	0.574	0.681	0.837	0.572	0.68	4	4	0	3.2	0.8	0	0	1

Error Rate 0.3

Tool	Run Time (sec)	Precision	Recall	F-measure	Precision Non Inter	Recall Non Inter	F-measure Non Inter	Precision Oracle	Recall Oracle	F-measure Oracle	Total Requests	Distinct Mappings	True Positives	True Negatives	False Positives	False Negative	Precision	Negative Precision
⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩	⇧ ⇩
Alin	99	0.57	0.568	0.569	0.888	0.259	0.401	0.967	0.767	0.855	303	535	93	279	120	42	0.437	0.87
AML	30	0.718	0.651	0.683	0.841	0.659	0.739	0.929	0.75	0.83	284	269	47	143	58	20	0.448	0.878
LogMap	26	0.803	0.585	0.677	0.818	0.59	0.686	0.804	0.563	0.662	144	144	33	67	28	15	0.541	0.818
XMap	22	0.837	0.574	0.681	0.837	0.574	0.681	0.837	0.572	0.68	4	4	0	2.9	1.1	0	0	1

Table 6. - Participation of ALIN in interactive alignment - Conference track.

As ontologies of the Anatomy Track contains almost no properties, techniques used in ALIN can not be utilized, the selection of properties associated with classes assessed as belonging to the alignment, this has limited the increase in recall, which influenced the f-measure, as can be seen in Table 5.

Conference Track.
In this track ALIN stood out, showing the greatest f-measure among the four tools when the error rate is zero, as with a loss of f-measure when the error rate increases, as can be seen in Table 6.

3 General Comments

Evaluating the results it can be seen that the system can be improved towards:
(a) handling user error rate;
(b) generating a higher quality (especially w.r.t. recall) initial alignment in its non-interactive phase;
(c) reducing the number of interactions with the expert; and
(d) optimize the process to reduce its execution time.

4 Conclusions

Within certain characteristics, the ALIN system stands out in ontology alignment process in interactive application scenarios, especially when the amount of data and object properties are also subject to the alignment and when the expert does not make mistakes. With these features there is an alignment generated with relatively high precision and recall.

References

[1] H. Paulheim, S. Hertling, e D. Ritze, "Towards Evaluating Interactive Ontology Matching Tools", *Lect. Notes Comput. Sci.*, vol. 7882, p. 31–45, 2013.

[2] R. W. Irving, D. F. Manlove, e G. O'Malley, "Stable marriage with ties and bounded length preference lists", *J. Discret. Algorithms*, vol. 7, nº 2, p. 213–219, 2009.

[3] J. Euzenat e P. Shvaiko, *Ontology Matching - Second Edition*, 2°. Springer-Verlag, 2013.

[4] A. Guedes, F. Baião, e K. Revoredo, "Digging Ontology Correspondence Antipatterns", *Proceeding WOP '14 Proc. 5th Int. Conf. Ontol. Semant. Web Patterns*, vol. 1302, p. 38–48, 2014.

OAEI 2016 Results of AML

Daniel Faria[1], Catia Pesquita[2], Booma S. Balasubramani[3], Catarina Martins[2], João Cardoso[4], Hugo Curado[2], Francisco M. Couto[2], and Isabel F. Cruz[3]

[1] Instituto Gulbenkian de Ciência, Portugal
[2] LaSIGE, Faculdade de Ciências, Universidade de Lisboa, Portugal
[3] ADVIS Lab, Department of Computer Science, University of Illinois at Chicago, USA
[4] INESC-ID, Instituto Superior Técnico, Universidade de Lisboa

Abstract. AgreementMakerLight (AML) is an automated ontology matching system based primarily on element-level matching and on the use of external resources as background knowledge. This paper describes its configuration for the OAEI 2016 competition and discusses its results.
For this OAEI edition, we tackled instance matching for the first time, thus expanding the coverage of AML to all types of ontology matching tasks. We also explored OBO logical definitions to match ontologies for the first time in the OAEI.
AML was the top performing system in five tracks (including the Instance and instance-based Process Model tracks) and one of the top performing systems in three others (including the novel Disease and Phenotype track, in which it was one of three prize recipients).

1 Presentation of the System

1.1 State, Purpose, General Statement

AgreementMakerLight (AML) is an automated ontology matching system derived from AgreementMaker [3, 4] and designed to tackle large-scale matching problems [6]. It is based primarily on lexical matching techniques, with an emphasis on the use of external resources as background knowledge.
This year, our development of AML was focused primarily on tackling instance matching, an aspect of ontology matching that was missing from its portfolio. However, we also made several developments with regard to class matching, namely with the use of OBO logical definitions.
For this OAEI edition, we also decided to adopt the solution of using configuration files for each track in order to specify the parameters of the matching task (such as whether to match classes, properties, and/or instances) rather than submit a preconfigured system. With this, we aim at providing a more transparent approach to our participation in the OAEI.

1.2 Specific Techniques Used

For the sake of brevity, this section describes only the features of AML that are new for this edition of the OAEI. For a complete description of AML's matching strategy,

please refer to last year's OAEI results paper [5].

1.2.1 Ontology Data

To store data about ontology individuals, we expanded AML's *Lexicon* and *RelationshipMap* data structures [6] and created the new *ValueMap*. The current organization of these data structures is the following:

- The *Lexicon* of each *Ontology* stores local names (if not alpha-numeric codes), labels and other lexical annotations of classes, individuals, and properties, after normalizing them.
- The *ValueMap* of each *Ontology* stores all other annotations of individuals and their data property values.
- The global *RelationshipMap* stores relations between classes, between individuals, between properties, classes instanced by individuals, and property domains and ranges.

1.2.2 Instance Matching

For instance matching, AML's core strategy consists of three matching algorithms:

- The *HybridStringMatcher* which matches two entities by computing the maximum of the string similarity, word similarity, and WordNet similarity between their *Lexicon* entries. It is the algorithm AML already used to match properties.
- The *ValueStringMatcher* which matches two individuals by computing the maximum string similarity between their *ValueMap* entries, penalizing matches where the annotation or data property is not the same.
- The *Value2LexiconMatcher* which employs the same combination of similarity metrics as the *HybridStringMatcher*, but compares *Lexicon* entries of one entity with *ValueMap* entries of the other and vice versa.

AML's similarity score is the maximum of these three algorithms, but it uses a linear combination of the three to break similarity ties when performing alignment selection. AML deviates from this core strategy in three circumstances:

- When the matching problem requires translation, in which case it employs the same matching strategy used for classes and properties when translation is involved.
- When the ontologies have a high individual connectivity (indicating that there is a network or pipeline of individuals), in which case it employs the *ProcessMatcher* algorithm that was developed for matching business process models [2]. It combines string similarity with structural similarity.
- When the fraction of individuals with exactly matching values in the ontologies is high (meaning that matches based on values have low significance), in which case it employs only the *HybridStringMatcher*.

1.2.3 Exploring OBO Logical Definitions

OBO [12] logical definitions (or cross-products) provide definitions of ontology classes by establishing intersections between other classes, typically from different ontologies.

For example, the logical definition of the class Human Phenotype Ontology (HP) [10] class HP:0005815 (“supernumerary ribs”) corresponds to an intersection of the Phenotypic Quality Ontology class PATO:0002002 (“has extra parts of type”) and the Foundational Model of Anatomy (FMA) class [11] FMA:7574 (“rib”) via an ‘inheres_in’ relation. We had previously developed a variant of AML for computing this type of compound mapping [8].
For this year’s OAEI, we explored the use of these logical definitions to match ontologies that contain them. Continuing the previous example, the Mammalian Phenotype Ontology (MP) [13] contains the class MP:0000480 (“increased rib number”) which to an English-speaking human should be obvious that it corresponds to the HP class above. However, to lexical ontology matching algorithms this correspondence is very hard to detect. Logical definitions can help us find this mapping, as MP defines that the class above corresponds to an intersection of the same class PATO:0002002 and the UBERON class [7] UBERON:0002228 (“rib”). As UBERON has cross-references to FMA, we can automatically establish a correspondence between UBERON:0002228 and FMA:7574, and thus find the mapping HP:0005815 <=> MP:0000480.
Because the versions of HP and MP used in the OAEI didn’t include the logical definitions in the ontology files (as the versions available at the OBO portal do), we used an external file containing these definitions as background knowledge.

1.2.4 Thesaurus Matching

For this year’s OAEI we also employed a matching algorithm based on a thesaurus that is automatically derived from the ontologies by comparing labels and synonyms for the same classes, as we have described in a previous study [9]. We hadn’t used this strategy in previous OAEI editions because our original implementation was too broad and consequently both too imprecise and too inefficient computationally. We addressed these problems by making a more restrictive implementation.
Currently, the algorithm infers synonyms to populate the thesaurus only when two *Lexicon* entries for a class have the same number of words and all their words are equal except for one, in which case the words in which they differ are inferred to be synonymous. Additionally, the new *Lexicon* entries generated for classes using the thesaurus are now only used to check for literal full-name matches, whereas previously they were also used with string similarity algorithms.

1.3 Adaptations made for the evaluation

The adaptations made for the evaluation were: the preprocessing of cross-references from Uberon and DOID for use in the Anatomy and Large Biomedical Ontologies tracks, due to namespace differences; the use of an external logical definitions file, due to the absence of these in the versions of the ontologies used in the Disease and Phenotype track; and the precomputing of translations, due to Microsoft® Translator’s query limit.

1.4 Link to the system and parameters file

AML is an open source ontology matching system and is available through GitHub (https://github.com/AgreementMakerLight) as an Eclipse project, as a stand-alone Jar application, and as a package for running through the SEALS client.

2 Results

2.1 Anatomy

Thanks to the use of the new thesaurus matching algorithm, AML improved both its recall and recall+ to the highest ever results in this track (93.6% and 83.2% respectively). However, it had a 0.6% drop in precision and a 0.1% drop in F-measure in comparison with last year. It remains the best performing system in this track.

2.2 Benchmark

As in previous years, AML obtained a very high precision in this track (this year the highest, at 100%) but a low recall (0.24%) and consequently a low F-measure as well (38%). We maintain AML focused on matching real-world ontologies, and have not prioritized the Benchmark track.

2.3 Conference

AML's performance in the Conference track was exactly the same as last year, as the new developments do not affect its performance in this track. It remains the best performing system overall in this track, with the highest F-measure on the full reference alignment 1 (74%), on the full reference alignment 2 (70%, tied with CroMatch), and on both evaluation modalities with the uncertain reference alignment (Discrete: 78%; Continuous: 77%).
Concerning the logical reasoning evaluation, AML again had no consistency principle violations, but did have conservativity principle violations as this is an aspect AML deliberately doesn't take into account given that many of these violations are false positives.

2.4 Disease and Phenotype

AML was considered one of the three top systems in the Disease and Phenotype track. In the HP-MP task, it obtained F-measures of 86% and 89.7% according to the 2-vote and 3-vote silver standards, respectively, and produced 122 unique mappings with 86.7% precision. In the DOID-ORDO task, it obtained F-measures of 90.8% and 87.5% according to the 2-vote and 3-vote silver standards, respectively, and produced 308 unique mappings, with an estimated precision of 86.7%. AML's performance in capturing the manually created mappings was poorer (75.9% and 0% recall, for HP-Mp and DOID-ORDO respectively), since the majority of these mappings are subsumption ones and AML focuses on equivalence matching.

2.5 Instance Matching

In the Sabine sub-track, AML obtained the second highest F-measure in the Sabine Linguistic task, with 91.8%, and the highest F-measure in the Sabine Linking task, with 88.9%.
In the Synthetic sub-track, AML obtained the highest F-measure in the UOBM mainbox task, with 51.2%, and the second highest F-measure in the SPIMBENCH mainbox task, with 81.6%. Interestingly, it ranked lower on the corresponding sandbox versions (second in UOBM with 66.5%, and third in SPIMBENCH with 82%) and was the system that lost the least performance between the sandbox and the mainbox tasks. Additionally, it is important to mention that AML does not process or attempt to match individuals without class assignment, and that there were a number of these in both the UOBM and SPIMBENCH ontologies which were supposed to be matched, which resulted in lower scores for AML.
In the Doremus sub-track, AML obtained the highest F-measure in all three tasks, with 91.8% in the 9 heterogeneities task, 84.8% in the larger 4 heterogeneities task, and 88.60% in the false-positive track task.
Overall, AML obtained the top F-measure in five of the seven Instance Matching tasks, and second in the other two, making it overall the most successful instance matching system in the OAEI 2016.

2.6 Interactive Matching

AML had a worse performance than last year in this track, due to changes to its user interface to enable alignment revision, which affected the internal functioning of the interactive matching algorithm. We were unable to completely solve this issue in time for the evaluation. Nevertheless, in the Anatomy dataset, AML still had the highest F-measure (95.8% with 0% errors), the lowest number of oracle requests, and the lowest impact of errors, with a drop in performance under 3% between 0 and 30% errors. In the Conference dataset, it was surpassed by Alin in F-measure and by LogMap with regard to the lowest number of requests and lowest impact of errors.

2.7 Large Biomedical Ontologies

Like in the Anatomy track, the introduction of the thesaurus matching algorithm led to an improved recall from AML on the Large Biomedical Ontologies track, and as a result AML had a higher F-measure overall in all tasks than in previous years. Despite this, it was surpassed in F-measure on the FMA-NCI small and FMA-SNOMED small tasks, obtaining only the second-highest F-measure (ignoring the XMAP results, since this system uses the UMLS metathesaurus as background knowledge, which is the basis of the reference alignments). Nevertheless, it remains the best performing system able to complete all the tasks of this track, and the one that produces the most coherent alignments.

2.8 Multifarm

AML obtained the top F-measure when matching the same ontologies, and the third best when matching the same ontologies, due to lowered recall. Despite not being a systems specifically targeting cross-lingual matching, by using a translation module AML is able to achieve a good ranking in performance in this track.

2.9 Process Model

AML obtained the top F-measure result in this track, with 70.2%, surpassing not only all other ontology matching systems, but also all process model matching systems from last year's process model matching competition [1].

3 General comments

3.1 Comments on the results

AML remained among the top performing systems in nearly all preexisting tracks, while also obtaining top results in the new tracks: Disease and Phenotype, in which it was one of the prize winners; Process Model, in which it surpassed the results of (non-ontology) process model matchers; and Instance Matching with all new datasets. It was also consistently among the fastest systems and among those that produced the most coherent alignments. These results reflect our continued effort to extend AML to cover all types of ontology matching tasks while ensuring that it remains both effective and efficient.

3.2 Comments on the OAEI test cases

We welcomed the efforts to expand the scope of OAEI with new tracks and improve existing ones. We take this opportunity to highlight some issues we encountered during this year's competition, and suggest some possible improvements for future editions.
This year there were several issues with the test cases from the Instance Matching track: there were encoding problems associated with the Sabine datasets; there were instances without class assignments in the Synthetic and Doremus datasets, and in the case of the former, some of these instances were supposed to be matched; and the target ontology in the SPIMBENCH mainbox dataset was inconsistent. These are all issues that can be found in real-world datasets, and both the developers and users of ontology matching systems should be aware of them, but we believe that asking systems to handle such specific issues involves a high level of manual work and tuning of the systems, making their comparison less straightforward and transparent.
We also find that the evaluation in the Disease and Phenotype track still has room for improvement. Generating silver standards from the alignments produced by the participating systems via voting is a reasonable starting point for producing a reference alignment, but an insightful evaluation would then need that the silver consensus standards be manually validated, as well as the unique mappings produced by each system. Since only the latter manual evaluation was done, and for only up to 30 mappings, this

distorts the results as the evaluation will include wrong mappings (that multiple systems get wrong) and miss correct mappings (that only one system finds). Additionally, we propose that in next years the versions of the HP and MP ontologies used in this track include logical definitions, so other systems can also explore them.

4 Conclusion

In 2016, AML was the top performing system in five tracks (Anatomy, Conference, Instance, Multifarm, and Process Model) and one of the top performing systems in three others (Disease and Phenotype, Interactive, and Large Biomedical Ontologies). It fully met our goals and expectations for this year's competition, and rewarded our investment in instance matching (with top results in both Instance and Process Model) and our use of logical definitions (with a prize in the Disease and Phenotype track).
Nevertheless we remark with enthusiasm on the improvement of other matching systems in tracks such as Anatomy, Conference, and Large Biomedical Ontologies. While in previous years we could be led to the conclusion that ontology matching was stagnating, and that surpassing the results of the top systems would be a tall order, the results of this year's OAEI show that that is not the case.

Acknowledgments

The authors are thankful to Daniela Oliveira (Insight Centre for Data Analytics, NIU Galway, Ireland) for her support in the alignment of phenotype ontologies, and to André Oliveira, Filipa Marques and Tânia Maldonado, for their contribution to analyzing the test cases and results. FMC, CM and CP were funded by the Portuguese FCT through the LASIGE Strategic Project (UID/CEC/00408/2013). CP was also funded by FCT (PTDC/EEI-ESS/4633/2014). The research of IFC and BS was partially supported by a grant from the Bloomberg Philanthropies and by NSF awards CNS-1646395, III-1618126, CCF-1331800, III-1213013, and IIS-1143926.

References

1. G. Antunes, M. Bakhshandeh, J. Borbinha, J. Cardoso, S. Dadashnia, C. Francescomarino, M. Dragoni, P. Fettke, A. Gal, C. Ghidini, et al. The process model matching contest 2015. In *6th EMISA Workshop*, pages 127–155, 2015.
2. J. Cardoso, M. Bakhshandeh, D. Faria, C. Pesquita, and J. Borbinha. Ontology-Based Approach for Heterogeneity Analysis of EA Models. In *Workshop on Business Process Management and Ontologies*, 2016.
3. I. F. Cruz, F. Palandri Antonelli, and C. Stroe. AgreementMaker: Efficient Matching for Large Real-World Schemas and Ontologies. *PVLDB*, 2(2):1586–1589, 2009.
4. I. F. Cruz, C. Stroe, F. Caimi, A. Fabiani, C. Pesquita, F. M. Couto, and M. Palmonari. Using AgreementMaker to Align Ontologies for OAEI 2011. In *ISWC International Workshop on Ontology Matching (OM)*, volume 814 of *CEUR Workshop Proceedings*, pages 114–121, 2011.

5. D. Faria, C. Martins, A. Nanavaty, D. Oliveira, B. S. Balasubramani, A. Taheri, C. Pesquita, F. M. Couto, and I. F. Cruz. AML results for OAEI 2015. In *Ontology Matching Workshop*. CEUR, 2015.
6. D. Faria, C. Pesquita, E. Santos, M. Palmonari, I. F. Cruz, and F. M. Couto. The AgreementMakerLight Ontology Matching System. In *OTM Conferences - ODBASE*, pages 527–541, 2013.
7. C. J. Mungall, C. Torniai, G. V. Gkoutos, S. Lewis, and M. A. Haendel. Uberon, an Integrative Multi-species Anatomy Ontology. *Genome Biology*, 13(1):R5, 2012.
8. D. Oliveira and C. Pesquita. Compound matching of biomedical ontologies. In *International Conference on Biomedical Ontology*, volume 1515. CEUR, 2015.
9. C. Pesquita, D. Faria, C. Stroe, E. Santos, I. F. Cruz, and F. M. Couto. What's in a "nym"? Synonyms in Biomedical Ontology Matching. In *International Semantic Web Conference (ISWC)*, pages 526–541, 2013.
10. P. N. Robinson, S. Köhler, S. Bauer, D. Seelow, D. Horn, and S. Mundlos. The human phenotype ontology: a tool for annotating and analyzing human hereditary disease. *The American Journal of Human Genetics*, 83(5):610–615, 2008.
11. C. Rosse, J. L. Mejino Jr, et al. A reference ontology for biomedical informatics: the foundational model of anatomy. *Journal of biomedical informatics*, 36(6):478–500, 2003.
12. B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, W. Ceusters, L. J. Goldberg, K. Eilbeck, A. Ireland, C. J. Mungall, et al. The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, 25(11):1251–1255, 2007.
13. C. L. Smith and J. T. Eppig. The mammalian phenotype ontology as a unifying standard for experimental and high-throughput phenotyping data. *Mammalian genome*, 23(9-10):653–668, 2012.

CroLOM: Cross-Lingual Ontology Matching System
Results for OAEI 2016

Abderrahmane Khiat

LITIO Laboratory, University of Oran1 Ahmed Ben Bella, Oran, Algeria
abderrahmane_khiat@yahoo.com

Abstract. The current work describes an automatic system especially designed for aligning cross-lingual ontologies. The CroLOM software, unlike existing systems, uses the Yandex translator, NLP techniques and a similarity computation based on the categories of the words and synonyms. CroLOM participated for the first time in OAEI2016 evaluation campaign and the results obtained are so far been quite promising. The paper also discusses some important issues related to multilingualism treatment.

Keywords: Cross lingual Alignment, Multilingual Ontologies Survey, Ontology Matching, Yandex, Semantic Similarity, OAEI, Direct matching.

1 Introduction

Recently, with the growing number of ontologies defined in different languages, multilingualism has become an issue of major interest in ontology matching field. Multilingual ontology alignment, defined as the process of identification of semantic correspondences between entities of different ontologies described in different natural language, represents the solution to the problem of semantic interoperability between different sources of distributed information [1, 2]. Several methods have been elaborated to semantically align multilingual ontologies. These methods can be generally split into two main categories direct and indirect matching approaches [3]. The approaches of the first category are based on external resources (i.e. translation) to align cross-lingual ontologies. However, the approaches of the second category are based on the composition of alignments such as the work proposed in [4] where the authors reuse the mappings between ontologies that already exist.

In this study, we consider the approaches of the first category, since we develop an approach which implements a direct strategy. However, there are many exciting questions regarding these approaches to address the multilingualism issue. These questions are as follows: (1) Which machine translation should be used, (2) which translation path should be considered and (3) which ontologies features and dictionaries can be exploited. In the following paragraphs, we describe the points mentioned above.

First, several translators have been developed to translate automatically the text from one natural language to another. We can mention for example: Google, Bing, SDL and Gengo translators. Each translator has its specific characteristics such as: number of source/target languages and execution time. However, selecting one or several translators (by combining them) remains an open problem. This choice is crucial in ”direct

approaches", since they apply a monolingual matching techniques in cross-lingual ontology mapping.

Second, the translation path also plays an important role to resolve the heterogeneity problem. Two translation paths can be considered, *(i)* either considering the translation path from one to another or *(ii)* selecting a pivot language which is often the English language. This choice highly depends on available sources (dictionaries, thesaurus, etc.) in different natural languages. Most matching systems consider the translation path using English as a pivot language due to available sources in English language.

Finally, in some cases, the results of a translation machine could be poor, however, to avoid this situation some ontology features can be exploited such Description Logics.

Most matching systems which implement a direct translation approach uses a well-known translators mentioned above. The current work uses also a direct matching approach. However, unlike existing approaches, it addresses the multilingualism challenge by using *(a)* the Yandex translator[1], *(b)* a translation into a pivot language after applying NLP techniques and *(c)* a similarity computation based on the categories of the words and synonyms.

The rest of the paper is organized as follows. First, in Section 2, we discuss the top systems that participated in the last editions of the multifarm track. In section 3 we describe the CroLOM system. Section 4 contains the experiment results. Finally, some concluding remarks and future work are presented in Section 5.

2 Related Work

In this section, we continue our previous work [5] by covering the main cross-lingual ontology matching systems that have participated in the last editions of the Multifarm track of OAEI evaluation campaign. These systems use a direct translation-based matching approach.

Table 1 summarizes the results of the top systems in the multifarm track.

The AUTOMSv2 system [14] uses a free Java API named WebTranslator[2] in order to solve the multi-language problem by translating label and properties in English language. The GOMMA system [15] uses a free translation API named "mymemory"[3] to automatically translate non-English terms. The WeSeE-Match system [16] translates the fragments, labels, and comments in English as a pivot language using the Bing[4] Search APIs translation capabilities. The WikiMatch system [17] employs the Google Translation API[5] for addressing multi-lingual ontologies. The CLONA system [18] translates the entities described in different natural languages into English as a pivot language using Bing translator. Then it uses Lucene search engine and WordNet to determine alignment candidates. The XMap system [7] uses an automatic translation

[1] https://translate.yandex.com/?lang=es-en&text=administrar&ncrnd=5317

[2] http://webtranslator.sourceforge.net/

[3] http://mymemory.translated.net/

[4] https://www.microsoft.com/en-us/translator/translatorapi.aspx

[5] http://code.google.com/apis/language/translate/overview.html

Table 1: Top systems in the multifarm track

OAEI	Top Systems	Multifarm Track	Precision	F-measure	Recall
2012	AUTOMSv2	without Arabic	.49	.36	.10
2012	WeSeE	without Arabic	.61	.41	.32
2012	GOMMA	without Arabic	.29	.31	.36
2012	WikiMatch	without Arabic	.34	.27	.23
2013	YAM++	without Arabic	0.51	0.40	0.36
2015	AML		0.53	0.51	0.50
2015	LogMap		0.75	0.41	0.29
2015	XMap		0.23	0.25	0.28
2015	CLONA		0.46	0.39	0.35

for obtaining correct matching pairs in multilingual ontology matching. The translation is done by querying Microsoft Translator for the full name. The AML system [8] uses an automatic translation module based on Microsoft Translator. The translation is done by querying Microsoft Translator for the full name (rather than word-by-word). To improve performance, AML stores locally all translation results in dictionary files, and queries the Translator only when no stored translation is found. The LogMap system that participated in the OAEI 2014 campaign used a multilingual module based on Google translate; however the new version of the LogMap system uses both Microsoft and Google translator APIs [11]. The YAM++ system [9] uses a multilingual translator based on Microsoft Bing to translate the annotations to English.

The multifarm track of OAEI 2015 contains our dataset in Arabic language (ADOM) [5, 6]. Contrary to AUTOMSv2, GOMMA, WeSeE-Match, WikiMatch and YAM++ systems which have not participated in OAEI2015; CLONA system participated for the first time in OAEI2015 initiative.

Except these systems, the results of XMap, LogMap and AML systems on multifarm track (includes Arabic) are slightly lower than previous editions of OAEI (i.e. in OAEI2014). According to the results obtained from the systems mentioned above, this is explained by the fact that the Arabic dataset brings an additional complexity to the multifarm track.

We have also observed that the best system (in all OAEI editions including this year) achieved an F-measure of 0.51. This is surprising, in spite of many research works that have been established in the field of multilingual ontology matching.

3 CroLOM: Cross-Lingual Ontology Matching System

We summarize the process of our approach to provide a general idea of the proposed solution. It consists in the following successive phases:

3.1 Extraction and Normalization

CroLOM extracts first the entities of the input ontologies. Then, it employs NLP techniques to normalize the entities described in different natural languages. Unlike existing approaches, we have applied lemmatization, stemming and stopword elimination for each natural language separately before translation step. First, for each language considered by multifarm, we have established the stop words of each language in order to eliminate them from entities labels. Second, we have developed morphological algorithms to obtain lemmatization of the entities words.

This step is important [6], since one of matchers used is (1) based on string comparison algorithm to compute similarity and (2) the categories of the words are stoked in lemma form.

3.2 Translation

Once the entities are normalized, CroLOM uses the Yandex translator in order to translate the entities described in different natural languages in English as a pivot language. After translation, CroLOM employs for the second time the normalization step in order to eliminate the stop words of the English language from entities labels.

We have mentioned before that the translation path and the used translator play important role to resolve the multilingualism heterogeneity problem. Our choice for the Yandex translator is justified by the fact that it is ranked as the 4th largest search engine in the world and it has not previously used to align multilingual ontologies. However, we have chosen English as a pivot language because there a lot dictionaries that are available in English language. These dictionaries could be exploited in order to improve our system in the future. In addition, to compute the similarity between entities, we have used dictionaries (word categories and WordNet) in English. Due to automatic translation, we have observed that some stop words can be appeared in translated entities. For this purpose, we have employed the normalization for the second time.

3.3 Similarity Computation

Once the translation and standardization are carried out, CroLOM applies first, a case conversion by converting all entities words in lower case then it passes to the similarity computation step. Unlike existing systems, which use well known matchers, we have developed a matcher which calculates the similarity between entities based on the categories of the Words, string-based algorithm and synonyms using Wordnet[7].

The matcher developed establishes a Cartesian product between the two entities words, then it returns the maximum similarity value using Levenshtein distance, similarity based on WordNet and similarity based on the categories of the words. The similarity based on the categories of the words has been adapted with some modification from the project ”Calculate Semantic Similarity” [8]. The project has been developed to

[6] This step allows to obtain good results such as the results of our previous work [19] (STRIM system) in instance matching.

[7] `http://wordnet.princeton.edu/`

[8] `https://sourceforge.net/projects/semantics/`

match sentences, however we have modified the code in order to compute similarity between words.

3.4 Alignment Identification

Finally, CroLOM applies a filter to select candidate correspondences which possess the maximum similarity value in each line of Cartesian product between entities. Then it applies a second a filter to identify the correspondences that possess similarity value upper than a given threshold.

4 Experimental Study

The results obtained by running our CroLOM system on multifarm tracks of OAEI 2016 evaluation campaign are obtained from the following website: `http://oaei.ontologymatching.org/2016/results/multifarm/index.html`.

Table 2: The Results of CroLOM System

System	Track	Precision	F-measure	Recall
CroLOM	Multifarm	0.55	0.36	0.28
LogMap	Multifarm	0.71	0.37	0.26
AML	Multifarm	0.56	0.40	0.34

The multifarm[13] track has been integrated in the Ontology Alignment Evaluation Initiative (OAEI) in 2012 with the goal of estimating and comparing different techniques and systems related to multilingual ontology alignment. From 2012 to 2014 the multifarm track contains conference ontologies[12] described in eight different languages (i.e., Chinese, Czech, Dutch, French, German, Portuguese, Russian, Spanish). However, in 2015 the multifarm includes the Arabic language.

The results obtained by our CroLOM system on multifarm are quite promising with F-measure equal to 36%. Comparing these results against the results of the systems which have participated in OAEI previous editions (Table 1), CroLOM with this first participation, is among the best systems with respect to F-measure. Regarding this year [Table 2], only AML (F-measure equals to 0.40) and LogMap (F-measure equals to 0.37) systems whose results are slightly better than CroLOM system.

The major drawback of CroLOM system is the execution time compared to other systems. We are working forward to identify this problem and improve our system.

5 Conclusion

In this paper, we have presented our CroLOM system, (not) yet another cross-lingual ontology matching system. CroLOM unlike existing approaches, applies first NLP techniques on each natural language before translation. Then, it uses the Yandex translator

in order to translate all entities in English as pivot language. Finally, CroLOM computes the similarity between translated entities based on the category of the words and WordNet.

As future challenges, we aim to (1) improving the quality results of our system and especially the execution time, (2) conduct a survey study that addresses all the issues mentioned above, (3) taking into account the indirect approaches.

References

1. A. Khiat and M. Benaissa, "A New Instance-Based Approach for Ontology Alignment". International Journal on Semantic Web and Information Systems (IJSWIS), Vol. 11, No. 3, ISSN 1683-3198, 2015.
2. A. Khiat and M. Benaissa, "Boosting Reasoning-Based Approach by Structural Metrics for Ontology Alignment". The Journal of Information Processing Systems (JIPS), 2015.
3. S Zhang and O. Bodenreider, "Alignment of Multiple Ontologies of Anatomy: Deriving Indirect Mappings from Direct Mappings to a Reference", AMIA 2005 Symposium Proceedings, 2005.
4. J. J. Jung, A. Hakansson, and R. H. . "Indirect Alignment between Multilingual Ontologies: A Case study of Korean and Swedish Ontologies," in Proceedings of the Third KES International Symposium on Agent and Multi-Agent Systems: Technologies and Applications,2009.
5. A. Khiat and M. Benaissa and Ernesto Jimnez-Ruiz "ADOM: arabic dataset for evaluating arabic and cross-lingual ontology alignment systems". In Proceedings of the 10th International Workshop on Ontology Matching co-located with the 14th International Semantic Web Conference (ISWC 2015), USA, 2015.
6. A. Khiat, G. Diallo, B. Yaman, E. Jimnez-Ruiz and M. Benaissa, "ABOM and ADOM: Arabic Datasets for the Ontology Alignment Evaluation Campaign". In Proceedings of the 14th International Conference (ODBASE 2015), Greece, 2015.
7. W. Djeddi, M. T.Khadir and S. Ben-Yahia, "XMap++ results for OAEI 2015". In Proceedings of the 10th International Workshop on Ontology Matching ISWC 2015, USA, 2015.
8. D. Faria, C. Martins, A. Nanavaty, D. Oliveira, B. Sowkarthiga, A. Taheri, C. Pesquita, F. Couto and I. Cruz , "AML results for OAEI 2015". In Proceedings of the 10th Workshop on Ontology Matching ISWC 2015, USA, 2015.
9. D. Ngo and Z. Bellahsene, "YAM++ results for OAEI 2013", In Proceedings of the 8th Workshop on Ontology Matching ISWC 2013, pp. 211-218, Australia, 2013.
10. A. Khiat and M. Benaissa, "AOT / AOTL results for OAEI 2014". In Proceedings of the 9th International Workshop on Ontology Matching ISWC 2014, pp. 113-119, Italy, 2014.
11. E. Jiménez-Ruiz, BC. Grau, A. Solimando, V. Cross, "LogMap family results for OAEI 2015". In Proceedings of the 10th Workshop on Ontology Matching ISWC 2015, USA, 2015.
12. O. Svab, V. Svatek, P. Berka, D. Rak and P. Tomasek, "OntoFarm: Towards an Experimental Collection of Parallel Ontologies", In: Poster Track of ISWC 2005, Galway, 2005.
13. C. Meilicke, R. Garca-Castro, F. Freitas, WR. Van Hage, E. Montiel-Ponsoda, R.R. De Azevedo, H. Stuckenschmidt, O. vb-Zamazal, V. Svtek and A. Tamilin, "MultiFarm: A benchmark for multilingual ontology matching". Web Semant. Sci. Serv. Agents World Wide Web. Vol. 15, pp. 62-68, 2012.
14. K. Kotis, A. Katasonov and J. Leino, "AUTOMSv2 results for OAEI 2012", In Proceedings of the 7th Workshop on Ontology Matching ISWC 2012, USA, 2012.
15. A. Gro, M. Hartung, T. Kirsten and E. Rahm, "GOMMA results for OAEI 2012", In Proceedings of the 7th Workshop on Ontology Matching ISWC 2012, USA, 2012.

16. H. Paulheim, "WeSeE-Match results for OEAI 2012", In Proceedings of the 7th Workshop on Ontology Matching ISWC 2012, USA, 2012.
17. S. Hertling and H. Paulheim, "WikiMatch results for OEAI 2012", In Proceedings of the 7th Workshop on Ontology Matching ISWC 2012, pp., USA, 2012.
18. M. El-Abdi, H. Souid, M. Kachroudi and S. Ben-Yahia, "CLONA results for OAEI 2015", In Proceedings of the 10th Workshop on Ontology Matching ISWC 2015, USA, 2015.
19. A. Khiat, M. Benaissa and M. A. Belfdhal, "STRIM results for OAEI 2015 instance matching evaluation". In Proceedings of the 10th International Workshop on Ontology Matching co-located with the 14th International Semantic Web Conference (ISWC 2015), USA, 2015.

CroMatcher - Results for OAEI 2016

Marko Gulić[1], Boris Vrdoljak[2], Marko Banek[3]

[1] Faculty of Maritime Studies, Rijeka, Croatia
marko.gulic@pfri.hr
[2] Faculty of Electrical Engineering and Computing, Zagreb, Croatia
boris.vrdoljak@fer.hr
[3] Ericsson Nikola Tesla d.d., Zagreb, Croatia
marko.banek@gmail.com

Abstract. Ontology matching plays an important role in the integration of heterogeneous data sources that are described by ontologies. In order to find correspondences between entities of different ontologies, a matching system has to be built. CroMatcher is an ontology matching system that consists of several string and structural basic matchers. As individual basic matcher computes similarity between entities using information obtained from one or more components of the entire ontology, all individual matching results need to be aggregated in order to achieve the better final matching results of compared ontologies. The CroMatcher system uses weighted aggregation method that automatically determines the weighting factors of each basic matchers considering quality of its matching result. Also, the system uses iterative final alignment method that selects appropriate correspondences between entities of compared ontologies from the aggregated matching results. This is the third time CroMatcher has been involved in the OAEI campaign. The system is upgraded by introducing two new basic matchers that improved the matching results at this OAEI campaign. CroMatcher achieved excellent matching results for the three ontology matching tracks in which it participated.

1. Presentation of the system

1.1. State, purpose, general statement

Ontology matching is the process of finding semantic relationships or correspondences between entities of different ontologies [1]. A matching system has to be built in order to determine correspondences between entities. CroMatcher is an ontology matching system in which the matching process is carried out automatically. It supports the matching between ontologies expressed in Web Ontology Language (OWL) [2] that is recommended by W3C (World Wide Web Consortium) [3] as an international standard for ontology representation. There are several string and structural basic matcher in CroMatcher system. Each basic matcher determines similarity between entities using

information obtained from one or more components of the compared ontologies, therefore matching results obtained by all basic matchers need to be aggregated in order to achieve the better final matching results. The string basic matchers, as well as the structural basic matchers, are related by parallel composition of basic matchers. First, the string basic matchers are executed. The results obtained by string basic matchers are automatically aggregated using our weighted aggregation method. These aggregated results are then used in the execution of the structural matchers as initial values of correspondences between entities. Again, the results obtained by structural basic matchers are aggregated using the weighted aggregation. Before the final alignment, the aggregated results of the string matchers and the aggregated results of the structural matchers are aggregated using the weighted aggregation. Eventually, the iterative final alignment method is executed in order to select appropriate correspondences between entities of compared ontologies from the aggregated matching results. The CroMatcher system that participated at OAEI 2016 is the third version of the system. Unlike the first two versions of the system [4, 5, 6] that have the identical architecture of matching process, a two new basic matchers are implemented into the newest version of the system. These matchers improved the matching results for the three ontology matching tracks in which CroMatcher participated in the OAEI campaign. CroMatcher is fully prepared for the *Benchmark* [7], *Anatomy* [8] and *Conference* [9] ontology tracks and produces excellent results for these tracks.

1.2. Specific techniques used

In this section, the architecture of CroMatcher system as well as the main components will be briefly presented. As already mentioned, this version of CroMatcher (OAEI campaign 2016) has two more string basic matchers implemented than last version presented in [6]. Like last year, some basic matchers are modified to speed up the matching process for *Anatomy* ontology matching track that contains a large number of entities. The system activates the lite version of these basic matchers if the compared ontologies contain more than thousand entities. The workflow and the main components of the system can be seen in the Figure 1. The CroMatcher consists of the following components:

1. **Ontology data processing** - Initial step of an ontology matching process is the extraction of information about entities within compared ontologies. After the extraction of data, the matching process starts to determine correspondences between entities of compared ontologies.
2. **String basic matchers** – determine correspondences between entities considering the character arrays (strings) that describe compared entities.
   - *Annotation matcher* – determines the correspondence between entities by comparing the strings obtained from entities' IDs and annotations using n-gram similarity [1].
   - *Profile matcher* - determines the correspondence between entities by comparing the textual profiles of two entities. The methods TF/IDF [10] and cosine similarity [11] are used to calculate similarity between these textual profiles.

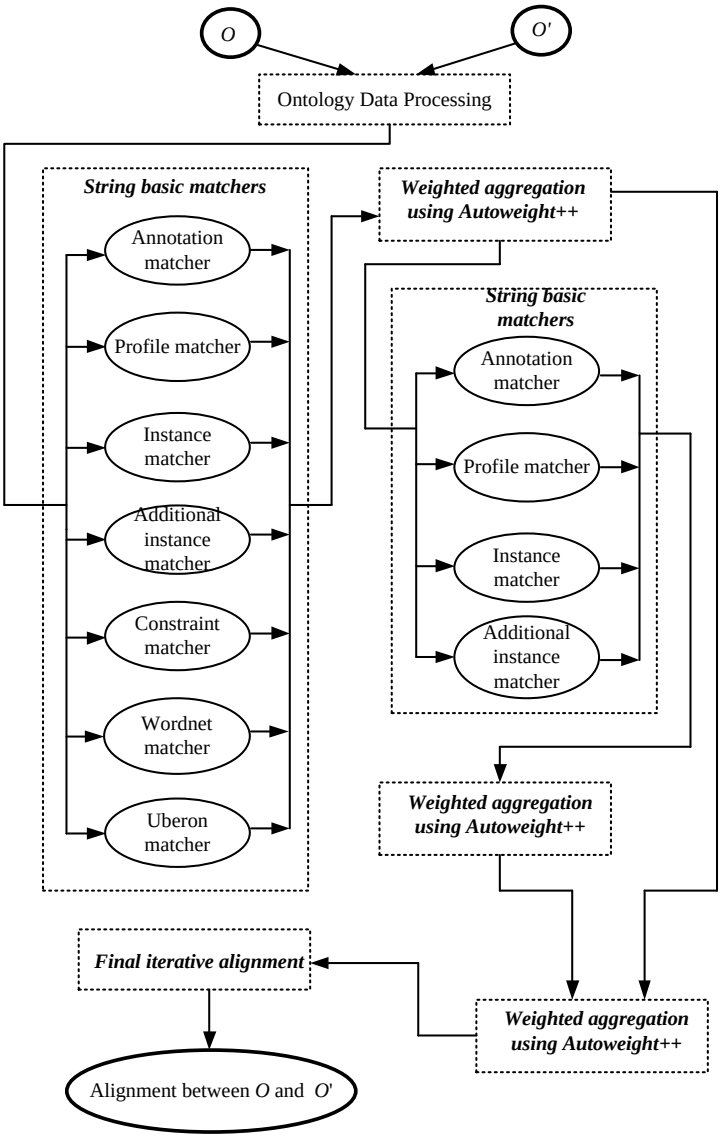


Figure 1. Workflow and the main components of CroMatcher

The textual profile is a large text that describes an entity. A content of textual profile is precisely defined in [6]. Considering the size of textual profile, the matching process is slow because the TF/IDF method has to retrieve the text of all entities before starting comparing two entities. When a target ontology contains more than 1000 entities, a modified *Profile matcher* is activated. This matcher determines correspondences using the fast string metric described in [12]. The results of this modified *Profile matcher* are a bit worse than results of

the *Profile matcher* that uses TF/IDF method but it is acceptable considering the faster matching process.

- *Instance matcher* – determines the correspondence between instances of compared entities by using the methods TF/IDF and cosine similarity.
- *Additional instance matcher* - determines the correspondence between additional instances of compared entities by using the methods TF/IDF and cosine similarity. Additional instances contain not only the instances of compared entities but also the instances of entities that are related to the compared entities.
- *Constraint matcher* – determines the correspondence between entities by comparing various features of compared entities (number of object and data properties, cardinality constraints…).
- *WordNet matcher* – a newly implemented matcher. It determines the correspondence between entities by comparing the strings obtained from entities' IDs and annotations using WordNet [13]. WordNet is a large lexical database of English. The WordNet matcher can find similarities between two tokens of compared strings considering the relations (synonyms, hypernyms etc.) defined between these tokens within WordNet. The deficiency of the previous systems was its inability to recognize these language relations.
- *Uberon matcher* – a newly implemented matcher. It determines the correspondence between entities by using the mediator ontology Uberon (Uber Anatomy Ontology) [14]. This matcher is used for the Anatomy matching track. Uberon is an integrated cross-species ontology covering anatomical structures in animals. Hence, Uberon contains a lot of information about the anatomy, therefore it is very helpful when matching ontologies of the Anatomy track.

3. **Structural basic matchers** – determine correspondences between entities by comparing their relations with other entities. All these matchers are executed iteratively. Like in the previous OAEI campaign, in order to speed up the matching process, we made modified structural matchers when comparing ontologies that contain more than 1000 entities. When ontologies contain more than 1000 entities, all structural matchers are executed just once. Modified matchers decreases the quality of matching process but speed up the process.
    - *SuperEntity matcher* – determines the correspondence between entities by comparing the mutual correspondences between their parent entities.
    - *SubEntity matcher* – determines the correspondence between entities by comparing the mutual correspondences between their children entities.
    - *Domain matcher* – this matcher has two modes, one for calculating similarity between class entities and the other one for property entities. First version determines correspondences between classes by comparing all the properties that have the compared classes as their domains. Second version determines correspondences between properties by comparing the classes defined as the domain of the considered properties.

- *Range matcher* – this matcher determines correspondences only between two property entities by comparing the classes defined as the range of the considered properties.

The procedure of executing these structural matchers is described in [6] in detail.

4. **Weighted aggregation using Autoweight++ method** – As stated before, CroMatcher system executes the weighted aggregation three times during the matching process. In this system, we have introduced the Weighted aggregation that uses a new method for automatically determining the weighting factors of basic matchers. This new method determines the weighting factors of basic matchers according to the importance of the highest correspondences found within the matching results of each basic matcher. A correspondence between two entities e_i and e_j' is the highest correspondence if and only if it has higher value than any other correspondence of either e_i or e_j' with some other entity. The importance of each highest correspondence found within the matching results of a particular basic matcher is calculated comparing the complete results of this basic matcher, without taking into consideration the matching results of other basic matchers, which is the case in Autoweight++ method [6] that is used in our previous version of the system (CroMatcher 2015).
5. **Final alignment** – The final alignment method iteratively selects relevant correspondences between entities of compared ontologies. This method is presented in detail in [6].

2. Results

2.1. Benchmarks

In OAEI 2016 campaign, the *Benchmark* ontology track includes a well-known *biblio* test case. In Table 1. the results for biblio test case achieved in OAEI campaigns 2015 and 2016 by running the CroMatcher ontology system are presented.

Table 1. The matching results of CroMatcher system for Benchmark biblio test set

OAEI	Recall	Precision	F-Measure
2015	0.82	0.94	0.88
2016	0.83	0.96	0.89

As CroMatcher system already has achieved very good results, the improvement of the new version of the system is small, but significant. Our system achieved the best results in the *Benchmark* ontology track together with the Lily system. The introduction of the new basic matcher based on WordNet and the modified Weighted aggregation method has led to better matching results.

2.2. *Anatomy*

The Anatomy ontology track consists of two large ontologies (*mouse.owl* and *human.owl*) that have to be matched. These ontologies represent a formal description of human and mouse anatomies. In Table 2. the results for *Anatomy* ontology track achieved in OAEI campaigns 2015 and 2016 by running the CroMatcher ontology system are presented.

Table 2. The matching results of CroMatcher system for Anatomy track

OAEI	Recall	Precision	F-Measure	Time (s)
2015	0.814	0.914	0.861	569
2016	0.902	0.949	0.925	573

CroMatcher significantly improved the matching results for *Anatomy* ontology track considering the previous results of this system. The results are improved due to introducing the *Uberon* string matcher. As stated before, Uberon is an integrated cross-species ontology covering anatomical structures in animals, therefore it is very useful when determining correspondences between ontologies of the *Anatomy* track. CroMatcher achieved the second best results in the *Anatomy* track. Only the AML system has better matching results. Furthermore, only CroMatcher and AML have the F-measure higher than 0.9. However, a remaining challenge for future work is to speed up the execution of the complete system. The focus will be on the execution performance of the iterative structural matchers.

2.3. *Conference*

Conference ontology track contains 16 similar ontologies that all describe organization of a conference. The systems are evaluated according to three different modes of evaluation of which the first mode (crisp reference alignments) is the most comprehensive one. Furthermore, there exist three variants of crisp reference alignments: ra1 (the original reference alignment), ra2 (the entailed reference alignment generated as a transitive closure computed on the ra1) and ra3 (the violation free version of ra2). Each of these three variants consists of three different tests according to three different alignments between 16 conference ontologies: M1 (contains classes only), M2 (contains properties only) and M3 (contains classes and properties together). Hence, the evaluation mode crisp reference alignments produces nine different evaluation tests for matching systems: ra1-M1, ra1-M2… ra3-M3. In this section, we will present the results of these nine different evaluation tests according to standard F-measure (the harmonic mean of precision and recall). CroMatcher system produces the best results for three tests (ra1-M1, ra2-M1 and ra3-M1). For two tests (ra2-M3 and ra3-M3), our system also produces the best results alongside the AML system. Furthermore, for remained four tests (ra1-M2, ra1-M3, ra1-M2 and ra3-M2), our system produces the second best result behind the AML system. Considering the overall results of the previous and the current version of CroMatcher (Table 3.), it can be seen that we made a great improvement in matching ontologies of *Conference* track.

Table 3. The matching results of CroMatcher system for Conference track

OAEI	Recall	Precision	F-Measure
2015	0.46	0.56	0.51
2016	0.64	0.77	0.70

2.4. *Other ontology tracks*

This year, we have not participated in other ontology tracks because we did not prepare our system for these tracks. Next year, we will try to improve our system to be able to obtain the considerable matching results for more ontology tracks than this year.

3. General comments

OAEI campaign provides not only the evaluation of our system but also the comparison with other state-of-the-art system. We consider that OAEI evaluation of the ontology matching systems is the most authoritative criterion for comparing various matching system because the complete evaluation is performed publicly by the OAEI organizers. There are also many different ontology tracks and we think that these tracks can help anybody to make additional improvements of matching system.

3.1. Comments on the results

CroMatcher achieved great matching results in the ontology tracks (Benchmarks, Anatomy, Conference) for which it was prepared. Considering the results of each individual track, our system achieved the best or the second best matching results.

3.2 Discussions on the way to improve the proposed system

We will try to solve the problem with the slow iterative structural matcher in order to improve the matching process when comparing large ontologies. Also, we will have to store the data about the entities in a separate file instead of java objects in order to reduce the usage of memory in the system. Furthermore, we will try to prepare the system for all OAEI ontology tracks.

4. Conclusion

The third version of the CroMatcher ontology matching system and its results in the OAEI campaign were presented in this paper. As in the previous versions of the system, CroMatcher consists of several string and structural basic matchers. The Autoweight++ method is used to aggregate the results obtained by these matchers. At the end of the matching process, the iterative final alignment method is executed. In this version of

the system, two new string matchers are introduced: *WordNet* matcher and *Uberon* matcher. *WordNet* matcher can find similarities between entities considering the language relations like synonyms, hypernyms etc. Uberon is an integrated cross-species ontology covering anatomical structures in animals. Considering the *Anatomy* track, Uberon is very useful when finding correspondences between ontologies of this track. The evaluation results show that CroMatcher achieved great results for *Benchmark*, *Anatomy* and *Conference* tracks for which it was prepared. According to the results of these three tracks, CroMatcher achieved better matching results than last year. Furthermore, there is still room for improvement considering the speed of the matching process. Also, we will try to prepare the system for all ontology tracks in the OAEI campaign next year.

References

1. J. Euzenat, P. Shvaiko, Ontology matching, 2nd Edition, Springer-Verlag, Heidelberg (DE), 2013.
2. G. Antoniou, F. van Harmelen, A Semantic Web Primer, MIT Press, 2004.
3. World wide web consortium, http://www.w3.org/, accessed: 25-09-2016.
4. M. Gulić, B. Vrdoljak, CroMatcher - results for OAEI 2013, in: P. Shvaiko, J. Euzenat, K. Srinivas, M. Mao, E. Jiménez-Ruiz (Eds.), Proc. of the 8th Int. Workshop on Ontology Matching co-located with the 12th Int. Semantic Web Conf. (ISWC 2013), Sydney, Australia, October 21, 2013, Vol. 1111 of CEUR Workshop Proceedings, CEUR-WS.org, pp. 117–122.
5. M. Gulić, B. Vrdoljak, M. Banek, CroMatcher results for OAEI 2015, in: P. Shvaiko, J. Euzenat, E. Jiménez-Ruiz, M. Cheatham, O. Hassanzadeh (Eds.), Proc. of the 10th Int. Workshop on Ontology Matching collocated with the 14th Int. Semantic Web Conf. (ISWC 2015), Bethlehem, PA, USA, October 12, 2015, Vol. 1545 of CEUR Workshop Proceedings, CEUR-WS.org, 2016, pp. 130–135.
6. M. Gulić, B. Vrdoljak, M. Banek, CroMatcher: An ontology matching system based on automated weighted aggregation and iterative final alignment, Journal of Web Semantics: Science, Services and Agents on the World Wide Web (2016), http://dx.doi.org/10.1016/j.websem.2016.09.001
7. J. Euzenat, M.-E. Rosoiu, C. Trojahn, Ontology matching benchmarks: generation, stability, and discriminability, Journal of Web Semantics, Vol 21 (2013) 30–48.
8. O. Bodenreider, T.F. Hayamizu, M. Ringwald., S. de Coronado, S. Zhang, Of mice and men: Aligning mouse and human anatomies, AMIA Annu Symp Proc, 2005, pp. 61-65
9. M. Cheatham, P. Hitzler, Conference v2.0: An Uncertain Version of the OAEI Conference Benchmark. International Semantic Web Conference (2), 2014, pp. 33-48.
10. G. Salton, M.H. McGill, Introduction to Modern Information Retrieval. McGraw-Hill, New York, 1983
11. R. Baeza-Yates, B. Ribeiro-Neto, Modern Informationl Retrieval. Addison-Wesley, Boston, 1999
12. Strike a match, http://www.catalysoft.com/articles/strikeamatch.html, accessed 25-09-2016
13. G. A. Miller, WordNet: A Lexical Database for English. Communications of the ACM, 38(11):39–41, 1995.
14. C. J. Mungall, C. Torniai, G. V. Gkoutos, S. Lewis, and M. A. Haendel. Uberon, an Integrative Multi-species Anatomy Ontology. Genome Biology, 13(1): R5, 2012.
15. M. Gulić, I. Magdalenić, B. Vrdoljak, Automatically specifying parallel composition of matchers in ontology matching process, Communications in Computer and Information Science, vol. 240, 2011, pp. 22–33.

DisMatch results for OAEI 2016

Maciej Rybiński *, María del Mar Roldán-García, José García-Nieto, and José F. Aldana-Montes

Dept. de Lenguajes y Ciencias de la Computación, University of Malaga, ETSI Informática, Campus de Teatinos, Malaga - 29071, Spain
maciek.rybinski@lcc.uma.es, mmar@lcc.uma.es, jnieto@lcc.uma.es, jfam@lcc.uma.es

Abstract. DisMatch is an experimental ontology matching system based on the use of corpus based distributional measure for approximating semantic relatedness. Through the use of a domain-related corpus, the measure can be applied to a problem focused on the domain of the corpus, here being the Disease and Phenotype track. In this paper, we aim to briefly present the proposed approach and the results obtained in the evaluation, as well as some early conclusions regarding the performance of DisMatch.

Keywords: Ontology Matching, Bench-marking, Lexical Semantic Relatedness

1 Presentation of the system

1.1 State, purpose, general statement

It has been demonstrated that corpus based measures can be used to successfully approximate human judgment, w.r.t. semantic relatedness between pairs of concepts [1,3,4]. DisMatch is an experimental system built for the purpose of evaluating the applicability of a state-of-the-art domain-focused corpus based measure of semantic relatedness, to a task of ontology alignment.

For a pair of ontologies, DisMatch calculates the matrix of semantic relatedness between labels representing their concepts. It then uses this matrix as the input for the classic algorithm of Similarity Flooding [2], in order to incorporate the taxonomic information into our final results.

1.2 Specific techniques used

The workflow of DisMatch can be broken down into the following steps:

1. Preprocessing: extraction of the taxonomies and labels of the concepts.
2. Assigning distributional representations to the concepts of the ontologies

* Corresponding author maciek.rybinski@lcc.uma.es

3. Calculating the semantic relatedness for the pairs of concepts of the respective ontologies
4. Calculating the similarity propagation given the taxonomies and initial relatedness scores (SimFlood)
5. Calculating the final similarity scores
6. Filtering

In step (2), we use vector based representations of an ESA (Explicit Semantic Analysis [1]) style approach adapted to the biomedical domain related use. The representations are created for inputs that are the labels of individual concepts. The distributional representations are obtained through a combined use of Wikipedia and a domain-focused corpus of scientific documents, i.e. Medline.

In step (3), we use the vectors from step (2) to calculate the relatedness approximation as the cosine similarity of these vectors. To calculate the similarity propagation in step (4), we use the very basic version of the algorithm applied to the taxonomic structures. We do however restrict the propagation graph size by not including the nodes that do not surpass a certain minimal initial relatedness threshold.

We calculate the final similarity scores (step 5) as an average between the initial scores (semantic relatedness) and the similarity propagation output. This gives more importance to the relatedness score (which is the point of our experiment), and also caters for cases in which Similarity Flooding is poorly applicable.

The filtering is done by: i) accepting only a maximal number of candidate matches per node of an ontology; ii) eliminating candidate matches below a certain similarity threshold; iii) accepting a globally maximal number of candidate matches.

1.3 Adaptations made for the evaluation

No specific adaptations were made for the experiments, apart from minor changes of the filtering parameters (i.e. the global number of candidate matches accepted in the final alignment).

1.4 Link to the set of provided alignments

The set of provided alignments is available in URL `http://bit.ly/2dPA9H5`

2 Results of the Disease and Phenotype track

DisMatch has been evaluated in both tasks of the Disease and Phenotype track: HP-MP (alignment of Human Phenotype Ontology with Mammalian Phenotype Ontology) and DOID-ORDO (alignment of Human Disease Ontology with Orphanet Rare Disease Ontology). A summary of results is reported in the Official site of OAEI 2016::Disease and Phenotype Track[1].

[1] In URL http://oaei.ontologymatching.org/2016/results/phenotype/.

Table 1. Unique mappings in the HP-MP task

OM Algorithm	Unique Equivalence Mappings	Precision (Manual Assessment)	Positive Contribution (TP)	Negative Contribution (FP)
AML	122	0.8667	8.63%	1.33%
DisMatch	**291**	0.8333	**19.80%**	3.96%
FCA_Map	26	0.9615	2.04%	0.08%
LogMap	130	0.9330	9.90%	0.71%
LogMapLite	0	0.0000	0.00%	0.00%
LogMapBio	176	0.9330	13.40%	0.96%
LYAM++	226	0.7000	12.91%	5.53%
PhenoMF	89	1.0000	7.27%	0.00%
PhenoMM	85	1.0000	6.94%	0.00%
PhenoMP	80	1.0000	6.53%	0.00%
XMap	0	0.0000	0.00%	0.00%
Totals	**1225**		**87.42%**	**12.58%**

It can be observed that the results of DisMatch are relatively far off the silver standard created in the evaluation process. We believe that this is largely due to setting up the system with parameters that resulted in overly strict filtering that created a relatively low number of mappings. In turn, the low number of mappings led to poor recall, both in the silver standard evaluation and w.r.t. the set of manually created mappings.

The precision of DisMatch in the HP-MP alignment looks quite promising, especially if we consider the number of unique alignments produced by the system. Out of the total of 644 mappings, 353 mappings are confirmed by at least one another system (thus falling into 'correct' category in the silver standard 2). Out of these 353, 293 are confirmed by at least 2 other systems ('correct' in silver standard 3). The remaining 291 mapping are unique to DisMatch. Table 1 presents an overview of unique mappings produced by the respective systems. The precision of the unique mappings produced by Dismatch is estimated at 0.8333, which accounts for a large portion of unique and correct mappings discovered by our system. In this regard, the proposed approach obtained the highest percentage of positive contribution (19.80%), with a relatively low negative contribution (3.96%).

In the case of DOID-ORDO alignment, the performance of our system is limited, as it is affected not only by the low recall related to the poor parameter selection, but also by the inability of our structural mapping component to cope with the structure of the Orphanet ontology. This shortcoming will be addressed in the future versions of DisMatch. Nonetheless, as shown in Table 2, even in this setting, the system managed to produce a considerable number (estimated 40% of 259 is > 100) of correct unique mappings.

Table 2. Unique mappings in the DOID-ORDO task

OM Algorithm	Unique Equivalence Mappings	Precision (Manual Assessment)	Positive Contribution (TP)	Negative Contribution (FP)
AML	308	0.8667	30.40%	4.68%
DisMatch	259	0.4000	11.80%	17.70%
FCA_Map	61	0.8330	5.79%	1.16%
LogMap	80	0.9000	8.20%	0.91%
LogMapLite	7	0.5000	0.40%	0.40%
LogMapBio	144	0.9667	15.85%	0.55%
LYAM++	0	0.0000	0.00%	0.00%
PhenoMF	3	1.0000	0.34%	0.00%
PhenoMM	0	0.0000	0.00%	0.00%
PhenoMP	0	0.0000	0.00%	0.00%
XMap	16	0.5625	1.03%	0.80%
Totals	**878**		**87.42%**	**12.58%**

3 General comments

Relatedness measure seems to capture non-trivial matches better than, for example, string edit distance. At the same time, it still works for the trivial cases, as common words will generate similar distributional representations. The main strength of DisMatch (and its distributional semantic relatedness component) is its ability of finding non-trivial mappings, which seems to be confirmed by the number of unique correct matches generated by the system (and the unique-to-total mappings ratio).

Nonetheless, the structural matching strategy still seems to be an important component of the system, as the relatedness matcher itself will, for example, generate high confidence matches for inputs, such as 'X syndrome' and 'Y syndrome', if X and Y are very rare in the background corpus. The importance of the structural matching step seems to be consistent with the performance gap between HP-MP (where the structural matcher worked) and DOID-ORDO (where it did not work properly) cases.

We believe that DisMatch could be improved substantially through improving the relatedness-structure matching combination, i.e. by employing a better suited structural matcher. Furthermore, our current structural matching strategy relied solely on strictly taxonomic relationships, which is not always enough (i.e. in the case of the OrphaNet ontology).

Furthermore, semantic relatedness module generates candidate mappings that are not necessarily 'equivalent', as the measure does not distinguish between the possible relationship types. It is worth considering adding an additional 'prediction' module to provide a classification output of the relationship type of the mappings.

Moreover, when it comes to improving the performance of the relatedness module itself, it seems that the measure provides more accurate results for

shorter input texts. This points to two possible improvements: (a) in finding a better suited compositional approach for the lexical relatedness measure, or (b) in using shorter inputs (possibly through synonym properties of the ontologies to be aligned).

4 Conclusions

The results obtained with the DisMatch system show enough promise to continue the experiments with corpus-based distributional relatedness measures applied to the problem of ontology alignment. We believe, that our focus should now be on providing an optimal set of additional components around the relatedness measure. In addition, we expect that tuning of the filtering parameters will lead the proposed system to reach higher precision with respect to silver standards.

Acknowledgments

This work has been partially funded by Grants TIN2014-58304-R (Spanish Ministry of Education and Science) and P11-TIC-7529 (Innovation, Science and Enterprise Ministry of the regional government of the Junta de Andalucía) and P12-TIC-1519 (Plan Andaluz de Investigación, Desarrollo e Innovación). José Garía-Nieto is recipient of a Post-Doctoral fellowship of "Captación de Talento para la Investigación" at Universidad de Málaga.

References

1. Evgeniy Gabrilovich and Shaul Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *IJCAI*, volume 7, pages 1606–1611, 2007.
2. Sergey Melnik, Hector Garcia-Molina, and Erhard Rahm. Similarity flooding: A versatile graph matching algorithm and its application to schema matching. In *Data Engineering, 2002. Proceedings. 18th International Conference on*, pages 117–128. IEEE, 2002.
3. Ted Pedersen, Serguei V S Pakhomov, Siddharth Patwardhan, and Christopher G Chute. Measures of semantic similarity and relatedness in the biomedical domain. *Journal of biomedical informatics*, 40(3):288–299, 2007.
4. Maciej Rybinski and José F Aldana-Montes. Calculating semantic relatedness for biomedical use in a knowledge-poor environment. *BMC bioinformatics*, 15(Suppl 14):S2, 2014.

DKP-AOM: results for OAEI 2016

Muhammad Fahad
Centre Scientifique et Technique du Bâtiment *(CSTB),*
290 Route des Lucioles, Sophia-Antipolis, *FRANCE*
firstname.lastname@cstb.fr

Abstract
In this paper, we present the results obtained by our DKP-AOM system within the OAEI 2016 campaign. DKP-AOM is an ontology merging tool designed to merge heterogeneous ontologies. In OAEI, we have participated with its ontology mapping component which serves as a basic module capable of matching large scale ontologies before their merging. This is our second successful participation in the OAEI 2016 campaign and first in the Process Model Matching track of OAEI. DKP-AOM is participating with two versions (DKP-AOM and DKP-AOM_lite). The reference alignments contain correspondences between instances of the class task as well as some correspondences between events. In the lite version of DKP, it does not match classes with the events, as it is of natural semantics that events should not be mapped on classes and vice versa. Therefore, we designed our system with two variants. But, our DKP-AOM system identifies cases where tasks are matched on events (where it makes sense). This is the only difference between two variant, hence for other tracks these two variants produce the same results. In this track, we can see its competitive results in the evaluation initiative among other reputed systems. Finally, we discuss some future work towards the development of DKP-AOM.

Keywords: Ontology matching, Ontology merging, disjoint knowledge, inconsistency, incompleteness, inconciseness, validation of mappings, verification of merged ontology

1 Presentation of the System

Ontology merging is a process of building a new ontology from two or more existing ontologies with overlapping parts. The merged ontology can be either virtual or physical, but must be consistent, coherent and include all the information from the source ontologies [1]. Ontology merging is based on two primary steps. Firstly, the source ontologies are looked-up for correspondences between them. Secondly, duplicate-free and conflict-free union of source ontologies is achieved based on the established correspondences [2]. The first part mainly comes under the ontology matching, whereas the second part targets to achieve the merged ontology based on the results of the first part, i.e., mappings between source ontologies. To produce accurate merged ontology, there should be some mechanism to avoid erroneous intermediate mappings and also to merge them in such a way that produces consistent, complete and coherent merged ontology. There are many hurdles that come

across in the generation of desired merged output. Firstly, ontological errors and design anomalies that can occur in the source ontologies detract from reasoning and inference mechanisms, and create bottleneck in their integration tasks [3]. In addition, conceptualization of domain, explication and modeling of knowledge over ontologies and semantic heterogeneities make their integration more difficult [4]. Secondly, even if the individual ontologies are free from errors, some of the identified mappings lead towards the erroneous situations producing several types of errors in the merged ontology [5]. For building an effective ontology merging algorithm, it is essential to incorporate ontological error checking during the validation of ontology mapping process and the verification of merged ontology to attain the accuracy of resultant output.

In order to meet the above mentioned challenges for the ontology merging research, we proposed semi-automatic DKP-OM system implemented in Jena framework for the merging of heterogeneous ontologies with the human user expert [6]. Later, we released a fully Automatic Ontology Merging (AOM) system named DKP-AOM implemented in OWLAPI 3 [7]. The name DKP comes from the concept of performing *Disjoint Knowledge Analysis (DKA)* and *Disjoint Knowledge Preservation (DKP)* during the merging process. Disjoint Knowledge Analysis plays a vital role in controlling the search space for finding similarities between source ontologies. Look-up within disjoint partitions of source ontologies significantly reduces the time complexity of the mapping phase. Disjoint Knowledge Preservation in the merged ontology helps to preserve disjoint axioms in the sub-hierarchies of merged ontology to avoid incompleteness in the resultant merged ontology. In this way, it also pin-points different conflicts between source ontologies based on disjoint axioms in the source ontologies and detects inconsistent mappings. Computed mappings that lead in many cases to a large number of unsatisfiable classes are eliminated so the resultant merged ontology should not suffer from inconsistencies. The next sub-sections provide more details about DKP-AOM and then discuss our results of OAEI participation.

1.1 Adaptations made for the evaluation

As you read above, DKP is an automatically merging system. Therefore it was developed based on user GUIs such as source ontology trees for display, visual alignments between ontologies, merged ontology tree, etc. The original version of DKP has changed and these visual components are removed so that it can participate under the seals platform. However, still it needs proper clean-up to improve its runtime for the future OAEI participations.

1.2 Link to the system

Various versions of my system can be found at my personal site: http://sites.google.com/site/mhdfahad under *plugins* tab. The mapping system is separated from the merging system, and can be downloaded according to needs. For the merging of ontologies, use the same command of seals platform with –o following three paths, two for source ontologies and one for the output merged ontology. As a result of this command, a list of ontology mappings and a resultant merged ontology are produced.

2 Results

In order to show the efficiency and effectiveness of our system, this year we participated in Process Modeling track. The results are very encouraging provided by the OAEI 2016 campaign as our system is acceptable and comparable with other participants, and are discussed in the following subsections.

2.1 Process Model Matching

This track concerns with the task of matching process models, originally represented in BPML. These models have been converted to an ontological representation. The resulting matching task is a special case of an interesting instance matching problem. Organizers have 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. By offering this track, OAEI hope to gain insights in how far ontology matching systems are capable of solving the more specific problem of matching process models. The collection consists of 9 models ("Cologne", "Frankfurt", "FU_Berlin", "Hohenheim", "IIS_Erlangen", "Muenster", "Potsdam", "TU_Munich", "Wuerzburg"), for each pair exists an alignment in the gold standard. However, there is only an alignment named "Cologne-Frankfurt.rdf" and no alignment "Frankfurt-Cologne.rdf". This is the first time DKP-AOM is participating in this track.

We have participated with two versions of DKP with some differences. The reference alignments contain correspondences between instances of the class task as well as some correspondences between events. In the lite version, we have not matched classes with the events, as it is of natural semantics that events should not be mapped on classes and vice versa. Therefore, we separated our system with two variants. Our DKP-AOM system identifies some cases where tasks are matched on events (where it makes sense). But in its lite version, we did not add this functionality. For an example, consider a scenario where:

BPMN1: Task (Receive Rejection)
BPMN2: Event (Rejected)

Although in real world for someone, it has the impression that the "Rejected-Event" has within the workflow the same semantics as the "Receive rejection Task". In these cases, its about getting informed, receiving a message. That is why in this case an event and a task are used to model the same real world event/task. Indeed its even hard to say, if this is an event or a task. This leads to have two variant of DKP in the participation. The following table 1 shows the comparative analysis of DKP-AOM with other systems participated in the process matching track.

Table 1. Comparative analysis of DKP-AOM with other systems [10]

Participants			Standard				Probabilistic			
Name	OAEI/PMMC	Size	P	R	FM	Ranking	ProP	ProR	ProFM	Ranking
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
Know-Match-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

Participants of the Process Model Matching Contest are depicted in grey font, while OAEI participants are shown in black font [for details see ref 10]. The OAEI participants are ranked on position 1, 8/9 and 11 with an overall number of 16 systems listed in the table. In the probabilistic evaluation, however, the OAEI participants (AML, LogMap, DKP, DKP*) gain position 2, 3, 9 and 10, respectively. Our system DKP generates mediocre results, this indicates 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.

Table 2 presents the results obtained by DKP-AOM on the PM track of OAEI campaign 2016.

Test Case ID	Precision	Recall	F-measure	Test Case ID	Precision	Recall	F-measure
Cologne-FU_Berlin	1	1	1	Frankfurt-Potsdam	0.4	1	0.571
Cologne-Frankfurt	0.889	1	0.941	Frankfurt-TU_Munich	0.857	1	0.923
Cologne-Hohenheim	0	0	0	Frankfurt-Wuerzburg	0.5	0.333	0.4
Cologne-IIS_Erlangen	0.5	1	0.667	Hohenheim-IIS_Erlangen	0.5	0.2	0.286
Cologne-Muenster	0.5	1	0.667	Hohenheim-Muenster	1	0.375	0.545
Cologne-Potsdam	0.5	1	0.667	Hohenheim-Potsdam	0	0	0
Cologne-TU_Munich	0.692	1	0.818	Hohenheim-TU_Munich	0	0	0
Cologne-Wuerzburg	0.5	0.333	0.4	Hohenheim-Wuerzburg	1	0.25	0.4
FU_Berlin-Hohenheim	0	0	0	IIS_Erlangen-Muenster	0.714	0.385	0.5
FU_Berlin-IIS_Erlangen	1	0.857	0.923	IIS_Erlangen-Potsdam	0.857	0.857	0.857
FU_Berlin-Muenster	1	0.5	0.667	IIS_Erlangen-TU_Munich	0.5	0.222	0.307
FU_Berlin-Potsdam	1	0.929	0.963	IIS_Erlangen-Wuerzburg	1	0.333	0.5
FU_Berlin-TU_Munich	0.5	0.333	0.4	Muenster-Potsdam	0.714	0.455	0.556
FU_Berlin-Wuerzburg	0.667	0.333	0.444	Muenster-TU_Munich	0.5	0.222	0.307
Frankfurt-FU_Berlin	0.4	1	0.571	Muenster-Wuerzburg	1	0.333	0.5
Frankfurt-Hohenheim	0	0	0	Potsdam-TU_Munich	0.5	0.333	0.4
Frankfurt-IIS_Erlangen	0.4	1	0.571	Potsdam-Wuerzburg	0.667	0.333	0.444
Frankfurt-Muenster	0.2	1	0.333	TU_Munich-Wuerzburg	0	0	0
Global	**0.718**	**0.547**	**0.621**				

Table 2. presents the results obtained by running DKP-AOM

2.2 Conference

The goal of conference track is to find alignments among 16 ontologies relatively smaller in size (between 14 and 140 entities) but rich in semantic heterogeneities about the conference organization domain. As a result, Alignments are evaluated automatically against reference alignments. Therefore, it is very interesting to measure the Precision, Recall and F-measure of our system and also does a comparison between existing systems to see their performance on real world datasets. Table 2 presents

the results obtained by running DKP-AOM on the Conference track of OAEI campaign 2016. Our system DKP-AOM has produced very competitive results among top ranked systems. Our precision measure is significantly high, recall is good giving comparable F-measure value to depict a real effort towards detecting heterogeneities for the goal of ontology matching.

Matcher	Runtime	Precision	F-Measure	Recall
DKP-AOM	9913	0.844	0.626	0.498

Table 2. DKP-AOM results on conference track ontologies

3 Conclusion and Future Directions

The participation of DKP-AOM in OAEI 2016 is a success in the Process Model Matching track. Our aim was to implement BPMN model matching; therefore, we have only implemented processing model strategy in our last version of DKP-AOM that participated in 2015. Therefore, it produces (more or less) the same output in the evaluation tracks as OAEI 2015, hence we haven't discuss output on other tracks. We can see DKP-AOM has produced competitive results in the evaluation Process Model initiative among other reputed systems.

References

1. Bruijn, J.d., Ehrig, M., Feier, C., Martín-Recuerda, F., Scharffe, F., and Weiten., M., Ontology mediation, merging and aligning. In Semantic Web Technologies. Wiley 2006
2. Euzenat, J., and Shvaiko, P., Ontology Matching. Springer, 2007, ISBN 978-3-540-49611-3.
3. Fahad, M., Qadir, M.A., Noshairwan, M.W., Ontological Errors - Inconsistency, Incompleteness and Redundancy. In Proceedings of 10th Intl Conference on Enterprise Information Systems, pp. 253-285, 2008, Spain, Springer,
4. Klein, M., (2001): Combining and relating ontologies: an analysis of problems and solution. In Proc. of Workshop on Ontologies and Information Sharing (IJCAI), pp. 53-62. Seattle, USA (2001)
5. Fahad, M., and Qadir, M.A., A Framework for Ontology Evaluation, 16th ICCS Supplement Proceeding, vol. 354, 2008, France, pp.149-158.
6. Fahad, M., Qadir, M.A., Noshairwan, W., Iftakhir, N., DKP-OM: A Semantic based Ontology Merger, Proceedings of 3rd International Conference on Semantic Technologies (I-Semantics 07) Graz, Austria, 2007, Pages 313-322
7. Fahad, M., Moalla, N., Bouras, A., Detection and Resolution of Semantic Inconsistency and Redundancy in an Automatic Ontology Merging System, Journal of Intelligent Information System (JIIS), Vol. 39(2) pp. 535-557, 29/4/2012, DOI 10.1007/s10844-012-0202-y
8. Fahad, M., Moalla, N., Bouras, A., Qadir, M.A., Farukh, M., Disjoint Knowledge Analysis and Preservation in Ontology Merging Process, proceedings of 5th International Conference on Software Engineering Advances (ICSEA'10), IEEE CS, August 22-27, 2010 - Nice, France.
9. Fahad, M., Moalla, N., Bouras, A., Towards ensuring Satisfiability of Merged Ontology, International conference on computational science, ICCS 2011, Procedia Computer Science 4 (2011), pp. 2216–222, 1-3 june, 2011
10. Process Matching Results: http://web.informatik.uni-mannheim.de/oaei/pm16/results.html

11. Fahad, M., Merging of axiomatic definitions of concepts in the complex OWL ontologies. Artificial Intelligence Review, (2016) doi:10.1007/s10462-016-9479-5 pp 1–35

FCA-Map Results for OAEI 2016

Mengyi Zhao[1] and Songmao Zhang[2]

[1,2]Institute of Mathematics, Academy of Mathematics and Systems Science,
Chinese Academy of Sciences, Beijing, P. R. China
[1]myzhao@amss.ac.cn, [2]smzhang@math.ac.cn

Abstract. FCA-Map is an automatic ontology matching system based on Formal Concept Analysis (FCA), which is a well developed mathematical model for analyzing individuals and structuring concepts. More precisely, we construct three types of formal contexts and extracts mappings from the lattices derived. Firstly, token-based formal context describes how class names, labels and synonyms share lexical tokens, leading to lexical mappings (anchors) across ontologies. Secondly, relation-based formal context describes how classes are in taxonomic or disjoint relationships with the anchors, leading to positive and negative structural evidence for validating the lexical matching. Lastly, after incoherence repair, positive relation-based context can be used to discover additional structural mappings. In this paper, we briefly introduce FCA-Map and its results of three tracks (i.e., Anatomy, Large Biomedical Ontologies, Disease and Phenotype) on OAEI 2016.

1 Presentation of the system

Among the first batch of OM algorithms and tools proposed in the early 2000s, FCA-Merge [4] distinguished in using Formal Concept Analysis (FCA) formalism to derive mappings from classes sharing textual documents as their individuals. Proposed by Wille [5], FCA is a well developed mathematical model for analyzing individuals and structuring concepts. FCA starts with a formal context consisting of a set of objects, a set of attributes, and their binary relations. Concept lattice, or Galois lattice, can be computed based on formal context, where each node represents a formal concept composed of a subset of objects (extent) with their common attributes (intent). The extent and the intent of a formal concept uniquely determine each other in the lattice. Further, a concept hierarchy can be derived where one formal concept becomes sub-concept of the other if its objects are contained in the latter. FCA can be naturally applied to ontology construction [3], and is also widely used in data analysis, information retrieval, and knowledge discovery.

Following the steps of FCA-Merge, several OM systems continued to use FCA as well as its alternative formalisms, exploiting different entities as the sets of objects and attributes for constructing formal contexts [1, 2, 6]. Different types of formal contexts decide the information used for ontology matching, and we observed that some intrinsic and essential knowledge of ontology has not been involved yet, including both textual information within classes (e.g., class names, labels, and synonyms) and relationships among classes (e.g., ISA, sibling, and disjointedness relations). In order to empower

FCA with as much as ontological information as possible, we proposed FCA-Map, which generates three types of formal contexts and extracts mappings from the lattices derived. The next sub-sections provide more details about FCA-Map and then discuss our results of OAEI.

1.1 State, purpose, general statement

Given two ontologies, FCA-Map builds formal contexts and uses the derived concept lattices to cluster the commonalities among ontology classes, at lexical level and structural level, respectively. Concretely, FCA-Map performs step-by-step as follows.

1. **Acquiring anchors lexically.** The token-based formal context is constructed, and from its derived concept lattice, a group of lexical anchors $\mathcal{A}$ across ontologies can be extracted.
2. **Validating anchors structurally.** Based on $\mathcal{A}$, the relation-based formal context is constructed, and from its derived concept lattice, positive and negative structural evidence of anchors can be extracted. Moreover, an enhanced alignment $\mathcal{A}'$ without incoherences among anchors is obtained.
3. **Discovering additional matches.** Based on $\mathcal{A}'$, the positive relation-based formal context is constructed, and from its derived concept lattice, additional matches across ontologies can be identified.

1.2 Specific techniques used

The process of our system consists of the following successive steps.

Step 1: Constructing the token-based formal context to acquire lexical anchors. The token-based formal context $\mathbb{K}_{lex} := (G_{lex}, M_{lex}, I_{lex})$ is described as follows. Names of ontology classes as well as their labels and synonyms, when available, are exploited after normalization that includes inflection, tokenization, stop word elimination, and punctuation elimination. In $\mathbb{K}_{lex}$, G_{lex} is the set of strings each corresponding to a name, label, or synonym of classes in two ontologies, M_{lex} is the set of tokens in these strings, and binary relation $(g, m) \in I_{lex}$ holds when string g contains token m, or a synonym or lexical variation of m. For the derived formal concepts, we restrict our attention to formal concepts whose *simplified extent* or *class-origin extent* contains exactly two strings or classes across ontologies, and extract two types of lexical anchors, namely ***Type I anchor*** for the exact match, and ***Type II anchor*** for the partial match, respectively.

Sept 2: Constructing the relation-based formal context to validate lexical anchors. Structural relationships of ontologies are exploited to validate the matches obtained at the lexical level. [7] proposed using positive and negative structural evidence among anchors for the purpose of validation. In this step, we build the relation-based formal context to obtain both positive and negative structural evidence for lexical anchors. The relation-based formal context $\mathbb{K}_{rel} := (G_{rel}, M_{rel}, I_{rel})$ is described as

follows. Classes in two source ontologies are taken as object set G_{rel}, and lexical anchors prefixed with different relational labels are taken as attribute set M_{rel}. For example, relationships *ISA*, *SIBLING-WITH*, *PART-OF*, and *DISJOINT-WITH* are labeled by "(ISA)", "(SIB)", "(PAT)", and "(I-D)" (or "(D-I)"), respectively. Binary relation $(g, m) \in I_{rel}$ holds if g has the corresponding relationship (as in the prefix of m) with the class from the same source ontology as g in the anchor of m. Formal concepts whose extents include both classes in some anchors indicate structural evidence. Such anchors are positive evidence to anchors with label"(ISA)", "(SIB)" or "(PAT)" in the intent, and vice versa. On the other hand, they are negative evidence to anchors with label "(I-D)" or "(D-I)" in the intent, and vice versa. In this way, positive and negative structural evidence set of each anchor a can be obtained, denoted by $P(a)$ and $N(a)$, respectively. Then we utilize all the positive evidence sets $\mathcal{P}$ and negative evidence sets $\mathcal{N}$ to eliminate incorrect lexical anchors and retain the correct ones.

Setp 3: Constructing the positive relation-based formal context to discover additional matches. After incoherence repair and screening, anchors retained are those supported both lexically and structurally. Based on the enhanced alignment, FCA-Map goes further to build the positive relation-based formal context aiming to identify new, structural mappings. The way positive relation-based formal context $\mathbb{K}'_{rel}$ constructed is similar to $\mathbb{K}_{rel}$, i,e., using classes in two source ontologies as object set and anchors prefixed with relationship labels as attribute set, where disjointedness relationship is no longer necessary. For the derived formal concepts, we restrict our attention to those with exactly two classes across ontologies in the *simplified extent*.

1.3 Link to the system and parameters file

SEALS wrapped version of FCA-Map for OAEI 2016 is available at https://drive.google.com/open?id=0B810qAwN1CIoM0NMV3ZJMzVsTlk.

1.4 Link to the set of provided alignments

The results obtained by FCA-Map during OAEI 2016 are available at https://drive.google.com/open?id=0B810qAwN1CIodGdPUjVWY0M3U0U.

2 Results

In this section, we present the results of FCA-Map achieved on OAEI 2016. Our system mainly focuses on Anatomy, Large Biomedical Ontologies, Disease and Phenotype.

2.1 Anatomy Track

The Anatomy track consists of finding an alignment between the Adult Mouse Anatomyand a part of the NCI Thesaurus describing the human anatomy. The results are shown in Table 1. The evaluation was run on a server with 3.46 GHz (6 cores) and 8GB RAM allocated. FCA-Map ranked fifth in Anatomy track.

Matcher	Precision	Recall	F-Measure	Runtime (s)
AML	0.95	0.936	0.943	47
CroMatcher	0.949	0.902	0.925	573
XMAP	0.929	0.865	0.896	45
LogMapBio	0.888	0.896	0.892	758
FCA-Map	0.932	0.837	0.882	117

Table 1: Results for Anatomy track

2.2 Large BioMed Track

The Large BioMed track consists of finding alignments between the Foundational Model of Anatomy (FMA), SNOMED CT, and the National Cancer Institute Thesaurus (NCI). The results obtained by FCA-Map for the small fragments of the FMA, NCI and SNOMED CT ontologies are summarize in Table 2. The evaluation of first two tasks was run on a Ubuntu Laptop with an Intel Core i7-4600U CPU @ 2.10GHz x 4 and 15Gb RAM allocated with 2 hours timeout. And the last task was run on a PC with Intel i7-4790 CPU @ 3.60GHz and 8GB RAM allocated. FCA-Map ranks second in the first two tasks.

Task	Precision	Recall	F-Measure	Runtime (s)
FMA-NCI (small)	0.954	0.917	0.935	236
FMA-SNOMED (small)	0.936	0.803	0.865	1,865
SNOMED-NCI (small)	0.914	0.666	0.771	13,542

Table 2: Results of FCA-Map for the Large BioMed Track

2.3 Disease and Phenotype Track

The Pistoia Alliance Ontologies Mapping project team organises 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). The evaluation was run on a Ubuntu Laptop with an Intel Core i7-4600U CPU @ 2.10GHz x 4 and 15Gb RAM allocated.

Matcher	Task	Precision Silver 2	Recall Silver 2	F-Measure Silver 2	Sum F-Measure Silver 2	Precision Silver 3	Recall Silver 3	F-Measure Silver 3	Sum F-Measure Silver 3
LogMap	HP-MP	0.9354	0.9125	0.9238	1.8372	0.7732	0.9729	0.8617	1.7828
	DOID-ORDO	0.9520	0.8779	0.9134		0.9052	0.9375	0.9211	
FCA-Map	HP-MP	0.9836	0.7543	0.8539	1.8162	0.9421	0.9244	0.9332	1.8706
	DOID-ORDO	0.9662	0.9586	0.9624		0.8880	0.9926	0.9374	
AML	HP-MP	0.9305	0.7998	0.8602	1.7684	0.8536	0.9446	0.8968	1.7714
	DOID-ORDO	0.8532	0.9708	0.9082		0.7784	0.9981	0.8747	
PhenoMF	HP-MP	0.7568	0.9164	0.8290	1.7149	0.6292	0.9452	0.7555	1.6905
	DOID-ORDO	0.9498	0.8301	0.8859		0.9472	0.9233	0.9351	

Table 3: Results against silver standard with vote 2 and 3

Table 3 shows the results against the silver standard which is automatically built by voting the outputs of the participating systems. LogMap is the system closer to the mappings voted by at least 2 systems, and FCA-MAP produces results very close to the silver standard with vote 3.

3 General comments

This is the first time FCA-Map system participates in the OAEI campaign. It is competitive with other systems in some tracks such as Anatomy, Large Biomedical Ontologies, Disease and Phenotype. Three types of formal contexts are constructed one-by-one, and their derived concept lattices are used to cluster the commonalities among classes at lexical and structural level, respectively. The tokens shared by two classes in these mappings are unique to their names. The lexical matching method of FCA-Map is suitable for domain ontologies having class names, labels, or synonyms from domain-specific vocabulary.

4 Conclusions

In this paper, we have presented FCA-Map and its results of three tracks (i.e.,Anatomy, Large Biomedical Ontologies, Disease and Phenotype) on OAEI 2016. The evaluation results show the good performance of FCA-Map. Future work would introduce more elements of ontology into FCA-Map including properties, individuals, and logical constructors and axioms. Optimization techniques for handling large-scale FCA contexts will also be worth exploring.

Acknowledgements. This work has been supported by the National Key Research and Development Program of China under grant 2016YFB1000902, the Natural Science Foundation of China under No. 61232015, the Knowledge Innovation Program of the Chinese Academy of Sciences (CAS), Key Lab of Management, Decision and Information Systems of CAS, and Institute of Computing Technology of CAS.

References

1. de Souza, K.X.S., Davis, J.: Aligning ontologies and evaluating concept similarities. In: OTM Confederated International Conferences" On the Move to Meaningful Internet Systems", Springer (2004) 1012–1029
2. Guan-yu, L., Shu-peng, L., et al.: Formal concept analysis based ontology merging method. In: Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on. Volume 8., IEEE (2010) 279–282
3. Obitko, M., Snsel, V., Smid, J.: Ontology design with formal concept analysis. CLA **128**(3) (2004) 1377–1390
4. Stumme, G., Maedche, A.: Fca-merge: Bottom-up merging of ontologies. In: IJCAI. Volume 1. (2001) 225–230
5. Wille, R.: Restructuring lattice theory: an approach based on hierarchies of concepts. In: Ordered sets. Springer (1982) 445–470

6. Xu, X., Wu, Y., Chen, J.: Fuzzy fca based ontology mapping. In: 2010 First International Conference on Networking and Distributed Computing, IEEE (2010) 181–185
7. Zhang, S., Bodenreider, O.: Experience in aligning anatomical ontologies. International journal on Semantic Web and information systems **3**(2) (2007) 1

Lily Results for OAEI 2016

Peng Wang[1], Wenyu Wang[1,2]

[1] School of Computer Science and Engineering, Southeast University, China
[2] Chien-Shiung Wu College, Southeast University, China
{pwang, ms} @ seu.edu.cn

Abstract. This paper presents the results of Lily in the ontology alignment contest OAEI 2016. As a comprehensive ontology matching system, this year Lily is intended to participate in three tracks of the contest: benchmark, conference, and anatomy. The specific techniques used by Lily will be introduced briefly. The strengths and weaknesses of Lily will also be discussed.

1 Presentation of the system

With the use of hybrid matching strategies, Lily, as an ontology matching system, is capable of solving some issues related to heterogeneous ontologies. It can process normal ontologies, weak informative ontologies [5], ontology mapping debugging [7], and ontology matching tunning [9], in both normal and large scales. In previous OAEI contests [1–3], Lily has achieved preferable performances in some tasks, which indicated its effectiveness and wideness of availability.

1.1 State, purpose, general statement

The core principle of matching strategies of Lily is utilizing the useful information correctly and effectively. Lily combines several effective and efficient matching techniques to facilitate alignments. There are four main matching strategies: (1) Generic Ontology Matching (GOM) is used for common matching tasks with normal size ontologies. (2) Large scale Ontology Matching (LOM) is used for the matching tasks with large size ontologies. (3) Ontology mapping debugging is used to verify and improve the alignment results. (4) Ontology matching tuning is used to enhance overall performance.

The matching process mainly contains three steps: (1) Pre-processing, when Lily parses ontologies and prepares the necessary information for subsequent steps. Meanwhile, the ontologies will be generally analyzed, whose characteristics, along with studied datasets, will be utilized to determine parameters and strategies. (2) Similarity computing, when Lily uses special methods to calculate the similarities between elements from different ontologies. (3) Post-processing, when alignments are extracted and refined by mapping debugging.

This time, Lily has few changes compared to the OAEI 2015 version.

1.2 Specific techniques used

Lily aims to provide high quality 1:1 concept pair or property pair alignments. The main specific techniques used by Lily are as follows.

Semantic subgraph An element may have heterogeneous semantic interpretations in different ontologies. Therefore, understanding the real local meanings of elements is very useful for similarity computation, which are the foundations for many applications including ontology matching. Therefore, before similarity computation, Lily first describes the meaning for each entity accurately. However, since different ontologies have different preferences to describe their elements, obtaining the semantic context of an element is an open problem. The semantic subgraph was proposed to capture the real meanings of ontology elements [4]. To extract the semantic subgraphs, a hybrid ontology graph is used to represent the semantic relations between elements. An extracting algorithm based on an electrical circuit model is then used with new conductivity calculation rules to improve the quality of the semantic subgraphs. It has been shown that the semantic subgraphs can properly capture the local meanings of elements [4].

Based on the extracted semantic subgraphs, more credible matching clues can be discovered, which help reduce the negative effects of the matching uncertainty.

Generic ontology matching method The similarity computation is based on the semantic subgraphs, which means all the information used in the similarity computation comes from the semantic subgraphs. Lily combines the text matching and structure matching techniques.

Semantic Description Document (SDD) matcher measures the literal similarity between ontologies. A semantic description document of a concept contains the information about class hierarchies, related properties and instances. A semantic description document of a property contains the information about hierarchies, domains, ranges, restrictions and related instances. For the descriptions from different entities, the similarities of corresponding parts will be calculated. Finally, all separated similarities will be combined with the experiential weights.

Matching weak informative ontologies Most existing ontology matching methods are based on the linguistic information. However, some ontologies may lack in regular linguistic information such as natural words and comments. Consequently the linguistic-based methods will not work. Structure-based methods are more practical for such situations. Similarity propagation is a feasible idea to realize the structure-based matching. But traditional propagation strategies do not take into consideration the ontology features and will be faced with effectiveness and performance problems. Having analyzed the classical similarity propagation algorithm, *Similarity Flood*, we proposed a new structure-based ontology matching method [5]. This method has two features: (1) It has more strict but reasonable propagation conditions which lead to more efficient matching processes and better alignments. (2) A series of propagation strategies are used to

improve the matching quality. We have demonstrated that this method performs well on the OAEI benchmark dataset [5].

However, the similarity propagation is not always perfect. When more alignments are discovered, more incorrect alignments would also be introduced by the similarity propagation. So Lily also uses a strategy to determine when to use the similarity propagation.

Large scale ontology matching Matching large ontologies is a challenge due to its significant time complexity. We proposed a new matching method for large ontologies based on reduction anchors [6]. This method has a distinct advantage over the divide-and-conquer methods because it does not need to partition large ontologies. In particular, two kinds of reduction anchors, positive and negative reduction anchors, are proposed to reduce the time complexity in matching. Positive reduction anchors use the concept hierarchy to predict the ignorable similarity calculations. Negative reduction anchors use the locality of matching to predict the ignorable similarity calculations. Our experimental results on the real world datasets show that the proposed methods are efficient in matching large ontologies [6].

Ontology mapping debugging Lily utilizes a technique named *ontology mapping debugging* to improve the alignment results [7]. Different from existing methods that focus on finding efficient and effective solutions for the ontology mapping problems, mapping debugging emphasizes on analyzing the mapping results to detect or diagnose the mapping defects. During debugging, some types of mapping errors, such as redundant and inconsistent mappings, can be detected. Some warnings, including imprecise mappings or abnormal mappings, are also locked by analyzing the features of mapping result. More importantly, some errors and warnings can be repaired automatically or can be presented to users with revising suggestions.

Ontology matching tuning Lily adopted ontology matching tuning this year. By performing parameter optimization on training datasets [9], Lily is able to determine the best parameters for similar tasks. Those data will be stored. When it comes to real matching tasks, Lily will perform statistical calculations on the new ontologies to acquire their features that help it find the most suitable configurations, based on previous training data. In this way, the overall performance can be improved.

Currently, ontology matching tuning is not totally automatic. It is difficult to find out typical statistical parameters that distinguish each task from others. Meanwhile, learning from test datasets can be really time-consuming. Our experiment is just a beginning.

1.3 Adaptations made for the evaluation

For benchmark, anatomy and conference tasks, Lily is totally automatic, which means Lily can be invoked directly from the SEALS client. It will also determine which strategy to use and the corresponding parameters.

1.4 Link to the system and parameters file

SEALS wrapped version of Lily for OAEI 2016 is available at `https://drive.google.com/folderview?id=0B5j4YFThSEQkRXdUVUg5eHRFSUE&usp=sharing`.

1.5 Link to the set of provided alignments

The set of provided alignments, as well as overall performance, is available at each track of the OAEI 2016 official website, `http://oaei.ontologymatching.org/2016/`.

2 Results

2.1 Benchmark track

There are two datasets in different sizes: *biblio* and *film*. The biblio dataset concerns bibliographic references and is inspired freely from BibTeX. The film dataset contains a movie ontology in English and French. Especially, the film dataset was not known from the participants when submitting their systems, and actually have been generated afterwards. This biblio dataset will be matched using Generic Ontology Matching, because the ontology size is generally small. Lily will automatically choose matching methods and strategy to handle with film dataset.

There are five groups of test suites in each dataset. Each test suite has 94 matching tasks. The overall results of one test suite will be represented by the mean value of Precision, Recall and F-Measure. Test suites were generated from the same seed ontologies, which means they are all equal. Thus, the harmonic mean values of all test suites will be used to evaluate how well Lily worked.

The detailed results are shown in Table 1.

Table 1. The performance in the Benchmark track

Test suite	Precision	Recall	F-Measure
biblio-r1	0.97	0.84	0.90
biblio-r2	0.96	0.83	0.89
biblio-r3	0.97	0.84	0.90
biblio-r4	0.97	0.83	0.89
biblio-r5	0.97	0.83	0.89
H-mean	**0.97**	**0.83**	**0.89**
film-r1	0.97	0.69	0.80
film-r2	0.97	0.69	0.80
film-r3	0.97	0.70	0.81
film-r4	0.97	0.70	0.81
film-r5	0.97	0.70	0.81
H-mean	**0.97**	**0.70**	**0.81**

As Table 1 has shown, Lily handles Benchmark datasets well. According to the Benchmark results of OAEI2016[1], Lily has the highest overall F-Measure among all matching systems.

2.2 Anatomy track

The anatomy matching task consists of two real large-scale biological ontologies. Table 2 shows the performance of Lily in the Anatomy track on a server with one 3.46 GHz, 6-core CPU and 8GB RAM allocated. The time unit is second (s).

Table 2. The performance in the Anatomy track

Matcher	Runtime	Precision	Recall	F-Measure
Lily	272s	0.87	0.79	0.83

Compared with the result in OAEI 2011 [8], there is a small improvement of Precision, Recall and F-Measure, from 0.80, 0.72 and 0.76 to 0.87, 0.79 and 0.83, respectively. One main reason for the improvement is that we found the names of classes not semantically useful, which would confuse Lily when the similarity matrix was calculated. After the names were excluded, better alignments were generated. Besides, there is a significant reduction of the time consumption, from 563s to 272s. This is not only the result of stronger CPU, but also because more optimizations, like parallelization, were applied to the algorithms in Lily.

However, as can be seen in the overall result, Lily lies in the middle position of the rank, which indicates it is still possible to make further progress. Addi-

[1] `http://oaei.ontologymatching.org/2016/results/benchmarks/index.html`

tionally, some key algorithms have not been successfully parallelized. After that is done, the time consumption is expected to be further reduced.

2.3 Conference track

In this track, there are 7 independent ontologies that can be matched with one another. The 21 subtasks are based on given reference alignments. As a result of heterogeneous characters, it is a challenge to generate high-quality alignments for all ontology pairs in this track.

Lily adopted ontology matching tuning for the Conference track this year. Table 3 shows its latest performance.

Table 3. The performance in the Conference track

Test Case ID	Precision	Recall	F-Measure
cmt-conference	0.53	0.6	0.56
cmt-confof	0.80	0.25	0.38
cmt-edas	0.64	0.54	0.58
cmt-ekaw	0.55	0.55	0.55
cmt-iasted	0.57	1.00	0.73
cmt-sigkdd	0.70	0.58	0.64
conference-confof	0.67	0.53	0.59
conference-edas	0.41	0.41	0.41
conference-ekaw	0.62	0.64	0.63
conference-iasted	0.67	0.43	0.52
conference-sigkdd	0.71	0.67	0.69
confof-edas	0.69	0.47	0.56
confof-ekaw	0.79	0.75	0.77
confof-iasted	0.46	0.67	0.55
confof-sigkdd	0.17	0.14	0.15
edas-ekaw	0.67	0.52	0.59
edas-iasted	0.50	0.37	0.42
edas-sigkdd	0.63	0.33	0.43
ekaw-iasted	0.50	0.80	0.62
ekaw-sigkdd	0.50	0.46	0.48
iasted-sigkdd	0.56	0.67	0.61
Average	**0.59**	**0.53**	**0.56**

Compared with the result in OAEI 2011 [8], there is a significant improvement of mean Precision, Recall and F-Measure, from 0.36, 0.47 and 0.41 to 0.59, 0.53 and 0.56, respectively. Besides, all the tasks share the same configurations, so it is possible to generate better alignments by assigning the most suitable parameters for each task. We will continue to enhance this feature.

3 General comments

On the whole, Lily is a comprehensive ontology matching system with the ability to handle multiple types of ontology matching tasks, of which the results are generally competitive. The performance of Lily is similar to the results of 2015 [10]. However, Lily still lacks in strategies for some newly developed matching tasks. The relatively high time and memory consumption also prevent Lily from finishing some challenging tasks.

4 Conclusion

In this paper, we briefly introduced our ontology matching system Lily. The matching process and the special techniques used by Lily were presented, and the alignment results were carefully analyzed.

There is still so much to do to make further progress. Lily needs more optimization to handle large ontologies with limited time and memory. Thus, techniques like parallelization will be applied more. Also, we have just tried out ontology matching tuning. With further research on that, Lily will not only produce better alignments for tracks it was intended for, but also be able to participate in the interactive track.

References

[1] Peng Wang, Baowen Xu: Lily: ontology alignment results for OAEI 2009. In The 4th International Workshop on Ontology Matching, Washington Dc., USA (2009)

[2] Peng Wang, Baowen Xu: Lily: Ontology Alignment Results for OAEI 2008. In The Third International Workshop on Ontology Matching, Karlsruhe, Germany (2008)

[3] Peng Wang, Baowen Xu: LILY: the results for the ontology alignment contest OAEI 2007. In The Second International Workshop on Ontology Matching (OM2007), Busan, Korea (2007)

[4] Peng Wang, Baowen Xu, Yuming Zhou: Extracting Semantic Subgraphs to Capture the Real Meanings of Ontology Elements. Journal of Tsinghua Science and Technology, vol. 15(6), pp. 724-733 (2010)

[5] Peng Wang, Baowen Xu: An Effective Similarity Propagation Model for Matching Ontologies without Sufficient or Regular Linguistic Information, In The 4th Asian Semantic Web Conference (ASWC2009), Shanghai, China (2009)

[6] Peng Wang, Yuming Zhou, Baowen Xu: Matching Large Ontologies Based on Reduction Anchors. In The Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI 2011), Barcelona, Catalonia, Spain (2011)

[7] Peng Wang, Baowen Xu: Debugging Ontology Mapping: A Static Approach. Computing and Informatics, vol. 27(1), pp. 2136 (2008)

[8] Peng Wang: Lily results on SEALS platform for OAEI 2011. Proc. of 6th International Workshop on Ontology Matching, pp. 156-162 (2011)

[9] Yang, Pan, Peng Wang, Li Ji, Xingyu Chen, Kai Huang, Bin Yu: Ontology Matching Tuning Based on Particle Swarm Optimization: Preliminary Results. In The Semantic Web and Web Science, pp. 146-155 (2014)

[10] Wenyu Wang, Peng Wang: Lily results for OAEI 2015. Proc. of 11th International Workshop on Ontology Matching, (2015)

LogMap family participation in the OAEI 2016

E. Jiménez-Ruiz[1], B. Cuenca Grau[2], and V. Cross[3]

[1] Department of Informatics, University of Oslo, Oslo, Norway
[2] Department of Computer Science, University of Oxford, Oxford, UK
[3] Computer Science and Software Engineering, Miami University, Oxford, OH, United States

Abstract. We present the participation of LogMap and its variants in the OAEI 2016 campaign. The LogMap project started in January 2011 with the objective of developing a scalable and logic-based ontology matching system. This is our seventh participation in the OAEI and the experience has so far been very positive. LogMap is one of the few systems that participates in all OAEI tracks.

1 Presentation of the system

Ontology matching systems typically rely on lexical and structural heuristics and the integration of the input ontologies and the mappings may lead to many undesired logical consequences. In [12] three principles were proposed to minimize the number of potentially unintended consequences, namely: *(i) consistency principle*, the mappings should not lead to unsatisfiable classes in the integrated ontology; *(ii) locality principle*, the mappings should link entities that have similar *neighbourhoods*; *(iii) conservativity principle*, the mappings should not introduce alterations in the classification of the input ontologies. Violations to these principles may hinder the usefulness of ontology mappings. The practical effect of these violations, however, is clearly evident when ontology alignments are involved in complex tasks such as query answering [20].

LogMap [11, 13] is a highly scalable ontology matching system that implements the consistency and locality principles. LogMap also supports (real-time) user interaction during the matching process, which is essential for use cases requiring very accurate mappings. LogMap is one of the few ontology matching system that *(i)* can efficiently match semantically rich ontologies containing tens (and even hundreds) of thousands of classes, *(ii)* incorporates sophisticated reasoning and repair techniques to minimise the number of logical inconsistencies, and *(iii)* provides support for user intervention during the matching process.

LogMap relies on the following elements, which are keys to its favourable scalability behaviour (see [11, 13] for details).

Lexical indexation. An inverted index is used to store the lexical information contained in the input ontologies. This index is the key to efficiently computing an initial set of mappings of manageable size. Similar indexes have been successfully used in information retrieval and search engine technologies [2].

Logic-based module extraction. The practical feasibility of unsatisfiability detection and repair critically depends on the size of the input ontologies. To reduce the size of the problem, we exploit ontology modularisation techniques. Ontology modules with

well-understood semantic properties can be efficiently computed and are typically much smaller than the input ontology (e.g. [5]).

Propositional Horn reasoning. The relevant modules in the input ontologies together with (a subset of) the candidate mappings are encoded in LogMap using a Horn propositional representation. Furthermore, LogMap implements the classic Dowling-Gallier algorithm for propositional Horn satisfiability [6]. Such encoding, although incomplete, allows LogMap to detect unsatisfiable classes soundly and efficiently.

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

Local repair. LogMap performs a greedy local repair; that is, it repairs unsatisfiabilities on-the-fly and only looks for the first available repair plan.

Semantic indexation. The Horn propositional representation of the ontology modules and the mappings is efficiently indexed using an interval labelling schema [1] — an optimised data structure for storing directed acyclic graphs (DAGs) that significantly reduces the cost of answering taxonomic queries [4, 21]. In particular, this semantic index allows us to answer many entailment queries as an index lookup operation over the input ontologies and the mappings computed thus far, and hence without the need for reasoning. The semantic index complements the use of the propositional encoding to detect and repair unsatisfiable classes.

1.1 LogMap variants in the 2016 campaign

In the 2016 campaign we have participated with two additional variants:

LogMapLt is a "lightweight" variant of LogMap, which essentially only applies (efficient) string matching techniques.

LogMapBio includes an extension to use BioPortal [8, 9] as a (dynamic) provider of mediating ontologies instead of relying on a few preselected ontologies [3].

This year we did not participate with LogMapC[4] since in OAEI 2016 there are not alignment tasks suitable for a correct evaluation of LogMapC.[5] The repair algorithm in LogMapC is more aggressive than in LogMap, which harms its results if the alignment task does not take into account the conservativity principle.

1.2 Adaptations made for the 2016 evaluation

LogMap's algorithm described in [11, 13, 14] has been adapted with the following new functionalities:

i **Extended multilingual support.** We have extended our multilingual module with additional translations.

[4] LogMapC is a variant of LogMap which, in addition to the consistency and locality principles, also implements the conservativity principle (see details in [22–24]).

[5] The interested reader please refer to [24, 17] for examples of alignment tasks suitable for LogMapC.

ii **Extended instance matching support.** We have partially adapted LogMap's instance matching module to cope with the new OAEI 2016 tasks.

iii **BioPortal module.** We have adapted LogMapBio with respect to the changes in the BioPortal API. Note that LogMapBio only participates in the biomedical tracks. In the other tracks the results are expected to be the same as LogMap.

1.3 Link to the system and parameters file

LogMap is open-source and released under GNU Lesser General Public License 3.0.[6] LogMap components and source code are available from the LogMap's GitHub page: `https://github.com/ernestojimenezruiz/logmap-matcher/`.

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

LogMap, including support for interactive ontology matching, can also be used directly through an AJAX-based Web interface: `http://krrwebtools.cs.ox.ac.uk/`. This interface has been very well received by the community since it was deployed in 2012. More than 2,500 requests coming from a broad range of users have been processed so far.

1.4 Modular support for mapping repair

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

We believe that these systems could significantly improve their output if they were to implement repair techniques similar to those available in LogMap. Therefore, with the goal of providing a useful service to the community, we have made LogMap's ontology repair module (LogMap-Repair) available as a self-contained software component that can be seamlessly integrated in most existing ontology matching systems [16, 7].

2 General comments and conclusions

Please refer to `http://oaei.ontologymatching.org/2016/results/` for the results of the LogMap family in the OAEI 2016 campaign.

2.1 Comments on the results

LogMap has been one of the top systems in the OAEI 2016 and one of the few system that participates in all tracks. Furthermore, it has also been one of the few systems implementing repair techniques and providing (almost) coherent mappings in all tracks.

LogMap's main weakness is that the computation of candidate mappings is based on the similarities between the vocabularies of the input ontologies; hence, in the cases where the ontologies are lexically disparate or do not provide enough lexical information LogMap is at a disadvantage.

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

2.2 Discussions on the way to improve the proposed system

LogMap is now a stable and mature system that has been made available to the community and has been extensively tested. There are, however, many exciting possibilities for future work. For example we aim at improving the current multilingual features and the current use of external resources like BioPortal. Furthremore, we are applying LogMap in practice in the domain of oil and gas industry within the FP7 Optique[7] [19, 15, 10, 18]. This practical application presents a very challenging problem.

Acknowledgements

This work was supported by the Centre for Scalable Data Access (SIRIUS), the EPSRC projects ED3, Score! and DBOnto, and by the EU FP7 project Optique (grant agreement 318338).

We would also like to thank Ian Horrocks, Alessandro Solimando, Anton Morant, Yujiao Zhou, Weiguo Xia, Xi Chen, Yuan Gong and Shuo Zhang, who have contributed to the LogMap project in the past.

References

1. Agrawal, R., Borgida, A., Jagadish, H.V.: Efficient management of transitive relationships in large data and knowledge bases. In: ACM SIGMOD Conf. on Management of Data. pp. 253–262 (1989)
2. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval. ACM Press / Addison-Wesley (1999)
3. Chen, X., Xia, W., Jiménez-Ruiz, E., Cross, V.: Extending an ontology alignment system with bioportal: a preliminary analysis. In: Poster at Int'l Sem. Web Conf. (ISWC) (2014)
4. Christophides, V., Plexousakis, D., Scholl, M., Tourtounis, S.: On labeling schemes for the Semantic Web. In: Int'l World Wide Web (WWW) Conf. pp. 544–555 (2003)
5. Cuenca Grau, B., Horrocks, I., Kazakov, Y., Sattler, U.: Modular reuse of ontologies: Theory and practice. J. Artif. Intell. Res. 31, 273–318 (2008)
6. Dowling, W.F., Gallier, J.H.: Linear-time algorithms for testing the satisfiability of propositional Horn formulae. J. Log. Prog. 1(3), 267–284 (1984)
7. Faria, D., Jiménez-Ruiz, E., Pesquita, C., Santos, E., Couto, F.M.: Towards annotating potential incoherences in bioportal mappings. In: 13th Int'l Sem. Web Conf. (ISWC) (2014)
8. Fridman Noy, N., Shah, N.H., Whetzel, P.L., Dai, B., et al.: BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic Acids Research 37, 170–173 (2009)
9. Ghazvinian, A., Noy, N.F., Jonquet, C., Shah, N.H., Musen, M.A.: What four million mappings can tell you about two hundred ontologies. In: Int'l Sem. Web Conf. (ISWC) (2009)
10. Giese, M., Soylu, A., Vega-Gorgojo, G., Waaler, A., Haase, P., Jimenez-Ruiz, E., Lanti, D., Rezk, M., Xiao, G., Ozcep, O., Rosati, R.: Optique — Zooming In on Big Data Access. Computer 48(3), 60–67 (2015)
11. Jiménez-Ruiz, E., Cuenca Grau, B.: LogMap: Logic-based and Scalable Ontology Matching. In: Int'l Sem. Web Conf. (ISWC). pp. 273–288 (2011)
12. Jiménez-Ruiz, E., Cuenca Grau, B., Horrocks, I., Berlanga, R.: Logic-based assessment of the compatibility of UMLS ontology sources. J. Biomed. Sem. 2 (2011)

[7] http://www.optique-project.eu/

13. Jiménez-Ruiz, E., Cuenca Grau, B., Zhou, Y., Horrocks, I.: Large-scale interactive ontology matching: Algorithms and implementation. In: Europ. Conf. on Artif. Intell. (ECAI) (2012)
14. Jiménez-Ruiz, E., Grau, B.C., Solimando, A., Cross, V.V.: Logmap family results for OAEI 2015. In: Proceedings of the 10th International Workshop on Ontology Matching collocated with the 14th International Semantic Web Conference (ISWC 2015), Bethlehem, PA, USA, October 12, 2015. pp. 171–175 (2015), http://ceur-ws.org/Vol-1545/oaei15_paper10.pdf
15. Jiménez-Ruiz, E., Kharlamov, E., Zheleznyakov, D., Horrocks, I., Pinkel, C., Skjæveland, M.G., Thorstensen, E., Mora, J.: BootOX: Practical Mapping of RDBs to OWL 2. In: International Semantic Web Conference (ISWC) (2015), http://www.cs.ox.ac.uk/isg/tools/BootOX/
16. Jiménez-Ruiz, E., Meilicke, C., Cuenca Grau, B., Horrocks, I.: Evaluating mapping repair systems with large biomedical ontologies. In: 26th Description Logics Workshop (2013)
17. Jimenez-Ruiz, E., Payne, T.R., Solimando, A., Tamma, V.: Limiting logical violations in ontology alignment through negotiation. In: Proceedings of the 15th International Conference on Principles of Knowledge Representation and Reasoning (KR). AAAI Press (April 2016)
18. Kharlamov, E., Hovland, D., Jiménez-Ruiz, E., Lanti, D., Lie, H., Pinkel, C., Rezk, M., Skjæveland, M.G., Thorstensen, E., Xiao, G., Zheleznyakov, D., Horrocks, I.: Ontology Based Access to Exploration Data at Statoil. In: International Semantic Web Conference (ISWC). pp. 93–112 (2015)
19. Kharlamov, E., Jiménez-Ruiz, E., Zheleznyakov, D., et al.: Optique: Towards OBDA Systems for Industry. In: Eur. Sem. Web Conf. (ESWC) Satellite Events. pp. 125–140 (2013)
20. Meilicke, C.: Alignment Incoherence in Ontology Matching. Ph.D. thesis, University of Mannheim (2011)
21. Nebot, V., Berlanga, R.: Efficient retrieval of ontology fragments using an interval labeling scheme. Inf. Sci. 179(24), 4151–4173 (2009)
22. Solimando, A., Jiménez-Ruiz, E., Guerrini, G.: Detecting and correcting conservativity principle violations in ontology-to-ontology mappings. In: Int'l Sem. Web Conf. (ISWC) (2014)
23. Solimando, A., Jiménez-Ruiz, E., Guerrini, G.: A multi-strategy approach for detecting and correcting conservativity principle violations in ontology alignments. In: Proc. of the 11th International Workshop on OWL: Experiences and Directions (OWLED). pp. 13–24 (2014)
24. Solimando, A., Jimenez-Ruiz, E., Guerrini, G.: Minimizing conservativity violations in ontology alignments: Algorithms and evaluation. Knowledge and Information Systems (2016), https://github.com/asolimando/logmap-conservativity/

LPHOM results for OAEI 2016

Imen Megdiche, Olivier Teste, and Cassia Trojahn

Institut de Recherche en Informatique de Toulouse (UMR 5505),
Toulouse, France
{Imen.Megdiche, Olivier.Teste, Cassia.Trojahn}@irit.fr

Abstract. This paper presents the results obtained by LPHOM (Linear Program for Holistic Ontology Matching) system in the OAEI 2016 campaign. This is the first participation of our system in the OAEI campaigns. It has participated in four tracks (Benchmark, Anatomy, Conference, and Multifarm). We report here a general discussion on the results and on the future improvements.

1 Presentation of the system

LPHOM (Linear Program for Holistic Ontology Matching) is a holistic ontology matching system [2], participating for the first time in the OAEI campaign. Altough the system has been designed to deal with holistic ontology matching [3] (i.e., matching multiple ontologies simultaneously), it is able as well to deal with pairwise ontology mathing, as described here. The reader can refer to [2] for a detailed description of the system.

LPHOM treats the ontology matching problem, at schema-level, as a combinatorial optimization problem. The problem is modeled through a linear program extending the maximum-weighted graph matching problem with linear constraints (matching cardinality, structural, and coherence constraints).

LPHOM follows the execution workflow as depicted in Figure 1. This workflow is composed of four main steps :

1. The first step consists in ontology loading, flattening and translating. After loading the N different ontologies (two ontologies in the case of OAEI) we flatten every ontology entity (classes, object properties and data properties) in a same structure, named *Node*. As shown in Figure 1, classes, object properties and data properties inherit from *Node*. The idea behind flattening the ontologies is to simplify the access to all information about each entity, which can be seen near to the structure of document-oriented NoSql databases. But actually, as duplication and treatment are done in memory, pre-processing is not very performant. This step also includes the translation of the labels of entities in case of the non-English ontologies. For that, we have used the Microsoft-translation Java API[1].
2. The second step consists of similarity matrices construction. For a set of N ontologies, we compute $N(N-1)/2$ similarity matrices representing the average results of different element-level matchers. These matrices are computed between each pair of ontologies and for each type of entity (classes, object properties and data

[1] https://www.microsoft.com/en-us/translator/translatorapi.aspx

properties). For OAEI, similarity matrices have been constructed with *character-based metrics* [4] (ISUB and 3-gram to compute similarity between tokens then generalized Mongue-Elkan method on these metrics to get the similarity between entities) and *token-based category* (Jaccard). Our system also uses the Lin's semantic measure [1], but due to some packaging problems, this metric was unfortunately not been used in the current OAEI 2016 version.

3. The third step consists of constructing the linear program, which is detailed in [2]. The algorithm was developed in Java by the mean of the methods proposed by the Java API of the CPLEX Solver[2]. For constructing the linear program, we consider only the pairs of correspondences (our decision variables), which similarity measure is higher than 0.65 (this threshold is equals to 0 for the Multifarm track). We highlight also that the used threshold is the same for each type of entity (classes, object properties and data properties).
4. The fourth step consists of resolving the linear program using the CPLEX solver. The solution represents the set of final correspondences, which will be flushed to the RDF file (output alignments).

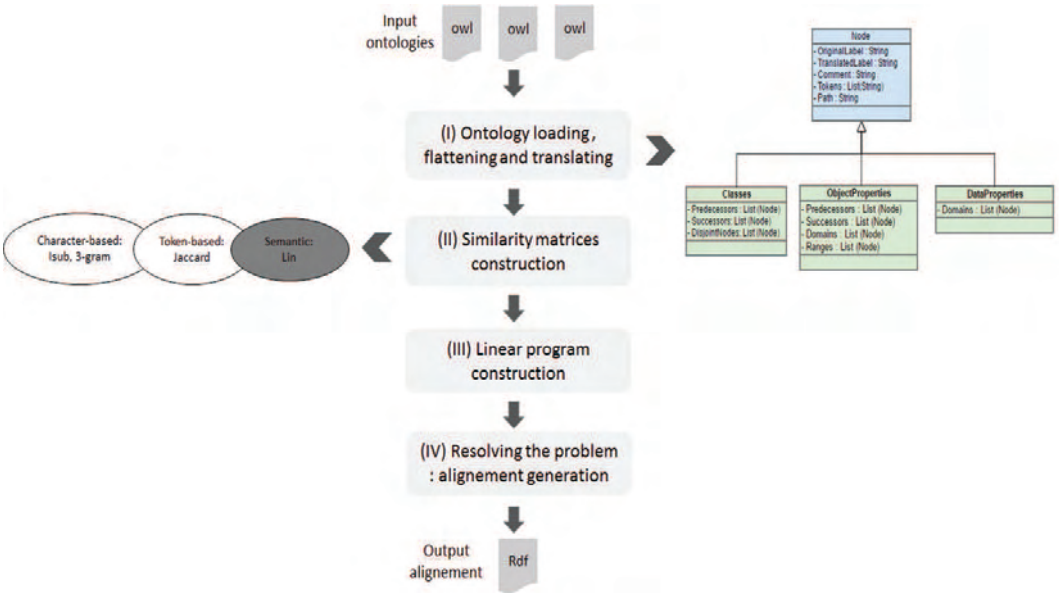


Fig. 1. LPHOM execution workflow.

[2] http://www.ibm.com/support/knowledgecenter/SSSA5P_12.6.2/ilog.odms.cplex.help/refjavacplex/html/index.html

2 Link to the system and configuration file

LPHOM is actually not an open-source system. This system is in its beta version and several improvements and refactoring have to be implemented to LPHOM before opening its source code. However, it can be dowloaded at `https://drive.google.com/drive/folders/0B5j4YFThSEQkTWxKRzRMWFlVQ2M`, together with the instructions on how to install all the dependencies (in particular CPLEX solver).

3 Results

The reader can refer to the OAEI web pages[3] for the results of LPHOM in the tasks Anatomy, Benchmark, Conference and Multifarm. In the following, we provide a complementary discussion on these results.

It is important to note that some results on the Conference and Anatomy tracks have been reported in [2], using the data sets provided in OAEI 2015. However, the results reported for OAEI 2015 care slightly different from the results of OAEI 2016 reported here. It is due to the fact that in OAEI we have not used any semantic measure.

3.1 Anatomy

Our results for the anatomy track are summarized in Table 1.

Table 1. LPHOM results for anatomy track.

Rank(F1)	Size	P	F1	R	R+	Coherent	Runtime
10/13	1555	0.79	0.718	0.727	0.497	-	1601 sec (26min)

First, we can observe that our tool is quite slow to perform the Anatomy track, and takes about 26 min (the faster system took 20 seconds). The non-scalability of our tool is closely dependant on the non-optimised pre-processing steps (in particular, first and second ones) in the execution workflow (Figure 1). In fact, flattening the structure of ontologies entails performance problems which also depend on the type of the executed similarity measure. To illustrate this problem, when using only Jaccard metric, LPHOM spent about 36 sec to run the Anatomy task (as reported in [2]).

Furthermore, we report that the chosen threshold (0.65) reveals to be very low for this track. That is why we get a higher number of generated alignments, in particular false positive ones.

Finally, we observed that some incoherent results have been obtained for this track. In fact, the constraints we have proposed in the LPHOM approach [2] are mainly limited to non-disjoint entities. We should may add some new constraints in our model in order to tackle the incoherences generated in this track.

[3] `http://oaei.ontologymatching.org/2016/`

3.2 Benchmark

The organizers of this track faced some problems to execute our package due to the external call of CPLEX. Hence, in the OAEI web pages[4] our results were not reported. Locally, we get quite interesting results (Table 2) for the biblio data set of this track.

Table 2. LPHOM results for benchmark track (biblio data set).

P	F	R
0.77	0.60	0.50

For the film data set, our system has launched some exceptions when pre-processing the ontologies and no alignments have been generated.

3.3 Conference

The whole results of LPHOM for the tasks RA1, RA2, RAR2 are reported in the Conference web page results[5].

We discuss in this section the differences between the results of LPHOM for OAEI 2015 (reported in [2]) and the results for OAEI 2016. Table 3 presents the results for both data sets, for the RA1 task.

Table 3. Comparison between the results of conference track in OAEI 2015 and OAEI 2016.

	Rank	P	F.5	F1	F2	R	threshold
RA1-M1 (2015)	7/15	0.76	0.73	0.69	0.66	0.64	0.65
RA1-M1(2016)	12/14	0.89	0.71	0.55	0.45	0.4	0.76
RA1-M2 (2015)	8/13	0.23	0.24	0.25	0.26	0.26	0.65
RA1-M2 (2016)	8/14	0.08	0.05	0.03	0.02	0.02	0
RA1-M3 (2015)	8/15	0.65	0.63	0.61	0.59	0.58	0.65
RA1-M3 (2016)	12/14	0.76	0.61	0.47	0.38	0.34	0.86

We can observe a slight difference between OAEI 2016 and OAEI 2015 results. This is mainly due to the fact that we did not use any semantic measure in the OAEI 2016 version (as reported above, due to some packaging problems).

Furthermore, compared to the results of OAEI 2015, the results of OAEI 2016 are filtered according to a different threshold computed by the organizers (and applied to the final alignments), which gives the better results on F-Measure.

Finally, we stress a very interesting aspect on our results, which concerns conservativity and consistency violation. In OAEI 2016, our approach have no conservativity principle violation nor consistency violation. This was also observed in OAEI 2015

[4] http://oaei.ontologymatching.org/2016/

[5] http://oaei.ontologymatching.org/2016/conference/eval.html

evaluations. In fact, we have removed $\sim$ 1 alignment which does not respect consistency violation. These results check the efficiency of the proposed linear constraints.

3.4 Multifarm

Our results for the Multifarm track are summarized in Table 4.

Table 4. LPHOM results for Multifarm track.

	Rank	Time	pairs	Size	P	F1M	R
Different ontologies	8/12	2497	34	84.22	0.01(0.02)	0.02(0.04)	0.08(.08)
Same ontologies	5/12	2497	34	127.91	0.13(0.22)	0.13(0.21)	0.13(0.13)

For this track, we have used a threshold equals to 0 (when filtering out the correspondences from the similarity matrices), which explains the important number of generated alignments (in average, 84.22 for the tests cases involving matching different ontologies in different languages, and 127.91 for the test cases involving matching same ontologies in different languages).

Although using a basic cross-lingual strategy based on translation, we obtained better results when matching the same ontologies, once our system takes advantage of the structure of the ontologies. However, matching different ontologies in different languages requires an improvement in the translation step and similarity metrics.

Finally, we have encountered problems when translating Chinese language, due to problems when accessing the translation server and its Chinese enconding, what will be corrected in the future version. In fact, the translation worked well on our local machine but did not correctly worked when accesing remotely via the SEALS platform.

4 General comments

In the current version of LPHOM, we have been almost focused on modeling and expressing the matching problem through a set of constraints (cardinality, structural, and coherence constraints) applied on similarity matrices. The similarity matrices have been calculated from a set of (few) lexical similarities with a same filtering threshold for most tracks (0.65 for Anatomy, Benchmark and Conference and 0 for Multifarm). However, the choice of similarity metrics or the choice of threshold are also important to success the OAEI tracks. In this regard, we plan to improve the criteria of selection of similarity measures and thresholds for our future participation.

As stated above, LPHOM is a system designed to deal with holistic ontology matching at schema-level. Hence, LPHOM was not able to generated alignments for the tasks involving instance matching (Instance Matching and Process Model tracks). We plan to implement instance matching strategies in future versions of the system.

Finally, our system was not able to deal at all with the large ontologies in the LargeBio and Phenotype tasks. In fact, it consumes a large amount of memory space on the pre-processing steps (first and second steps according to Figure 1), we plan to address these points by in the future.

5 Conclusion

This paper briefly introduced the LPHOM system and discussed the main points on the results of its first participation in OAEI campaigns. We have as well pointed out some directions for future improvements.

References

1. Dekang Lin. An information-theoretic definition of similarity. In *In Proceedings of the 15th International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann, 1998.
2. Imen Megdiche, Olivier Teste, and Cassia Trojahn. An extensible and linear approach for holistic ontology matching. In *Proceedings of the 15th International Semantic Web Conference, Kobe, Japan, October 17-21, 2016*, pages 393–410, 2016.
3. Erhard Rahm. *Schema Matching and Mapping*, chapter Towards Large-Scale Schema and Ontology Matching, pages 3–27. Springer Berlin Heidelberg, 2011.
4. Yufei Sun, Liangli Ma, and Wang Shuang. A comparative evaluation of string similarity metrics for ontology alignement. *Journal of Information & Computational Science*, 12(3):957 – 964, 2015.

LYAM++ Results for OAEI 2016

Abdel Nasser Tigrine, Zohra Bellahsene, Konstantin Todorov

{lastname}@lirmm.fr
LIRMM / University of Montpellier, France

Abstract. LYAM++ is a fully automatic ontology matching system based on the use of external sources. Our approach applies a novel orchestration of the components of the matching workflow. We present our results on anatomy, conference large biomedical and Multifarm tracks of OAEI2016.

1 Presentation of the System

In spite of the considerable advance that has been made in the field of ontology matching recently, many questions remain open [1]. The current work addresses the challenge of using background knowledge with a focus on aligning cross-lingual ontologies, i.e., ontologies defined in different natural languages [2].

Indeed, considering multilingual and cross-lingual information is becoming more and more important, in view particularly of the growing number of web content-creating non-English users and the clear demand of cross-language interoperability. In the context of the web of data, it is important to propose procedures for linking vocabularies across natural languages, in order to foster the creation of a veritable global information network.

The use of different natural languages in the concepts and relations labeling process is becoming an important source of ontology heterogeneity. The methods that have been proposed to deal with it most commonly rely on automatic translation of labels to a single target language [3,4] or apply machine learning techniques [2]. However, machine translation tolerates low precision levels and machine learning methods require large training corpus that is rarely available in an ontology matching scenario. An inherent problem of translation is that there is often a lack of exact one-to-one correspondence between the terms in different natural languages.

1.1 State, Purpose, General Statement

We present LYAM++ (Yet Another Matcher - Light)[5], a fully automatic ontology matching system based on the use of external sources. LYAM++ does not rely on machine translation for cross-lingual ontology matching. Instead, we make use of the openly available general-purpose multilingual semantic network BabelNet[1] in order to recreate the missing semantic context in the matching

[1] http://babelnet.org/

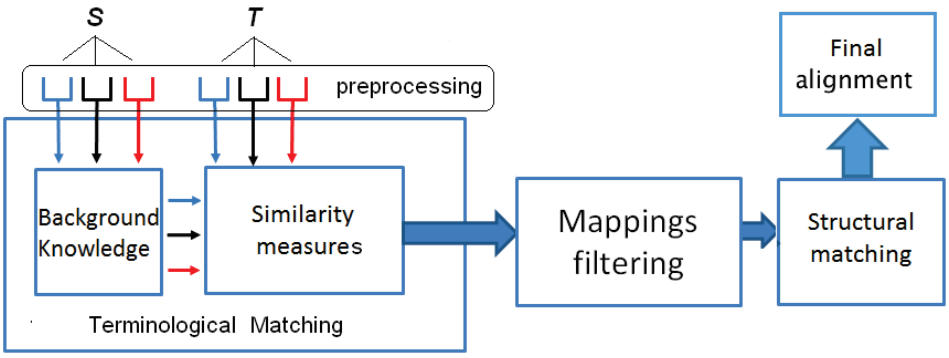


Fig. 1: The processing pipeline of LYAM++.

process. Another original feature of our approach is the choice of orchestration of the matching workflow. The novel workflow orchestration provides better results compared to the classical one. We refer the reader to the results reported in [5].

1.2 Specific Techniques Used

The workflow of LYAM++ is given in Fig 1. The overall process consists of four main components: a terminological matcher, a mapping selection module and, finally, a structural matcher. One of the original contributions of this work is the choice of orchestration of these components. Indeed, the places of the mapping selection module and the structural matcher are reversed in the existing OM tools [6]. However, we wanted to ensure that we feed only good quality mappings to the structural matcher, therefore we decided to filter the discovered correspondences right after producing the initial alignment. This decision is supported experimentally in[5].

The *terminological matching* module, the second contribution described in this paper, acts on the one hand as a preprocessing component and, on the other hand – as a light-weight terminological matcher between labels. We start by splitting the elements of each ontology in three groups: labels of classes, labels of object properties and labels of data object properties (in colors blue, black and red in the figure), since these groups of elements are to be aligned separately. A standard preprocessing procedure is applied on these sets of labels, comprising character normalization, stop-words filtering, tokenization and lemmatization.

For the cross-lingual ontology matching, at first every token of a given label s in in the source ontology S is enriched by related terms and synonyms from BabelNet and all of these terms are represented in the language l_T(language of the target ontology), which makes these terms comparable to the tokens of the labels in the target ontology T. A simple similarity evaluation by the help of the Jaccard coefficient selects the term in each set of related terms corresponding to a given token from s that has the highest score with respect to every token in each label of T. This helps to restitute the label s in the language l_T. Finally,

the labels in each group of S and T, seen as sets of tokens, are compared by using the Soft TFIDF similarity measure [7], which produces an intermediate terminological alignment. For monolingual ontology matching, the system uses the relations such as "hasSynonyms" present in a given BK to match between two concepts.

The three remaining components are standard OM modules [6], although ordered in a new manner. The *Mapping selection* is a module that transforms the initial 1 to many mapping to a 1:1 mapping based on the principle of iteratively retaining the pairs of concepts with maximal value of similarity. Finally, the *structural matcher* component filters the trustworthy pairs of aligned concepts by looking at the similarity values produced for their parents and their children in the ontology hierarchies.

1.3 Adaptations made for the evaluation

The adaptation made for the evaluation is in the preprocessing step. LYAM++ uses (1) Uberon [8] for anatomy and BioMed tracks, (2) BabelNet [9] for conference and multifarm tracks.

1.4 Links to the System and to the Set of Provided Alignments

Last year, the system was not available online because it depends heavily on the use of BabelNet 3.0 version, which is under a non-free licence. In this year, we used old version of BabelNet 2.0 which is under free license.

The alignments produced by LYAM++ for this year's can be found under the following link: `http://www.lirmm.fr/benellefi/Alignements.rar`. LYAM++ can be found under the following link: `http://www.lirmm.fr/benellefi/Lyam++.rar`

2 Results

We have evaluated our approach on data coming from the ontology alignment evaluation initiative (OAEI)[2] and particularly anatomy, conference, large biomedical and multifarm.

Anatomy This track aims to discovering alignments between a human anatomy ontology, part of the NCI Thesaurus[3] and a mouse anatomy ontology. This track is considered as a large-scale matching task because the input ontologies are of a large size and very rich semantically. Table 1 presents the results obtained by LYAM++ on this year's

Conference This track contains 16 ontologies from the scientific publication field. Table 2 presents the results obtained by LYAM++ on this year's

[2] http://oaei.ontologymatching.org/
[3] https://ncit.nci.nih.gov/ncitbrowser/

Table 1: Results of LYAM++ for anatomy .

	F-M	Recall	Precision
LYAM++	0.87	0.88	0.86

Table 2: Results of LYAM++ for conference .

	F-M	Recall	Precision
ra1-M1	0.36	0.18	0.48
ra1-M2	0.34	0.57	0.13
ra1-M3	0.29	0.15	0.38
ra2-M1	0.36	0.19	0.52
ra2-M2	0.35	0.59	0.13
ra2-M3	0.31	0.16	0.41

Large biomedical ontologies This track aims at aligning three large biomedical ontologies, namely FMA, SNOMED and the NCI Thesaurus. Table 3 presents the results obtained by LYAM++ on this year's

Table 3: Results of LYAM++ for BioMed.

	F-M	Recall	Precision
Small FMA-NCI	0.79	0.88	0.72

MultiFarm is a benchmark designed for evaluating cross-lingual ontology matching systems. Multifarm data consist of a set of 7 ontologies originally coming from the *Conference* benchmark of OAEI, translated into 8 languages. Two evaluation tasks are defined: *task 1* consists in matching two different ontologies given in different languages, while *task 2* aims to align different language versions of one single ontology.

Table 4 presents the results obtained by LYAM++ on this year's Multifarm evaluation campaign. What we see is the average F-measure value for all language-pairs without any threshold on the confidence measure. The value in the parenthesis corresponds to the average F-measure value for the generated alignments only (the pairs of languages that the system handles).

3 Conclusion

In this paper, we present the over view of the LYAM++ system and our results on the OAEI2016 tracks . In this year, our goal was to participate on monolingual ontology matching scenarios. We used Babelnet 2.0 version instead of Babelnet 3.0 version due to the licenses problems. Subjects of ongoing and

Table 4: Results of LYAM++ for Multifarm.

	Task1	**Task2**
LYAM++	0.01	0.02

future work are (1) testing and evaluating different sources of external knowledge, (2) applying semantic mappings selection methods to improve the results, (3) adaptation of the approach to the large scale ontology matching scenarios.

References

1. P. Shvaiko and J. Euzenat, "Ontology matching: state of the art and future challenges," *Knowledge and Data Engineering, IEEE Transactions on*, vol. 25, no. 1, pp. 158–176, 2013.
2. D. Spohr, L. Hollink, and P. Cimiano, "A machine learning approach to multilingual and cross-lingual ontology matching," in *The Semantic Web–ISWC 2011*, pp. 665–680, Springer, 2011.
3. D. Faria, C. Pesquita, E. Santos, M. Palmonari, I. F. Cruz, and F. M. Couto, "The agreementmakerlight ontology matching system," in *On the Move to Meaningful Internet Systems: OTM 2013 Conferences*, pp. 527–541, Springer, 2013.
4. D. Ngo and Z. Bellahsene, "YAM++ : A multi-strategy based approach for ontology matching task," in *Knowledge Engineering and Knowledge Management - 18th International Conference, EKAW 2012, Galway City, Ireland, October 8-12, 2012. Proceedings*, pp. 421–425, 2012.
5. A. N. Tigrine, Z. Bellahsene, and K. Todorov, "Light-weight cross-lingual ontology matching with LYAM++," in *On the Move to Meaningful Internet Systems: OTM 2015 Conferences - Confederated International Conferences: CoopIS, ODBASE, and C&TC 2015, Rhodes, Greece, October 26-30, 2015, Proceedings*, pp. 527–544, 2015.
6. D. Ngo, Z. Bellahsene, and K. Todorov, "Opening the black box of ontology matching," in *The Semantic Web: Semantics and Big Data*, pp. 16–30, Springer, 2013.
7. W. W. Cohen, P. D. Ravikumar, and S. E. Fienberg, "A comparison of string distance metrics for name-matching tasks," in *IIWeb*, pp. 73–78, 2003.
8. M. Haendel, J. P. Balhoff, F. B. Bastian, D. C. Blackburn, J. A. Blake, Y. Bradford, A. Comte, W. M. Dahdul, T. Dececchi, R. E. Druzinsky, T. F. Hayamizu, N. Ibrahim, S. E. Lewis, P. M. Mabee, A. Niknejad, M. Robinson-Rechavi, P. C. Sereno, and C. J. Mungall, "Unification of multi-species vertebrate anatomy ontologies for comparative biology in uberon," *J. Biomedical Semantics*, vol. 5, p. 21, 2014.
9. R. Navigli and S. P. Ponzetto, "Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network," *Artif. Intell.*, vol. 193, pp. 217–250, 2012.

Integrating phenotype ontologies with PhenomeNET

Miguel Angel Rodríguez García[1], Georgios V Gkoutos[2], Paul N Schofield[3], and Robert Hoehndorf[1]

[1] Computational Bioscience Research Center, King Abdullah University of Science and Technology, Thuwal 23955-6900, KSA
{miguel.rodriguezgarcia,robert.hoehndorf}@kaust.edu.sa
[2] College of Medical and Dental Sciences, Institute of Cancer and Genomic Sciences, Centre for Computational Biology, University of Birmingham, Birmingham, B15 2TT, UK,
g.gkoutos@bham.ac.uk
[3] Department of Physiology, Development & Neuroscience, University of Cambridge, Downing Street, Cambridge, CB2 3EG, UK
pns12@hermes.cam.ac.uk

Abstract. PhenomeNET is a system for disease gene prioritization that includes as one of its components an ontology designed to integrate phenotype ontologies. While not applicable to matching arbitrary ontologies, PhenomeNET can be used to identify related phenotypes in different species, including human, mouse, zebrafish, nematode worm, fruit fly, and yeast. Here, we apply the PhenomeNET to identify related classes from four phenotype and disease ontologies using automated reasoning. We demonstrate that we can identify a large number of mappings, some of which require automated reasoning and cannot easily be identified through lexical approaches alone.

Keywords: PhenomeNET, phenotype ontology

1 System Presentation

1.1 State, purpose, general statement

PhenomeNET [1] was built in 2011 as a system for disease gene discovery and prioritization. PhenomeNET consists of an ontology integrating species-specific phenotype ontologies based on the PATO ontology [2] and relations between anatomical structures and physiological processes, a database of gene-to-phenotype associations, and a measure of similarity between sets of phenotypes. Within PhenomeNET, species-specific phenotype ontologies are combined so that phenotypes observed in different species can be compared directly. The main application of PhenomeNET is the prioritization of candidate genes for human diseases by comparing human disease phenotypes to existing gene-phenotype associations derived from model organisms. In particular, human phenotypes associated with a disease can be compared to phenotypes observed in mouse or

other model organisms using the integrated PhenomeNET ontology, and similarity between phenotypes can then be used to indicate the genetic basis of a disease. PhenomeNET has been successfully used to find candidate genes for diseases [1, 3], identify novel pathways [4], and repurpose drugs using mouse model phenotypes [5, 6].

Here, we use the PhenomeNET ontology to identify alignments between phenotypes in different species. We present three versions of the PhenomeNET ontology; the first version consists of the plain ontology using only the axioms provided in the Human Phenotype Ontology (HPO) [7] and the Mammalian Phenotype Ontology (MP) [8]; the second version uses additional lexical mappings and represents them as equivalent class axioms in the ontology; the third version further uses mappings generated by the AgreementMakerLight [9] to generate equivalent class axioms between classes in the PhenomeNET ontology and the Disease Ontology (DO) [10] and the Orphanet Rare Disease Ontology (ORDO) [11].

1.2 Specific techniques used

Phenotype classes in the HP and MP ontologies are formally defined using the Entity-Quality (EQ) pattern [2, 12]. Based on the EQ patterns, a phenotype is decomposed into an affected entity and a quality that specifies how the entity is affected. The Entity will usually be a class taken either from an anatomy ontology or a physiology ontology. For example, the phenotype class *macroglossia* (`HP:0000158`) describes an anatomical abnormality and is defined as equivalent to `'has part' some ('increased size' and ('inheres in' some tongue) and ('has modifier' some abnormal))`, relying on the entity *tongue* (from the UBERON anatomy ontology) and the quality *increased size* (from PATO) in its definition. The class *abnormality of salivation* (`HP:0100755`) is a physiological abnormality and is defined as equivalent to `'has part' some (quality and ('inheres in' some 'saliva secretion') and ('has modifier' some abnormal))`, where *saliva secretion* is a class from the biological process branch of the GO.

The general pattern for defining a phenotype class in both the HP and MP ontologies, given Entity E and Quality Q, is to declare them equivalent to `'has part' some (Q and 'inheres in' some E)`. In some cases, the Entity E is further constrained, e.g., by a location in which a certain process may happen. The "E" classes are generally taken either from the UBERON cross-species anatomy ontology [13] or from the GO. As the use of anatomy and physiology ontologies (UBERON and GO) is shared between MP and HP, it should be possible to integrate both ontologies directly, based on the axiom patterns used to constrain their classes. However, the type of axiom pattern used in both ontologies results in a classification that is primarily based on the PATO ontology, as the Quality Q is the main feature that distinguishes different classes.

In the PhenomeNET ontology, we rewrite all axioms in HP and MP using a pattern-based approach that allows us to utilize axioms from anatomy and physiology ontologies and enrich the classification of phenotype classes [14]. In general, we declare phenotype classes defined using an Entity E and Quality

Q as equivalent to `'has part' some (E and has-quality some Q)` and we further add grouping classes that are defined as equivalent to `'has part' some (('part of' some E) and has-quality some Q)`. The aim of rewriting the axioms is to base the classification of phenotype classes primarily on anatomical or physiological entities instead of the quality, and to utilize the axioms involving parthood in anatomy and physiology ontologies. Crucially, all axioms we generate fall in the OWL 2 EL profile [15]. The first version of the PhenomeNET ontology (PhenomeNET-Plain) consists only of these axioms and no additional mappings.

In addition to this knowledge-based approach to linking the HP and MP ontologies, we also add lexical mappings, mappings derived from cross-references in the ontologies [3], and mappings between HP and MP from BioPortal [16]. Each mapping is added as a single equivalent classes axiom to the first version of the ontology (PhenomeNET-Plain) to generate a version of the PhenomeNET ontology with mappings (PhenomeNET-Map).

Neither version of these ontologies contains the DO or ORDO ontologies, despite there being a significant overlap between the four ontologies. Since neither DO nor ORDO contain axioms that follow a similar pattern to the axioms in HP and MP, we rely exclusively on lexical mappings to integrate DO and ORDO. We use the AgreementMaker Light (AML) [9] in its default settings to generate mappings between HP and DO, HP and ORDO, MP and DO, MP and ORDO, and DO and ORDO. We then add an equivalent class axiom for each mapping AML identifies and that has a score by AML over greater than 0.7. The resulting ontology contains HP, MP, ORDO, and DO, and can be used to generate mappings between these ontologies.

All versions of the PhenomeNET ontology contain the classes from the HP and MP ontologies as well as the subclass axioms between named classes asserted in these ontologies. Furthermore, the PhenomeNET ontology imports the ChEBI [17] and Mouse Pathology [18] ontologies using an OWL import statement. Additionally, PhenomeNET includes all classes from the UBERON anatomy ontology [13], the Gene Ontology [19], the BioSpatial Ontology [20], the Zebrafish Anatomy ontology [21], the PATO ontology [2], the Cell Ontology [22], and the Neuro-Behavior Ontology [23]. However, these ontologies are not directly imported but rather pre-processed so that all disjointness axioms from these ontologies are excluded while all other axioms contained within them are included in the PhenomeNET ontology. The aim of this pre-processing step is to avoid unsatisfiable classes due to different conceptualizations between anatomy and phenotype ontologies, or within anatomy ontologies (Zebrafish Anatomy and UBERON).

Mappings between ontologies included in PhenomeNET are generated using the ELK reasoner [24]. We use ELK to classify the PhenomeNET ontology and identify pairs of equivalent classes C_1 and C_2 that belong to the ontologies to be aligned. These constitute equivalent class mappings. Furthermore, subclass and superclass mappings are generated through queries for sub- and superclasses using ELK.

Ontology	Number of classes	Number of axioms	Mappings added
HP-MP	219,423	1,399,411	0
HP-MP+mappings	219,423	1,400,570	1,160(AML), 639(BioPortal)
HP-MP+DO-ORDO	241,817	1,631,543	1489(AML), 1018(BioPortal) HP-MP: 1,160 (AML), 639(BioPortal); DO-MP: 423 (AML); DO-HP: 1,074; ORDO-MP: 151; ORDO-HP: 531;

Table 1. Number of classes, axioms and mappings in the PhenomeNET ontologies

1.3 Adaptations made for the evaluation

Within PhenomeNET, we use an ontology consisting only of the (rewritten) axioms in MP and HP as well as equivalent class axioms derived from explicit mappings between HP and MP (expressed as `xref` annotation properties). For the evaluation, we further used the AML [9] to generate additional mappings. The AML mappings were generated using the default settings of AML with a confidence cutoff of 0.7. In the case of DOID and ORDO mappings we additionally included 18 mappings derived from BioPortal. Our systems relying on these mappings were submitted as separate submissions.

Initially, we developed our matching system to take into account not only the direct sub- and super-classes, but also all inferred classes. We modified our system to output only the most specific mappings instead for the evaluation; Table 2 shows both the number of direct and inferred mappings.

1.4 Link to the system, parameters file, alignments

Our submission consists of two modules: PhenomeNetBridge and PhenomeNetMatcher. The PhenomeNetBridge module wraps the SEALS infrastructure for the evaluation, and the PhenomeNetMatcher module performs the mappings, using one of three ontologies. Source code for the matching system, including parameter files, and the generated alignments, are available at http://github.com/bio-ontology-research-group/OAEI2016. Code to generate the PhenomeNET ontology is available at
https://github.com/bio-ontology-research-group/phenomeblast/tree/master/fixphenotypes.

2 Results

2.1 Phenotype ontologies: HP and MP

The PhenomeNET ontology is primarily intended to integrate the HP and MP ontologies. Using the axioms in the ontology alone (PhenomeNET-Plain submission), we identify 745 equivalent classes between the HP and MP ontologies

Ontology	HP-MP (≡)	HP-MP (⊑)	DO-ORDO (≡)	DO-ORDO (⊑)
HP-MP	745	2,707 (96,278)	0	0
HP-MP+mappings	1,536	3,999 (107,268)	0	0
HP-MP+DO-ORDO	1,582	4,144 (112,366)	1,527	4,576 (16,838)

Table 2. Equivalent and sub-equivalent classes found in the experiments

Ontology	Precision	Recall	F-Measure	Found	Correct	Reference
HP-MP task						
HP-MP	3.90 %	40.80%	7.10%	6,730	261	639
HP-MP+mappings	6 %	100 %	11.30%	10,698	639	639
HP-MP+DO-ORDO	5.80 %	100 %	10.90%	11,086	639	639
DOID-ORDO task						
HP-MP	0 %	0 %	0 %	0	0	1,018
HP-MP+mappings	0 %	0 %	0 %	0	0	1,018
HP-MP+DO-ORDO	12.70 %	99.90 %	22.50 %	8,036	1,017	1,018

Table 3. Precision, Recall, F-measure in HP-MP and DOID-ORDO experiments

(see Table 2). These correspond to a recall of 40.8% with respect to the reference mappings provided (see Table 3). Additionally, a large number of sub- and super-class mappings can be identified based on querying the ontology using the ELK reasoner [24] for sub- or super-classes in the two ontologies.

The number of pairs of equivalent classes identified increases to 1,536 when adding explicit mappings derived from AML. Of these, 370 are generated both by automated reasoning and are included in AML, 791 are generated from the AML-derived equivalent classes axioms, and 375 could only be derived through the automated reasoning. Total recall with respect to the reference mappings is 100% in this version of PhenomeNET. Additionally, we observe an improvement in the number of equivalent class mappings when adding the ORDO and DO ontologies to the PhenomeNET ontology. The increase in mappings (from 1,536 to 1,582 classes) is a result of additional inferences obtained from adding the mappings from HP and MP to ORDO and DO, and combining them with the axioms in the PhenomeNET ontology. For example, we infer a new mapping between *decreased IgG level* (`MP:0001805`) and *agammaglobulinemia* (`HP:0004432`) based on the equivalence axioms between both classes and *agammaglobulinemia* (`DOID:2583`) generated by AML (based on the shared synonym "hypogammaglobulinemia" between the class in DO and MP). Table 3 summarizes our results with respect to the reference mappings provided in the challenge.

2.2 Disease ontologies: ORDO and DO

PhenomeNET is primarily designed for ontologies that follow the Entity-Quality definition pattern based on the PATO ontology. Neither ORDO nor DO follow this pattern, and ORDO and DO are primarily included in the PhenomeNET ontology through equivalent class axioms based on lexical mappings generated

by AML. We achieve a recall of 99.9% with the PhenomeNET-Full ontology. Notably, the mappings we generate are increased by including HP and MP. For example, we identify a mapping between *mandibulofacial dysostosis* (`ORPHANET:155899`) and *treacher collins syndrome* (`DOID:2908`), based on common AML-generated mappings to *mandibulofacial dysostosis* (`HP:0005321`).

2.3 OAEI evaluation

In order to carry out the final evaluation, the OAEI utilized the SEALS infrastructure executed in a Ubuntu Laptop with an Intel Core i7-4600U CPU @ 2.10GHz x 4 and allocating 15Gb of RAM. The system carried out the evaluation according to following criteria:

- Precision and Recall with respect to a voted reference alignment automatically generated by merging/voting the outputs of the participating systems.
- Recall with respect to alignment manually generated.
- Manual assesment of a subset of generating mappings.
- Performance in other tracks.

Different mappings were used to evaluate the participating systems: i) Silver standard with vote 2, ii) Silver standard with vote 3, iii) manually dataset and manual assessment. In the first dataset, PhenomeNET including all mappings reached an F-measure of 0.82 in the HP-MP task, and 0.89 in the DO-ORDO task. In the second evaluation, although the system PhenoMP was able to find the largest number of mappings in HP-MP task, it reached an F-measure of 0.76 in the HP-MP task and 0.94 in the DO-ORDO task. When evaluating against manually created mappings, PhenomeNET achieved a recall of 0.897 in the HP-MP task but could not generate any new mappings between DO and ORDO. For this task, PhenomeNET achieved a precision of 1.0 in the manual assessment of a subset of the generated mappings.

3 General comments

3.1 Comments on the results

PhenomeNET is a system to match phenotypes; as such, it is not a system that can be applied to match ontologies in general. The axiom-based approach in PhenomeNET can be applied to any ontologies that utilize PATO and the Entity-Quality definition patterns [2]. In particular, PhenomeNET can not only be used to integrate MP and HPO, but also has been used to further integrate yeast, fly, worm, slime mold, and fish phenotypes [1, 25]. Furthermore, the combination of semantic matching (using automated reasoning) and lexical matching in PhenomeNET mitigates some of the limitations of using lexical approaches alone, and we demonstrate this by inferring several hundred mappings between HP and MP that cannot be inferred using AML.

However, relying on manually created axioms also has several limitations. In particular, the axioms are created by domain experts, and only about half

the classes in MP and HP are constrained by an Entity-Quality based axiom. Furthermore, the quality of the axioms is difficult to assess, and there are distinct differences between HP and MP in how the classes are constrained.

3.2 Discussions on the way to improve the proposed system

One of the main limitations in PhenomeNET is the need for manually created axioms that constrain classes in phenotype ontologies. A possible solution to this approach would be to generate phenotype ontologies fully automatically using anatomy and physiology ontologies as templates and applying the axiom patterns we use in the PhenomeNET [26].

Another limitation of PhenomeNET is the reliance on OWL 2 EL which limits the expressivity of axiom patterns. The choice is mainly due to the size of the PhenomeNET ontology and the complexity of reasoning. However, more complex axiom patterns would enable more comprehensive classification of phenotypes involving absences and abnormalities [14]; experiments with an updated ontology will likely require improvement in OWL reasoning technologies.

4 Conclusions

We have developed an ontology matching system for disease and phenotype ontologies. We generated three different version of the PhenomeNet ontology, each with different information and ontologies included. PhenomeNET is primarily based on deductive inference and automated reasoning, and while it can utilize lexically derived mappings in the ontology generation process, it does not on its own include any lexical matching algorithms. Our results demonstrate that a combination of lexical and semantic approaches may improve upon mappings between ontologies generated using only one of these methods.

References

1. Hoehndorf, R., Schofield, P.N., Gkoutos, G.V.: Phenomenet: a whole-phenome approach to disease gene discovery. Nucleic Acids Res **39**(18) (2011) e119
2. Gkoutos, G.V., Green, E.C., Mallon, A.M.M., Hancock, J.M., Davidson, D.: Using ontologies to describe mouse phenotypes. Genome biology **6**(1) (2005) R5
3. Hoehndorf, R., Schofield, P.N., Gkoutos, G.V.: An integrative, translational approach to understanding rare and orphan genetically based diseases. Interface Focus **3**(2) (2013) 20120055
4. Hoehndorf, R., Dumontier, M., Gkoutos, G.V.: Identifying aberrant pathways through integrated analysis of knowledge in pharmacogenomics. Bioinformatics **28**(16) (2012) 2169–2175
5. Hoehndorf, R., Oellrich, A., Rebholz-Schuhmann, D., Schofield, P.N., Gkoutos, G.V.: Linking PharmGKB to phenotype studies and animal models of disease for drug repurposing. Pacific Symposium on Biocomputing (PSB) (2012) 388–399
6. Hoehndorf, R., Hiebert, T., Hardy, N.W., Schofield, P.N., Gkoutos, G.V., Dumontier, M.: Mouse model phenotypes provide information about human drug targets. Bioinformatics **30**(5) (2014) 719–725

7. Köhler, S., Doelken, S.C., Mungall, C.J., Bauer, S., Firth, H.V., Bailleul-Forestier, I., Black, G.C.M., Brown, D.L., Brudno, M., Campbell, J., FitzPatrick, D.R., Eppig, J.T., Jackson, A.P., Freson, K., Girdea, M., Helbig, I., Hurst, J.A., Jähn, J., Jackson, L.G., Kelly, A.M., Ledbetter, D.H., Mansour, S., Martin, C.L., Moss, C., Mumford, A., Ouwehand, W.H., Park, S.M., Riggs, E.R., Scott, R.H., Sisodiya, S., Vooren, S.V., Wapner, R.J., Wilkie, A.O.M., Wright, C.F., Vulto-van Silfhout, A.T., Leeuw, N.d., de Vries, B.B.A., Washingthon, N.L., Smith, C.L., Westerfield, M., Schofield, P., Ruef, B.J., Gkoutos, G.V., Haendel, M., Smedley, D., Lewis, S.E., Robinson, P.N.: The human phenotype ontology project: linking molecular biology and disease through phenotype data. Nucleic Acids Res **42**(D1) (2014) D966–D974
8. Smith, C.L., Goldsmith, C.A.W., Eppig, J.T.: The mammalian phenotype ontology as a tool for annotating, analyzing and comparing phenotypic information. Genome Biol **6**(1) (2004) R7 DOI:10.1186/gb-2004-6-1-r7.
9. Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I.F., Couto, F.M. In: The AgreementMakerLight Ontology Matching System. Springer Berlin Heidelberg, Berlin, Heidelberg (2013) 527–541
10. Kibbe, W.A., Arze, C., Felix, V., Mitraka, E., Bolton, E., Fu, G., Mungall, C.J., Binder, J.X., Malone, J., Vasant, D., Parkinson, H., Schriml, L.M.: Disease ontology 2015 update: an expanded and updated database of human diseases for linking biomedical knowledge through disease data. Nucleic Acids Res **43** (2014) D1071–D1078
11. Sarntivijai, S., Vasant, D., Jupp, S., Saunders, G., Bento, A.P., Gonzalez, D., Betts, J., Hasan, S., Koscielny, G., Dunham, I., Parkinson, H., Malone, J.: Linking rare and common disease: mapping clinical disease-phenotypes to ontologies in therapeutic target validation. Journal of Biomedical Semantics **7**(1) (2016) 1–11
12. Mungall, C., Gkoutos, G., Smith, C., Haendel, M., Lewis, S., Ashburner, M.: Integrating phenotype ontologies across multiple species. Genome Biol **11**(1) (2010) R2+
13. Mungall, C., Torniai, C., Gkoutos, G., Lewis, S., Haendel, M.: Uberon, an integrative multi-species anatomy ontology. Genome Biology **13**(1) (2012) R5
14. Hoehndorf, R., Oellrich, A., Rebholz-Schuhmann, D.: Interoperability between phenotype and anatomy ontologies. Bioinformatics **26**(24) (10 2010) 3112 – 3118
15. Motik, B., Grau, B.C., Horrocks, I., Wu, Z., Fokoue, A., Lutz, C.: Owl 2 web ontology language: Profiles. Recommendation, World Wide Web Consortium (W3C) (2009)
16. Noy, N.F., Shah, N.H., Whetzel, P.L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D.L., Storey, M.A.A., Chute, C.G., Musen, M.A.: Bioportal: ontologies and integrated data resources at the click of a mouse. Nucleic acids research **37**(Web Server issue) (July 2009) W170–173
17. Degtyarenko, K., Matos, P., Ennis, M., Hastings, J., Zbinden, M., McNaught, A., Alcantara, R., Darsow, M., Guedj, M., Ashburner, M.: ChEBI: a database and ontology for chemical entities of biological interest. Nucleic Acids Research (2007)
18. Schofield, P.N., Sundberg, J.P., Sundberg, B.A., McKerlie, C., Gkoutos, G.V.: The mouse pathology ontology, mpath; structure and applications. Journal of Biomedical Semantics **4**(1) (2013) 1–8
19. Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, M.J., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Tarver, L.I., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M., Sherlock, G.: Gene ontology: tool for the unification of biology. Nature Genetics **25**(1) (May 2000) 25–29

20. Balhoff, J.P., Mik, I., Yoder, M.J., Mullins, P.L., Deans, A.R.: A semantic model for species description applied to the ensign wasps (hymenoptera: Evaniidae) of new caledonia. Systematic Biology **62**(5) (2013) 639–659
21. Dahdul, W.M., Balhoff, J.P., Engeman, J., Grande, T., Hilton, E.J., Kothari, C., Lapp, H., Lundberg, J.G., Midford, P.E., Vision, T.J., Westerfield, M., Mabee, P.M.: Evolutionary characters, phenotypes and ontologies: curating data from the systematic biology literature. PLoS One **5**(5) (2010) e10708
22. Bard, J., Rhee, S.Y., Ashburner, M.: An ontology for cell types. Genome Biology **6**(2) (2005)
23. Hoehndorf, R., Hancock, J.M., Hardy, N.W., Mallon, A.M., Schofield, P.N., Gkoutos, G.V.: Analyzing gene expression data in mice with the Neuro Behavior Ontology. Mamm Genome **25**(1-2) (2014) 32–40
24. Kazakov, Y., Krötzsch, M., Simancik, F.: The incredible elk. Journal of Automated Reasoning **53**(1) (2014) 1–61
25. Hoehndorf, R., Hardy, N.W., Osumi-Sutherland, D., Tweedie, S., Schofield, P.N., Gkoutos, G.V.: Systematic analysis of experimental phenotype data reveals gene functions. PLoS ONE **8**(4) (04 2013) e60847
26. Hoehndorf, R., Harris, M.A., Herre, H., Rustici, G., Gkoutos, G.V.: Semantic integration of physiology phenotypes with an application to the cellular phenotype ontology. Bioinformatics **28**(13) (2012) 1783–1789

RiMOM Results for OAEI 2016

Yan Zhang, Hailong Jin, Liangming Pan, Juanzi Li

Tsinghua University, Beijing, China.
{z-y14, jinhl, panlm14}@mails.tsinghua.edu.cn
ljz@keg.tsinghua.edu.cn

Abstract. This paper presents the results of RiMOM in the Ontology Alignment Evaluation Initiative (OAEI) 2016. RiMOM participated in all three tracks of Instance Matching this year. In this paper, we first describe the overall framework of our system (RiMOM). Then we detail the techniques used in the framework for instance matching. Last, we give a thorough analysis on our results and discuss some future work on RiMOM.

1 Presentation of the system

With the rapid development of the Semantic Web, knowledge base has become a dominant mechanism to represent the data semantics on the Web. In practice, data is always distributed on heterogeneous data sources. For example, there are a large number of ontological knowledge bases nowadays, such as DBpedia[1]. , YAGO [2, 3], Xlore [4], etc. It is inevitable that the knowledge about the same real-world entity may be stored in different knowledge bases. Therefore, data integration process requires the detection of such heterogeneous instances to ensure the integrity and consistency.

Most recently, it should be noticed that there are many knowledge bases described in different languages. For example, Wikipedia, a well-known public encyclopedia, contains 281 language versions. It is going to be norm that the same real-world entities are described by different language. Thus, there is a growing need to align instances in a cross-lingual environment so that we can share knowledge from all over the world. In consideration of this circumstance, based on previous version of RiMOM[5], we propose an extended version, which provides support for cross-lingual instance matching in a supervised or an unsupervised way.

There are three major techniques in our system, blocking, multi-strategy, machine learning:

1. **Blocking:** We index the instances based on their objects in two knowledge bases respectively, and then select the instances which contain the same keys as candidate instance pairs. We limit the number of pairs to be compared by this step, which significantly improve the efficiency of the system.
2. **Multi-strategy:** We implement several matchers in our instance matching system, we can execute these matchers in parallel and then aggregate the result according to the characteristics of the source ontologies.
3. **Machine learning:**In general, there are some existing alignments. For exmaple, there are a number of cross-lingual links between two different language versions

of Wikipedia. To make full use of these data, we formalize the instance matching as a binary classification problem, and use the reference mappings to train a classifier, which will determine whether an instance pair is equivalent or not.

Faced with challenges in large-scale instance matching, we propose an novel data integration framework RiMOM-2016 (the latest version of RiMOM), which is based on our former ontology and instance matching system RiMOM [5, 6]. The RiMOM-2016 framework is designed for large-scale and cross-lingual instance matching task specially. It presents a novel multi-strategy method to be fit for different kinds of ontology and employs a learning-based approach to get instance alignments in multilingual environments.

1.1 State, purpose, general statement

This section describes the overall framework of RiMOM2016. The overview of the instance matching system is shown in Fig. 1. The system includes seven modules, i.e., *Preprocess*, *Predicate Alignment*, *Mathcher Choosing*, *Candidate Pair Generation*, *Matching Score Calculation*, *Instance Alignment* and *Validation*. The sequences of the process are shown in the Fig. 1. We illustrate the process as follows.

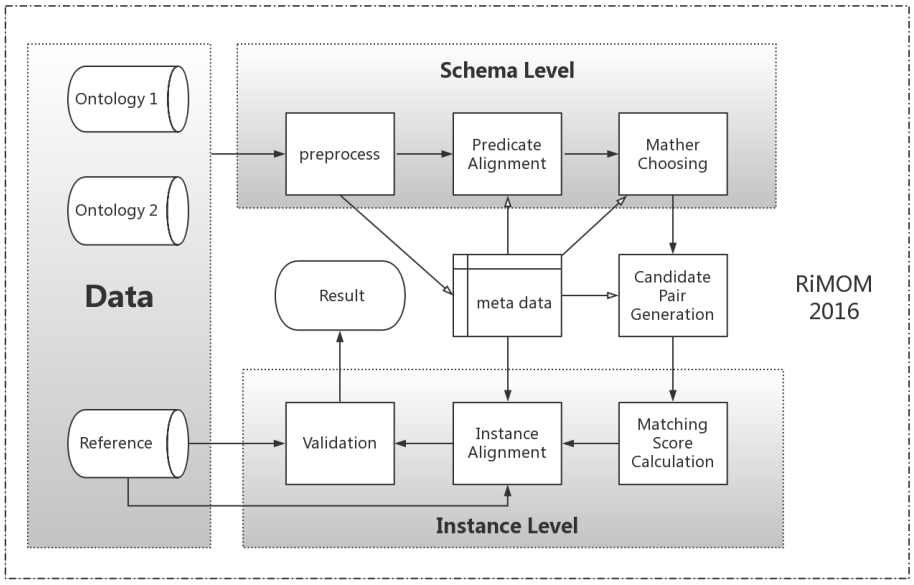


Fig. 1. Framework of RiMOM 2016

1. **Preprocess:** The system begins with *Preprocess*, which loads the ontologies and parameters into system. In the meantime, preprocessor can get some meta data about the two ontologies, which will be used in the later processes, *Predicate alignment* and *Matcher choosing*
2. **Predicate Alignment:** In this process, we will get the alignments of the predicates between the two ontologies.
3. **Matcher choosing:** The system will choose the most suitable one or more matchers according to the meta data of the ontologies.
4. **Candidate Pairs Generation:** In this step, we get candidate pairs when the instances have the same literal objects on some discriminatory predicates.
5. **Matching Score Calculation & Instance Alignment:** This procedure is the most striking difference with the last version of RiMOM. In RiMOM-2016, we get alignments in a supervised or an unsupervised way which depends on whether there exist reference alignments or not. In case of unsupervised method, we calculate similarities between two instances on each property, and then we aggregate these similarities according to the degree of identifying obtained in step 1. On the contrary, we conduct a supervised method when there exist reference alignments. For each instance pairs, we also calculate the similarities as unsupervised way. Then we construct a similarity vector for each pairs and train a logistic regression model [7]. For each candidate instance pair, we use this model to determine whether it is equivalent or not.
6. **Validation:** We will evaluate the alignment result on Precision, Recall and F1-Measure if there is validation data set.

1.2 Specific techniques used

This year we participate in all of three subtasks in the Instance Matching track. We will describe specific techniques in this section.

Data Preprocessing: First, we remove some stop words like ”a, of, the”, etc. Afterwards, we calculate the TF-IDF values of words in each knowledge base. We also calculate some information of each predicate, in order to obtain the degree of identifying of predicates which will be used in similarity aggregation.

Predicate Alignment: The predicates can express rich semantics, and there exist one-to-one, one-to-many, or many-to-many relationships among these predicates. It is apparent that we should get the alignments of the predicates before we calculate the similarity of instances. In RiMOM-2016, we use an object-based method to align predicates, which is similar with RiMOM-2015 [5].

Blocking: This step aims to pick a relatively small set of candidate pairs from all pairs. Due to the large scale of knowledge bases, it is impossible to calculate matching scores of all instance pairs. In our method, we firstly generate the inverted index on the objects. instance pairs are selected into the candidate set when they have common objects. This method may reduce the recall slightly, but it also reduce the scale of computation significantly.

Multi-Strategy: We implement several matchers in our system, e.g. label-based approach and structure-based approach. In the preprocess step, we will compare the

schema of the two ontologies. If the range of predicates is similar, the label-based approach will play a key role in the matching process. Otherwise, the literal properties are not similar (e.g. the two ontologies are defined in different languages or the intersection of values is really small), label-based approach will not be effective. In this case, we will get some supplementary information (e.g. machine translation, WordNet), or use structure-based appraoch (or use the structure similarity as a feature). In addition, we will use a learning-based method if we have data for training.

Similarity Calculation & Instance Alignment: In OAEI 2016 instance matching track, some of subtasks are defined in the same language, while others use multilingual data sets (e.g. SABINE Task).

Unsupervised method: we use a object-based method to get alignments, it is defined as follows:

$$f_{p_n}(i_1, i_2) = Sim(O_{i_1}^{p_n}, O_{i_2}^{p_n^{'}}) \tag{1}$$

where i_1 and i_2 are instances from two data sets respectively. $O_{i_1}^{p_n}$ represent the object value of instance i_1 on property p_n. $Sim(O_{i_1}^{p_n}, O_{i_2}^{p_n^{'}})$ represent the similarity of object values between these two instances on property p_n and its corresponding property $p_n^{'}$. The computing method of this similarity depends on the data type. For example, we use Levenshtein distance for *type:text* and indicator function for *type:int*.

$$Sim(i_1, i_2) = \omega_1 \times f_{p_1}(i_1, i_2) + \omega_2 \times f_{p_2}(i_1, i_2) + ... + \omega_n \times f_{p_n}(i_1, i_2) \tag{2}$$

For each property p_j, we calculate the similarity according to equation 1 and aggregate them by weights ω_j which indicate the importance of properties.

Supervised method: In equation 2, the weight w_i is determined by meta-data of ontology or manual. Intuitively, it could be improved by a learning-based method if we have some existing alignments. So, basically, we formulate this instance matching problem as a binary classification problem. For a pair of instance i_1 and i_2, the feature vector $\boldsymbol{f} = \{f_{p_i}\}_{i=1}^{n}$. Thus, we can use a sigmoid function to compute the probability that instances i_1 is equivalent with i_2.

$$P(i_1 \equiv i_2) = \frac{1}{1 + e^{\boldsymbol{w} \cdot \boldsymbol{f}(i_1, i_2)}} \tag{3}$$

If $i_1 \equiv i_2$, $P(i_1 \equiv i_2) > 0.5$; otherwise $P(i_1 \equiv i_2) < 0.5$ In this case, the weights $\boldsymbol{w}$ can be determined by the maximum likelihood estimation technique for logistic regression. The assumption in this model is that we can use the machine learning method to determine which property is more important for instance matching problem.

1.3 Link to the system and parameters file

The RiMOM system and configuration files (2016 version) can be found at `https://drive.google.com/file/d/0BzqVVt4Q8YUuaHpseWJOZkI4MnM/view?usp=sharing`.

2 Results

The Instance Matching track contains three tracks and seven subtasks. RiMOM-2016 participate in all of these tracks, and we will present the results and related analysis in this section.

2.1 SABINE Track

There are two subtasks in this track: Inter-linguistic mapping and Data linking. **Table 1** is the result for *Inter-linguistic mapping* task and **Table 2** is for *Data linking* task. Inter-linguistic mapping is a cross-lingual task between English and Italian. As shown in the result, RiMOM preform well in this task. Data linking task requires participants to link the entity to DBpedia, and RiMOM get high Recall but low Precision in this task.

Tool	Precision	Recall	F-measure
LogMapIm	0.012	0.016	0.014
AML	0.919	0.916	0.917
LogMapLite	0.358	0.153	0.214
RiMOM	0.955	0.932	0.943

Table 1. The result for Inter-linguistic mapping

Tool	Precision	Recall	F-measure
LogMapIm	NaN	0.000	NaN
AML	0.926	0.855	0.889
LogMapLite	NaN	0.000	NaN
RiMOM	0.424	0.917	0.580

Table 2. The result for Data linking

2.2 SYNTHETIC Track

There are two subtasks in this track: UOBM and SPIMBENCH. Each subtask contains two data set in different size: *sandbox* is small data set while *mainbox* is a large one. **Table 3 4 5 6** show the final results in this track. We think RiMOM produce satisfactory results in all of the subtasks.

Tool	Precision	Recall	F-measure
LogMapIm	0.701	0.207	0.320
AML	0.785	0.577	0.665
RiMOM	0.771	0.877	0.821

Table 3. The result for UOBM sandbox

Tool	Precision	Recall	F-measure
LogMapIm	0.625	0.023	0.044
AML	0.509	0.515	0.512
RiMOM	0.443	0.516	0.477

Table 4. The result for UOBM mainbox

2.3 DOREMUS Track

This track contains three subtasks: *9-heterogeneities*, *4-heterogeneities*, and *False Positive Trap*. **Table. 7** shows the final result in this track.

Tool	Precision	Recall	F-measure
LogMapIm	0.958	0.766	0.851
AML	0.907	0.749	0.820
RiMOM	0.984	1.000	0.992

Table 5. The result for SPIMBENCH sandbox

Tool	Precision	Recall	F-measure
LogMapIm	0.981	0.695	0.814
AML	0.900	0.747	0.816
RiMOM	0.991	1.000	0.995

Table 6. The result for SPIMBENCH mainbox

Sub-task	Precision	Recall	F-measure
9-heterogeneities	0.813	0.813	0.813
4-heterogeneities	0.746	0.746	0.746
False Positive Trap	0.707	0.707	0.707

Table 7. The result for DOREMUS Track

2.4 Discussions on the way to improve the proposed system

Our system can only align two ontologies at a time, and we think it will be a significant improvement if we can develop a system which is able to align several ontologies simultaneously. In addition, in cross-lingual environment, our system still rely on the machine translation. In this case, we hope to develop a method which is language-independent.

3 Conclusion and future work

In this paper, we present the system of RiMOM in OAEI 2016 Campaign. We participate all of the three tracks in instance matching track this year. We described specific techniques we used in the task. In our project, we design a new framework to align instances in different languages. The results turn out that our method is effective.

In the future, we will make great efforts to improve our system continuously.

4 Acknowledgement

The work is supported by 973 Program (No.2014CB340504), NSFC-ANR (No.61261130588),and NSFC key project(No.61533018), Tsinghua University Initiative Scientific Research Program (No.20131089256) and THU-NUS NExT Co-Lab.

References

1. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: Dbpedia - A crystallization point for the web of data. J. Web Sem. **7**(3) (2009) 154–165
2. Hoffart, J., Suchanek, F.M., Berberich, K., Weikum, G.: YAGO2: A spatially and temporally enhanced knowledge base from wikipedia. Artif. Intell. **194** (2013) 28–61
3. Mahdisoltani, F., Biega, J., Suchanek, F.: Yago3: A knowledge base from multilingual wikipedias. In: 7th Biennial Conference on Innovative Data Systems Research, CIDR Conference (2014)
4. Wang, Z., Li, J., Wang, Z., Li, S., Li, M., Zhang, D., Shi, Y., Liu, Y., Zhang, P., Tang, J.: Xlore: A large-scale english-chinese bilingual knowledge graph. In: Proceedings of the ISWC 2013 Posters & Demonstrations Track, Sydney, Australia, October 23, 2013. (2013) 121–124
5. Zhang, Y., Li, J.: Rimom results for oaei 2015. Ontology Matching (2015) 185
6. Li, J., Tang, J., Li, Y., Luo, Q.: Rimom: A dynamic multistrategy ontology alignment framework. IEEE Trans. Knowl. Data Eng. **21**(8) (2009) 1218–1232
7. Hosmer, D.W., Lemeshow, S.: Introduction to the logistic regression model. Applied Logistic Regression, Second Edition (2000) 1–30

SimCat Results for OAEI 2016

Abderrahmane Khiat[1], Elhabib Abdelillah Ouhiba[2], Mohammed Amine Belfedhal[3] and Chihab Eddine Zoua[4]

[1]LITIO Laboratory, University of Oran1 Ahmed Ben Bella, Oran, Algeria
[2]LAMOSI Laboratory, Oran University of Science and Technology - Mohamed Boudiaf
[3]EEEDIS Lab, University Djillali Liabes, Sidi Bel-Abbes, Algeria
[4]Ooredoo Algiers, Algeria
Email: abderrahmane_khiat@yahoo.com, ouhiba.ab@gmail.com,
Mohammed.belfedhal@gmail.com, z.chiheb.e@gmail.com

Abstract. Recently, the multilingualism issue has attracted considerable attention in the ontology matching field. Designed for this purpose, the SimCat system uses the Yandex translator and similarity computation based on the categories of the words. This is the first participation of SimCat in OAEI 2016 evaluation campaign and the obtained results are quite promising.

1 Presentation of the System

The Semantic Web relies on ontologies to describe the content of different information sources in order to overcome the heterogeneity issue and achieve their semantic interoperability [12, 14]. However, these ontologies are heterogeneous, distributed and even they are described in different languages. A solution to this heterogeneity is to use ontology alignment to bridge the semantic gap between these ontologies [11]. The ontology alignment system receives as input two or more ontologies and generates as output a set of semantic correspondences between the entities of the ontologies that are being processed [3, 2]. Indeed, these semantic correspondences are the bridges that hold the heterogeneous ontologies together and ensure their semantic interoperability. Moreover, with the enormous volume of ontologies already available on the web and their constant evolution, manual identification of semantic correspondences is not feasible [14]. Therefore, ontology alignment tools are required to have the ability of identifying semantic correspondences between entities of different ontologies in an automated way. However, the automatic identification of semantic correspondences is not a trivial task due to the conceptual diversity between the ontologies [4].

Performing an automatic ontology alignment task between mono-language ontologies such as English is difficult, however, the task is even more challenging when it comes to multilingual ontologies. Most existing approaches implement a direct strategy[15] i.e. using machine translation. However, the matching task is challenging for these approaches due to misinterpretations during the translation process.

The research conducted on direct strategy leaves many questions to address such as (1) is the use of various translators has a different impact on the output of the translation? (2) is the translation into a pivot language (English) performing better output than

a translation from language to another? and (3) how to proceed when translators give poor results?

The multifarm[10] track has been integrated in the Ontology Alignment Evaluation Initiative (OAEI) in 2012 with the goal of estimating and comparing different techniques and systems related to multilingual ontology alignment. From 2012 to 2014 the multifarm track contains conference ontologies[9] described in eight different languages (i.e., Chinese, Czech, Dutch, French, German, Portuguese, Russian, Spanish). However, in 2015 the multifarm includes the Arabic language [13, 14].

Back to results of the systems involved in previous editions (from 2012 to 2015) [5–8] of multifarm track, we have observed that the best system (in all previous OAEI editions) achieved an F-measure of 0.51 [15]. This is surprising, in spite of many research works that have been established in the field of multilingual ontology matching.

The proposed system also implements a direct strategy and its aim is to highlight the translator used and similarity calculated using the categories of the word.

1.1 State, Purpose, General Statement

In this paper, we describe our SimCat software, yet another cross-lingual ontology matching system. Unlike existing approaches which use well-known translators, SimCat employs the Yandex translatorr[1]. In addition, SimCat computes the similarities between translated entities based on the categories of the words.

1.2 Specific Techniques Used

The process of our system consists in the following successive steps.

Step 1: Extraction and Normalization In this step, our system extracts the entities of two ontologies to align. Then, it uses a segmentation technique to split labels into words; Finally, it converts all words in lower case.

Step 2: Translation and Cleaning In this step, SimCat translates the normalized entities using the Yandex translator into English as a pivot language. To the best of our knowledge, the Yandex translator has not been used before by multilingual ontology matching system. Our choice of Yandex translator is justified by the fact that it is one of largest search engine in the world and the obtained results are quite promising. However, we have used the English as a pivot language because the categories of the words which are used for similarity computation are in English language.

Once the translation is is carried out, SimCat employs NLP techniques. First, it eliminates the stop-words from translated entities; then it employs lemmatization and stemming. This step is necessary since the categories of the words are in that lemma form.

[1] `https://translate.yandex.com/?lang=es-en&text=administrar&ncrnd=5317`

Step 3: Similarity Computation In this step, our system computes the similarity between entities using the categories of words. This matcher is based on an open project named "Calculate Semantic Similarity".

The project[2] calculates the similarities between sentences and the results are stable. The description of the project is as follows: First, the list of words was obtained from using EOWL, then the categories for each word were calculated using the DISCO's semantics[3]. The semantic categories are obtained from disco as follows: (1) en-BNC-20080721 within 119 million tokens; (2) en-PubMedOA-20070501 within 181 million tokens and (3) en-wikipedia-20080101 within 267 million tokens. The matcher enhances the Vector-Space by the analysis found withing the Classifier4j, which does not take into account the semantic meanings of the words.

However, we have adapted it for our case. We have reprogram the matcher in a way that it can return the similarity value between words. We have some tests on the adapted matcher and the results are quite good.

Step 4: Identification of Alignment In this step, SimCat applies applies a filter to select candidate correspondences which possess the maximum similarity value in each line of Cartesian product between entities. Then it applies a second a filter to identify the correspondences that possess similarity value upper than a given threshold.

1.3 Adaptations Made for the Evaluation

We do not have made any specific adaptation for OAEI 2016 evaluation campaign regarding our SimCat system. All parameters are the same for aligning different ontologies of multifarm track.

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

The result of SimCat system can be downloaded from OAEI 2016 website `http://oaei.ontologymatching.org/2016/results/multifarm/index.html`

2 Results

The SimCat system is yet another multilingual ontology alignment system. Designed for this purpose, we present the results obtained by running our SimCat system on multifarm tracks of OAEI 2016 evaluation campaign following website `http://oaei.ontologymatching.org/2016/results/multifarm/index.html`.

The multifarm track is constituted of seven ontologies. These ontologies describe the conference domain and are based on the ontologies of the OAEI conference track. These ontologies have been translated in nine different languages (since 2015 the Arabic language is included, Chinese, Czech, Dutch, French, German, Portuguese, Russian, and Spanish) and the corresponding alignments between these ontologies. The purpose of multifarm is to evaluate and compare the performance of matching approaches with a special focus on multilingualism.

[2] `http://wordnet.princeton.edu/`

[3] `www.linguatools.de/disco/disco_en.html`

3 General Comments

The evaluation conducted on SimCat system confirmed the following points:

- The results obtained from the Yandex translator API are quite promising.
- The similarity based on the categories of the words could provide good results.
- In overall, the SimCat system provides promising results by achieving a good F-Measure, however, it consumes 24 min as computation time for each task. This is considered as a drawback of the proposed system, since the multifarm contains 55 tasks.

4 Conclusion

In this paper, We have presented SimCat, an automatic matching system developed specifically for aligning multilingual ontologies. The SimCat system implements a matcher based on the categories of the words and a translation based on Yandex engine to find the semantic correspondences between different concepts of the two ontologies described in different natural languages. Regarding the first participation of SimCat system in OAEI2016, the results are acceptable, however there is much work to do in order to improve our system.

References

1. M. Ehrig, "Ontology Alignment: Bridging the Semantic Gap", Springer, 2007.
2. P. Shvaiko and J. Euzenat, "Ontology Matching: State of the Art and Future Challenges", IEEE Transactions on Knowledge and Data Engineering vol. 25 no. 1, pp. 158-176, 2013.
3. J. Euzenat and P. Shvaiko, "Ontology Matching", Springer-Verlag, Heidelberg, 2013.
4. P. Bouquet, J. Euzenat, E. Franconi, L. Serafini, G. Stamou and S. Tessaris "Specification of a Common Framework for Characterizing Alignment", Deliverable 2.2.1, Knowledge Web NoE, Technical Report, Italy, 2004.
5. M. Cheatham, Z. Dragisic, J. Euzenat, D. Faria, A. Ferrara, G. Flouris, I. Fundulaki, R. Granada, V. Ivanova, E. Jiménez-Ruiz, P. Lambrix, S. Montanelli, C. Pesquita, T. Saveta, P. Shvaiko, A. Solimando, C. Trojahn and O. Zamazal, "Results of the Ontology Alignment Evaluation Initiative 2015", 10th Workshop on Ontology Matching, 2015.
6. Z. Dragisic, K. Eckert, J. Euzenat, D. Faria, A. Ferrara, R. Granada, V. Ivanova, E. Jiménez-Ruiz, A. O. Kempf, P. Lambrix, S. Montanelli, H. Paulheim, D. Ritze, P. Shvaiko, A. Solimando, C. Trojahn-dos-Santos, O. Zamazal and B. Cuenca Grau, "Results of the Ontology Alignment Evaluation Initiative 2014", 9th Workshop on Ontology Matching, 2014.
7. B. Cuenca Grau, Z. Dragisic, K. Eckert, J. Euzenat, A. Ferrara, R. Granada, V. Ivanova, E. Jiménez-Ruiz, A. Oskar Kempf, P. Lambrix, A. Nikolov, H. Paulheim, D. Ritze, F. Scharffe, P. Shvaiko, C. Trojahn dos Santos, O. Zamazal, "Results of the Ontology Alignment Evaluation Initiative 2013". 8th Workshop on Ontology Matching, 2013.
8. J. Aguirre, K. Eckert, J. Euzenat, A. Ferrara, W.R.v. Hage, L. Hollink, Ch. Meilicke, A. Nikolov, D. Ritze, F. Scharffe, P. Shvaiko, O. Svb-Zamazal, C. Trojahn, E. Jiménez-Ruiz, B. Cuenca-Grau and B. Zapilko:, "Results of the Ontology Alignment Evaluation Initiative 2012", 7th Workshop on Ontology Matching, 2012.

9. O. Svab, V. Svatek, P. Berka, D. Rak and P. Tomasek, "OntoFarm: Towards an Experimental Collection of Parallel Ontologies", In: Poster Track of ISWC 2005, Galway, 2005.
10. C. Meilicke, R. Garca-Castro, F. Freitas, WR. Van Hage, E. Montiel-Ponsoda, R.R. De Azevedo, H. Stuckenschmidt, O. vb-Zamazal, V. Svtek and A. Tamilin, "MultiFarm: A benchmark for multilingual ontology matching". Web Semant. Sci. Serv. Agents World Wide Web. Vol. 15, pp. 6268, 2012.
11. A. Khiat and M. Benaissa, A New Instance-Based Approach for Ontology Alignment. International Journal on Semantic Web and Information Systems (IJSWIS), Vol. 11, No. 3, ISSN 1683-3198, 2015.
12. A. Khiat and M. Benaissa, Boosting Reasoning-Based Approach by Structural Metrics for Ontology Alignment. The Journal of Information Processing Systems (JIPS), 2015.
13. A. Khiat and M. Benaissa and Ernesto Jimnez-Ruiz ADOM: arabic dataset for evaluating arabic and cross-lingual ontology alignment systems. In Proceedings of the 10th International Workshop on Ontology Matching co-located with the 14th International Semantic Web Conference (ISWC 2015), USA, 2015.
14. A. Khiat, G. Diallo, B. Yaman, E. Jimnez-Ruiz and M. Benaissa, ABOM and ADOM: Arabic Datasets for the Ontology Alignment Evaluation Campaign. In Proceedings of the 14th International Conference (ODBASE 2015), Greece, 2015.
15. A. Khiat, CroLOM: Cross-Lingual Ontology Matching System Results for OAEI 2016. In Proceedings of the 12th International Workshop on Ontology Matching co-located with the 15th International Semantic Web Conference (ISWC 2016), Japan, 2016.

XMap : Results for OAEI 2016

Warith Eddine DJEDDI[a,b], Mohamed Tarek KHADIR[a] and Sadok BEN YAHIA[b]

[a]LabGED, Computer Science Department, University Badji Mokhtar, Annaba, Algeria
[b]Faculty of Sciences of Tunis, University of Tunis El-Manar, LIPAH-LR 11ES14, 2092, Tunisia
{djeddi,khadir}@labged.net
sadok.benyahia@fst.rnu.tn

Abstract. We describe in this paper the XMap system and the results achieved during the 2016 edition of the Ontology Alignment Evaluation Initiative. XMap is an automated ontology matching system based on parallel composition of basic ontology matchers and on the use of external resources as background knowledge.

1 Presentation of the system

XMap, as for eXtended Mapping, is one of the leading ontology matching systems for large-scale ontology matching relying on the notion of context in order to deal with lexical ambiguity as well as a divide-and-conquer approach to tackle the issue of matching large ontologies.

A semantic similarity measure has been defined using UMLS [1] and WordNet [3] to provide a synonymy degree between two entities from different ontologies, by exploring both of their lexical and structural contexts. The translation into many languages is based on the Microsoft ®Translator. Our system stores locally all translation results from Microsoft ®Translator in dictionary files. The translator will also be queried only when no stored translation are found in order to gain time and avoid overloading the server.

In this version, the system architecture remained unchanged but the system implementation was modified as well as the implementation of several basic matchers in order to prepare the system for the following test sets: "'Interactive matching evaluation"' and "'Disease and Phenotype"' tracks.

2 State, purpose, general statement

As stated before, the architecture of the new version of the system remained unchanged according to the version from 2015 [2]. We only added an interactive matcher [4] in XMap using an oracle by modifying the validation process of the candidate mappings according to the quality of the interactive matching in terms of F-measure and number of required interactions. This process is performed after each round of candidate retrieving.

To recapitulate, our approach is based on semantic techniques and on a parallel execution strategy, to address the challenge of scalability and efficiency of matching

techniques. One of the main trusts of the introduced approach is the increasing scalability and speed of ontology alignment by matching linguistic and structural features. It is a multi-layer system which uses three different layers to perform the ontology alignment process: a terminological layer, a structural layer and an alignment layer. The output values of each layer serves as input to the upper one and each layer provides an improvement in the computation of the similarity between concepts.

3 Results

In this section, we present the evaluation results obtained by running XMap under the SEALS client with *Benchmark*, *Anatomy*, *Conference*, *Multifarm*, *Interactive matching evaluation*, *Large Biomedical Ontologies* and *Disease and Phenotype* tracks.

Benchmark XMap performs very well on the *biblio* and *film* data set. Table 1 summarises the average results obtained by XMap.

Table 1. Results for Benchmark track.

Test	Precision	Recall	F-Measure
biblio	0.95	0.40	0.56
film	0.78	0.49	0.60

Anatomy The Anatomy track consists of finding an alignment between the Adult Mouse Anatomy (2744 classes) and a part of the NCI Thesaurus (3304 classes) describing the human anatomy. XMap achieves a good F-Measure value of ≈89% in a reasonable amount of time (45 sec.) (see Table 2). In terms of F-Measure/runtime, XMap is ranked 3nd among the tools participated in this track.

Table 2. Results for Anatomy track.

System	Precision	F-Measure	Recall	Time(s)
XMap	0.929	0.896	0.865	45

Conference The Conference track uses a collection of 16 ontologies from the domain of academic conferences. Most ontologies were equipped with OWL DL axioms of various types; this opens a useful way to test our semantic matchers. The match quality was evaluated against the original (ra1) as well as entailed reference alignment (ra2) and violation free version of reference alignment (ra2). As Table 3 shows, for the three evaluations, we achieved a good F-Measure values.

For each reference alignment, three evaluation modalities are applied : a) M1 only contains classes, b) M2 only contains properties, c) M3 contains classes and properties.

XMap achieved the highest improvement between the 2016 and 2014 evaluation.

Table 3. Results for Conference track.

	Precision	F-Measure 1	Recall
Original reference alignment (ra1)			
ra1-M1	0.86	0.73	0.63
ra1-M2	0.75	0.32	0.2
ra1-M3	0.85	0.68	0.57
Entailed reference alignment (ra2)			
ra2-M1	0.81	0.68	0.58
ra2-M2	0.83	0.35	0.22
ra2-M3	0.81	0.63	0.52
Violation reference alignment (rar2)			
rar2-M1	0.8	0.69	0.6
rar2-M2	0.83	0.35	0.22
rar2-M3	0.8	0.65	0.55

Multifarm This track is based on the translation of the OntoFarm collection of ontologies into 9 different languages. XMap have low performance due to many internal exceptions. The results are showed in Table 4.

Table 4. Results for Multifarm track.

System	Different ontologies			Same ontologies		
	P	F	R	P	F	R
XMap	0.30	0.007	0.003	0.00	0.00	0.00

Interactive matching evaluation For the 2016 edition, participating systems are evaluated on the Conference and Anatomy data set using an oracle based on the reference alignment.

In this evaluation, we look at how interacting with the user improves the matching results, which methods are most promising and how many interactions are necessary.

XMap uses various similarity measures to generate candidate mappings. It applies two thresholds to filter the candidate mappings - one for the mappings that are directly added to the final alignment and another for those that are presented to the user for validation. The latter threshold is selected to be high in order to minimize the number of requests and the rejected candidate mappings from the oracle; the requests are mainly about incorrect mappings. The mappings accepted by the user are moved to the final

alignment. On the opposite side is XMap - it benefits the least from the interaction with the oracle. All XMap's measures differ with less than 0.2% from the non-interactive runs, and performance does not change at all with the increasing error rates.

Large biomedical ontologies This track consists of finding alignments between the Foundational Model of Anatomy (FMA), SNOMED CT, and the National Cancer Institute Thesaurus (NCI). The results obtained by XMAP are depicted by Table 5.

Table 5. Results for the Large BioMed track.

Test set	Precision	Recall	F-Measure	Time(s)
Small FMA-NCI	0.977	0.901	0.937	17
Whole FMA-NCI	0.902	0.847	0.874	116
Small FMA-SNOMED	0.989	0.846	0.912	54
Whole FMA- Large SNOMED	0.965	0.843	0.900	366
Small SNOMED-NCI	0.911	0.564	0.697	267

In general, we can conclude that XMap achieved a good precision/recall values. The high recall value can be explained by the fact that UMLS thesaurus contains definitions of highly technical medical terms.

Disease and Phenotype 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).

XMap achieved fair results according to the three evaluation (Silver standard, Manually generated mappings and Manual assessment of unique mappings).

4 General comments

4.1 Comments on the results

This is the 4th time that we participate in the OAEI campaign. The official results of OAEI 2016 show that XMap is competitive with other well-known ontology matching systems in all OAEI tracks. The current version of XMap has shown a significant improvement (both in terms of matching quality and runtime) in comparison to the version from 2015 [2].

4.2 Comments on the OAEI 2016 procedure

As a fourth participation, we found the OAEI procedure very convenient and the organizers very supportive. The OAEI test cases are various, and this leads to a comparison on different levels of difficulty, which is very interesting. We found that SEALS platform is a precious tool to compare the performance of our system with the others.

5 Conclusion

In this paper, we presented the results achieved during the 2016 edition of the OAEI campaign. The system managed to improve its performance significantly compared to the previous year, which is reflected in the performance on several tracks. XMap participated for the first year to the interactive track. The results are promising especially on large-scale tasks which is a critical challenge in ontology matching.

References

1. Bodenreider, O. : The Unified Medical Language System (UMLS): integrating biomedical terminology, Nucleic Acids Research, vol. 32, D267-D270 (2004).
2. Djeddi, W., Khadir, M. T. & Ben Yahia, S.: XMap: results for OAEI 2015. In Proceedings of the 10th International Workshop on Ontology Matching collocated with the 14th International Semantic Web Conference (ISWC 2015), October 12, pp. 216–221. Bethlehem, PA, USA (2015).
3. Fellbaum, C. : WordNet: An Electronic Lexical Database, MIT Press, Cambridge, MA (1998).
4. Paulheim, H., Hertling, S. & Ritze, D.: Towards evaluating interactive ontology matching tools. In Proc. 10th Extended Semantic Web Conference (ESWC), Montpellier (FR), pp. 31–45, (2015).

Introducing the Disease and Phenotype OAEI Track*

Ian Harrow[1], Ernesto Jimenez-Ruiz[2], Andrea Splendiani[1],
Martin Romacker[1], Stefan Negru[1], Peter Woollard[1], Scott Markel[1],
Yasmin Alam-Faruque[1], Martin Koch[1], Erfan Younesi[1] and James Malone[1]

[1] Pistoia Alliance Ontologies Mapping Project, Pistoia Alliance Inc. USA
[2] Department of Computer Science, University of Oxford, UK

1 Introduction

The Pistoia Alliance Ontologies Mapping project[1] was set up to find or create better tools and services for mapping between ontologies (including controlled vocabularies) in the same domain and to establish best practices for ontology management in the Life Sciences. The project has developed a formal process to define and submit a request for information (RFI) from existing ontologies mapping tool providers to enable their evaluation.[2] A critical component of any Ontologies Mapping tool is the embedded ontology matching algorithm, therefore the project is supporting their development and evaluation through sponsorship and organisation of the new *Disease and Phenotype* track for the OAEI campaign[3] [1] which is described in this paper.

2 Datasets

The *Disease and Phenotype* track[4] comprises two tasks that will involve the pairwise alignment of the HPO, MP, DOID and ORDO ontologies (Table 1 shows the metrics of these ontologies):

- **Task 1:** matching of the Human Phenotype Ontology (HPO) to the Mammalian Phenotype Ontology (MP).
- **Task 2:** matching of the Human Disease Ontology (DOID) to the Orphanet and Rare Diseases Ontology (ORDO).

The first task is important for translational science where HPO includes inherited diseases and MP originated from rodents as a model mammalian organism for many laboratory studies, including gene knock out. The second task includes representation of rare human diseases in both ontologies which are of fundamental importance for understanding how genetic variation can cause disease. Currently, such mappings are mostly curated by bioinformatics and disease experts who would benefit from automation supported by implementation of ontology matching algorithms into their workflows.

We have extracted a "baseline" reference alignments for the track based on the available BioPortal mappings [2] which are considered as a baseline since they are incomplete and may contain errors.

* We have also submitted a 4-pages paper about the Pistoia Alliance Ontologies Mapping Project to the ISWC 2016 posters and demos track.

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

[2] https://pistoiaalliance.atlassian.net/wiki/display/PUB/Ontologies+Mapping+Resources

[3] http://oaei.ontologymatching.org/2016/

[4] http://oaei.ontologymatching.org/2016/phenotype/description.html

Table 1. Metrics of the track ontologies. Source: NCBI BioPortal on 19th Aug 2016

Ontology	Number of classes	Maximum depth	Avg. number of children
HPO	15,319	15	3
MP	11,720	Undisclosed	Undisclosed
DOID	10,905	12	3
ORDO	13,105	11	16

3 Evaluation process

The evaluation of the Disease and Phenotype Track will be run with support of the SEALS infrastructure.[5] Systems will be evaluated and ranked according to the following criteria:

- Precision and Recall with respect to a voted reference alignment that will be built automatically to generate consensus voting for the outputs of the participating systems.
- Recall with respect to manually generated mappings for three areas (carbohydrate, obesity and breast cancer).
- Manual assessment of a subset of the generated mappings, specially the ones that are not suggested by other systems.
- Performance in other tracks will also be taken into account, especially the OAEI interactive track [3] where the *Disease and Phenotype* dataset is also used.[6]

Additionally, systems able to discover complex logic relations in mappings beyond equivalence and subsumption will also be considered. The evaluation of these mappings will be in parallel to the evaluation of standard equivalence and subsumption mappings. Complex mappings should be provided in OWL 2 format.

Acknowledgements

This work was partially funded by the Pistoia Alliance Ontology Mappings project, the EU project Optique (FP7-ICT-318338), and the EPSRC projects ED3 and DBOnto.

References

1. Cheatham, M., Dragisic, Z., Euzenat, J., Faria, D., Ferrara, A., Flouris, G., Fundulaki, I., Granada, R., Ivanova, V., Jiménez-Ruiz, E., Lambrix, P., Montanelli, S., Pesquita, C., Saveta, T., Shvaiko, P., Solimando, A., dos Santos, C.T., Zamazal, O.: Results of the ontology alignment evaluation initiative 2015. In: Proceedings of the 10th International Workshop on Ontology Matching. (2015) 60–115
2. Fridman Noy, N., Shah, N.H., Whetzel, P.L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D.L., Storey, M.A.D., Chute, C.G., Musen, M.A.: BioPortal: ontologies and integrated data resources at the click of a mouse. Nucleic Acids Research **37**(Web-Server-Issue) (2009)
3. Dragisic, Z., Ivanova, V., Lambrix, P., Faria, D., Jimenez-Ruiz, E., Pesquita, C.: User validation in ontology alignment. In: International Semantic Web Conference. (2016)

[5] `http://oaei.ontologymatching.org/2016/seals-eval.html`
[6] `http://oaei.ontologymatching.org/2016/interactive/index.html`

Annotating Web Tables Through Ontology Matching

Vasilis Efthymiou[1,2], Oktie Hassanzadeh[1], Mohammad Sadoghi[1], and Mariano Rodriguez-Muro[1]

[1] IBM T.J. Watson Research Center [2] University of Crete, Greece
{vefthymi,hassanzadeh,msadoghi,mrodrig}@us.ibm.com

1 Introduction

Web tables have been proven to constitute valuable sources of information for applications, ranging from Web search, to data discovery in spreadsheet software and KB augmentation [1]. A requirement for those applications is to understand the semantics of Web tables and potentially match their contents with existing URIs in the Web of Data, a process known as Web table annotation [4].

Recent works on Web table annotation follow an iterative approach between instance- and schema-level refinements, until convergence [6, 7]. In this work, we annotate Web tables using ontology matching. As this field has solid tools and benchmarks[3], we design a framework that provides the required input to any ontology matching tool, resulting in Web table annotations. Moreover, our blocking enables even the less scalable ontology matching tools provide annotations to large-scale KBs, such as DBpedia. The contributions of our work are:

- We introduce a generic and scalable framework for Web table annotation using existing ontology alignment systems.
- We evaluate our framework and compare the results against state-of-the-art Web table annotation tools, with promising results.
- Our framework can be extended as a benchmark for ontology matching tools.

2 Matching Framework

Model. We assume that each table row describes a real-world entity, and each column represents a property. Each cell of the *header row* defines the name of a property, except the cell of the *label column*, which defines the name of the table's class. All the entities in the table are instances of this class. The values of a column can be either literals, or references to other entities, corresponding to datatype, or object properties, respectively. To make this distinction, we sample the data types of each column, also identifying the label column, as in [6]. In a second scan, we create a new instance of the table class for each row, whose property values are the cell contents of this row for the respective column.

Blocking. To enable ontology matching tools that do not scale well be applicable in this framework, and to improve the efficiency of matching tools that do scale, we have applied a pre-processing step of candidate mappings selection, known as blocking [2]. Specifically, we retain from DBpedia, the target ontology, only those instances whose labels match with the labels of our table's instances.

[3] http://oaei.ontologymatching.org/

Finally, we call an ontology matching tool with the table ontology and the DBpedia ontology after blocking, as input, and return the mapping results.

Evaluation. We evaluate our approach using the instance mappings of the T2D gold standard[4] and LogMap [5], one of the most efficient ontology matching tools [3]. Our MapReduce-based framework annotates and evaluates the whole corpus in less than 4 minutes. Table 1 presents the micro-averaged recall, precision, and F-measure results, against T2K [6] and two baselines: **DBpedia lookup.** For each entity label in our table, we use top-1 DBpedia lookup[5] result as annotation. **DBpedia lookup refined.** We keep the type of the top-1 lookup result for each cell in a first scan of the table, and then the top-5 most frequent types for each column as acceptable types. Then, we perform a second lookup, restricting the results to the acceptable types, and use the top-1 result as the annotation.

Table 1. Results over T2D gold standard. Blocking results in parentheses.

Method	**Recall**	**Precision**	**F-measure**
DBpedia lookup	0.73	0.79	0.76
DBpedia lookup refined	0.76	0.86	0.81
T2K	0.76	0.90	0.82
Ontology matching	0.57 (0.71)	0.89 (0.32)	0.70 (0.44)

The results show that our framework, using LogMap, suggests a good number of correct results, with high precision. In the future, we plan to improve blocking and extend our model to provide a first alignment, which can be utilized by many ontology matching tools. Our goal is to provide an ontology matching benchmark for instance-, class- and property-mappings, that can result in a new track in the upcoming OAEI campaigns.

References

1. S. Balakrishnan, A. Y. Halevy, B. Harb, H. Lee, J. Madhavan, A. Rostamizadeh, W. Shen, K. Wilder, F. Wu, and C. Yu. Applying WebTables in Practice. In *CIDR*, 2015.
2. V. Christophides, V. Efthymiou, and K. Stefanidis. *Entity Resolution in the Web of Data*. Synthesis Lectures on the Semantic Web: Theory and Technology. Morgan & Claypool Publishers, 2015.
3. E. Daskalaki, G. Flouris, I. Fundulaki, and T. Saveta. Instance matching benchmarks in the era of linked data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 39:1–14, 2016.
4. O. Hassanzadeh, M. J. Ward, M. Rodriguez-Muro, and K. Srinivas. Understanding a large corpus of web tables through matching with knowledge bases: an empirical study. In *ISWC*, pages 25–34, 2015.
5. E. Jiménez-Ruiz and B. Cuenca Grau. Logmap: Logic-based and scalable ontology matching. In *ISWC*, pages 273–288, 2011.
6. D. Ritze, O. Lehmberg, and C. Bizer. Matching HTML tables to dbpedia. In *WIMS*, pages 10:1–10:6, 2015.
7. Z. Zhang. Towards efficient and effective semantic table interpretation. In *ISWC*, pages 487–502, 2014.

[4] http://webdatacommons.org/webtables/goldstandard.html
[5] http://wiki.dbpedia.org/projects/dbpedia-lookup

Ontology Matching Evaluation: A Statistical Perspective

Majid Mohammadi[1], Wout Hofman[2], Yao-hua Tan[1]

[1] Faculty of Technology, Policy and Management, Delft University of Technology, Netherlands
[2] Department of Technical Science, The Netherlands Institute of Applied Technology (TNO), Soesterberg, the Netherlands

Abstract. This paper proposes statistical approaches to test if the difference between two ontology matchers is real. Specifically, the performances of the matchers over multiple data sets are obtained and based on their performances, the conclusion can be drawn whether one method is better than one another or not. To do so, the paired t-test and Wilcoxon signed rank test are proposed and the comparisons over six recently proposed methods are reported.

Keywords: Ontology alignment, evaluation, statistical inference, paired t-test, Wilcoxon signed rank test

1 Introduction

There has been an increasing interest in ontology matching (or alignment) over the last years. As data come from various sources these days, the heterogeneity among data is inevitable. The solution to such an issue is ontology matching, which has a wide range of application from data integration and agent interoperability in computer science to matching ontologies in biomedical and geoscience. As a result, a plethora of methods have been proposed claiming that their method is better than, or competitive with, other state-ofthe-art algorithms. However, no evidence has been brought to support such a claim

2 Binary comparison of matchers

The hypothesis testing is one of the major topic in the realm of statistical inference. Here, we aim at utilizing this technique to indicate if the average difference in the performance scores of two matchers over multiple benchmarks is meaningful or not. To leverage the hypothesis testing, a null hypothesis is required. The null hypothesis (shown by H_0) states that there is no significant difference between two populations according to the available samples of the populations. The alternative hypothesis (shown by H_a), on the other hand, is the rival hypothesis and states that there is meaningful difference between two populations based on available samples. Thus, it is desirable to reject null hypothesis and accept the alternative hypothesis. In ontology matching case, the performance of various matchers over a range of data sets are available and we would like to test if the average of their performances is random. In other words, the null hypothesis and the alternative hypothesis in this case is

$$H_0 : \hat{P}^1 = \hat{P}^2$$

$$H_1 : \hat{P}^1 \neq \hat{P}^2 \tag{1}$$

where $\hat{P}^i$ is the average performances of the matcher i.
Before running any statistical test, the significant level must be determined. the α is the probability of rejecting null hypothesis when the null hypothesis is true. To the best of our knowledge, no statistical techniques have been employed to test the above-mentioned hypothesis. Firstly, the widely-used paired t-test is presented with more detail. Having hard preconditions to be satisfied, it must be warned that t-test might be inappropriate and statistically unsafe. Thus, the Wilcoxon signed-rank test is presented which is able to detect more difference even though the number of samples are not large enough.

2.1 Paired t-test

A common way to check if the difference between two matchers on different data sets is not random is to compute the paired t-test. Let $d_i = P_i^1 - P_i^2$ be the difference between the performances of two matchers over $i-th$ data set. The t statistics is computed as $t = \frac{x-\hat{x}}{\hat{\sigma}_d}$ where $\hat{x}$ and $\hat{\sigma}_d$ are sample average and standard deviation of samples, respectively. This statistics is distributed according to the Student distribution with N 1 degree of freedom. After obtaining the probability of observing the data given that H_0 being true (p-value) according to the Student distribution, the H_0 can be rejected if $p - value < \alpha$ and then Ha is accepted.

2.2 Wilcoxon Signed Rank test

The non-parametric alternative to the paired t-test is Wilcoxon singed rank test. This method ranks the absolute values of performance differences of two matchers. Then, it compares the rank of positive and negative differences. After computing the difference between two matchers over the the $i-th$ data set, d_i , the differences are ranked based on the values of d_i , disregarding its sign. if $d_i = 0$ it is ignored and the average ranks are assigned if the performances over one data set ties. Assume $W^+ = \sum_{d_i>0} rank(d_i)$ and $W^- = \sum_{d_i<0} rank(d_i)$ and $T = min(W^+, W^-)$. Then $z = \frac{T-\frac{1}{4}N(N+1)}{\sqrt{\frac{1}{24}N(N+1)(2N+1)}}$ is distributed according to the normal distribution.

3 Experimental Results

Table 1 tabulates the p-values obtained by paired t-test and Wilcoxon Signed Rank test over six recently proposed methods.

Table 1. The p-values obtained by paired t-test (above diagonal) and Wilcoxon Signed Rank test (below diagonal) over six recently proposed methods: XMAP, AML, AML2014, CroMatcher, edna and refalign.

	XMAP	AML	AML2014	CroMatcher	edna	refalign
XMAP		0.526403	0.23326767	0.00094182	0.000972	0.000939
AML	0.640625		0.05359674	0.00079181	0.113909	0.000697
AML2014	0.00647436	0.01596065		0.00026227	0.243871	0.000243
CroMatcher	0.00097656	6.10E-05	8.56E-05		2.83E-06	0.01664
edna	0.000822	0.011231	0.058088	0.000287		4.75E-06
refalign	0.000977	6.10E-05	8.50E-05	0.003906	0.000285	

Instance Matching Benchmark for Spatial Data: A Challenge Proposal to OAEI

Irini Fundulaki[1] and Axel-Cyrille Ngonga-Ngomo[2]

[1] Institute of Computer Science - FORTH, Greece
[2] Institute for Applied Informatics, University of Leipzig, Germany

Keywords: Instance Matching, Benchmarks, Spatial Datasets, Linked Data

1 Introduction

The number of datasets published in the Web of Data as part of the Linked Data Cloud is constantly increasing. The Linked Data paradigm is based on the unconstrained publication of information by different publishers, and the interlinking of Web resources across knowledge bases. In most cases, the cross-dataset links are not explicit in the dataset and must be automatically determined using Instance Matching (IM) tools (also known as record linkage [1], duplicate detection [2] and, entity resolution [3]) amongst others. The large variety of techniques requires their comparative evaluation to determine which one is best suited for a given context. Performing such an assessment generally requires well-defined and widely accepted benchmarks to determine the weak and strong points of the proposed techniques and/or tools.

A number of real and synthetic benchmarks that address different data linking challenges have been proposed for evaluating the performance of such systems. Those include, but are not limited to, IIMB 2012 [4], Sandbox 2012 [4], RDFT 2013 [5], ID-REC 2014 [6], ONTOBI 2010 [7], Author - Task 2015 [8] and Lance 2015 [9] to mention few. A more complete survey can be found in [10].

2 A benchmark for linking geo-spatial entities

So far, only a limited number of link discovery benchmarks target the problem of linking geo-spatial entities e.g., PABench [11]. However, some of the largest knowledge bases on the Linked Open Data Web are geo-spatial knowledge bases (e.g., LinkedGeoData with more than 30 billion triples). Linking spatial resources requires techniques that differ from the classical mostly string-based approaches. In particular, considering the topology of the spatial resources and the topological relations between them is of central importance to systems driven by spatial data.

We believe that due to the large amount of available geo-spatial datasets employed in Linked Data and in several domains, it is critical that benchmarks for geo-spatial link discovery are developed. For OAEI 2017, we propose the introduction of a new challenge for such systems. The benchmark that the systems will use will be based upon the Lance [9] scalable, schema-agnostic benchmark generator extended with appropriate transformations to tackle geo-spatial link

discovery tasks. Lance is able to produce, from an initial ontology, datasets of arbitrary size and complexity. The configuration of the transformations will be derived from real data. More specifically, we will use the widely accepted Linked Data datasets such as GeoNames, LinkedGeoData, and DBpedia for this task. The tasks proposed will focus on the different types of spatial object representations and will be provided with different severity levels for the applied transformations. In these transformations, objects may keep their representation, they may change their geometry, type or attributes, merge with other objects, or can completely disappear. This is a scenario that stems from the heterogeneous datasets (in structure and semantics) used to describe geo-spatial entities. The produced tasks will be used by IM tools that implement string-based as well as topological approaches for identifying matching entities. The IM frameworks will be evaluated for both accuracy (precision, recall and f-measure) and scalability. Furthermore, the results will be made available in both human and machine-readable form for further processing. Since Lance is schema-agnostic, contrary to PABench, it will be used to produce benchmarks for different (source) ontologies to accommodate the different requirements that stem from a variety of applications.

Acknowledgments

This work was supported by a grant from the EU H2020 Framework Programme provided for the project HOBBIT (GA no. 688227).

References

1. C. Li, L. Jin, and S. Mehrotra. Supporting efficient record linkage for large data sets using mapping techniques. In *WWW*, 2006.
2. A. K. Elmagarmid, P.G. Ipeirotis, and V.S. Verykios. Duplicate Record Detection: A Survey. *TKDE*, 19(1), 2007.
3. I. Bhattacharya and L. Getoor. *Entity resolution in graphs. Mining Graph Data.* Wiley and Sons, 2006.
4. J. Aguirre, K. Eckert, J. Euzenat, et al. Results of the ontology alignment evaluation initiative 2012. In *OM*, 2012.
5. B. Cuenca Grau, Z. Dragisic, K. Eckert, et al. Results of the ontology alignment evaluation initiative 2013. In *OM*, 2013.
6. Z. Dragisic, K. Eckert, J. Euzenat, et al. Results of the ontology alignment evaluation initiative 2014. In *OM*, 2014.
7. K. Zaiss, S. Conrad, and S. Vater. A Benchmark for Testing Instance-Based Ontology Matching Methods. In *KMIS*, 2010.
8. M. Cheatham, Z. Dragisic, J. Euzenat, et al. Results of the ontology alignment evaluation initiative 2015. In *OM*, 2015.
9. T. Saveta, E. Daskalaki, G. Flouris, I. Fundulaki, and A. Ngonga-Ngomo. LANCE: Piercing to the Heart of Instance Matching Tools. In *ISWC*, 2015.
10. E. Daskalaki, G. Flouris, I. Fundulaki, and T. Saveta. Instance matching benchmarks in the era of linked data. *Web Semantics: Science, Services and Agents on the World Wide Web*, 2016.
11. B. Berjawi and F. Duchateau and F. Favetta and M. Miquel and R. Laurini. Pabench: Designing a taxonomy and implementing a benchmark for spatial entity matching. In *GeoProcessing*, 2015.

LION'S DEN
Feeding the LINKLION

Mohamed Ahmed Sherif, Mofeed M. Hassan, Tommaso Soru, Axel-Cyrille Ngonga Ngomo, and Jens Lehmann

Department of Computer Science, University of Leipzig, 04109 Leipzig, Germany
{sherif,mounir,tsoru,ngonga,lehmann}@informatik.uni-leipzig.de

Introduction Over the last years, several tools have been developed with the aim of efficiently supporting the link discovery process [5,7]. This process consisting of two steps: (1) Discovering a Link Specifications (LS) for retrieving high-quality links (i.e. achieve high precision and recall). (2) Carry out the LS to compute the actual links. Several frameworks such as LIMES [3] and SILK [1] have been developed to create such links between the different knowledge bases (KB). While the importance of links between datasets is unequivocal, only few efforts have aimed at making LS available. Such a link repository would however enable a large number of applications, including transfer learning for LS, the provision of provenance and justification information for links, fuzzy inferences on Linked data sets and many more. The importance of links is further underlined by the community efforts have already led to the creation of link repositories such as LINKLION and `sameAs.org`. In view of the dispersed availability of LS in different formats (scripts, XML, RDF), we created LION'S DEN as a companion project to LINKLION. LINKLION is a store for the publication, retrieval and use of links between KB. The portal provides functionality for the upload and the storage of discovered links, as well as meta-information about these links. With LION'S DEN, we introduce an extension of such meta-information by letting the portal user upload files describing LS. We published the LION'S DEN dataset on the LINKLION link discovery portal so as to make them accessible and queryable via a SPARQL endpoint.[1].

The* LION'S DEN *Dataset The dataset is now hosted within the LINKLION project at `http://linklion.org`. Currently, LION'S DEN contains 436 LS that are described by 15 457 triples including the ontology. Metadata on the LION'S DEN dataset is available on *DataHub*.[2]

Ontology To represent the LS in RDF and OWL, we developed the *Lion's Den* vocabulary dubbed `LDEN`[3]. `LDEN` was specified with the aim of supporting any type of LS regardless of the way it was created. in its current version, `LDEN` contains a set of ten classes. Each LS is an instance of the `LinkSpecs` class. The `LinkSpecs` class provides properties that allow referencing the five basic components of any LS which are the *source* and *target* datasets, the *metric* used for linking as well as the *acceptance*

[1] for more details see the extended paper in the project web site `https://svn.aksw.org/papers/2016/ISWC_OM_LionDen/public.pdf`
[2] `http://datahub.io/dataset/lionsden`
[3] `http://www.linklion.org/lden/`

and *reviewing* criteria. In addition, the `LinkSpecs` class provides metadata such as the source LS's URL and creator, publisher, license and provenance information. Currently, our ontology contains three classes derived from the `LinkSpecs` class (`LimesSpecs`, `SilkSpecs` and `ScriptSpecs`), where each of the three classed contains special attributes related to the framework it represents.

Data Sources LION's DEN original LS were collected from four different sources: (1) *The LATC project* provides the interlinking *24/7 Platform*[4]. (2) *LinkedGeoData*[5] is a project to convert spatial information provided by *OpenStreetMap* to the Web of Data. (3) *DBpedia-links*[6] is a repository that contains links, LS and link extraction scripts. (4) The LIMES[7] Link discovery framework supports manual configuration for linking tasks through XML based specification files.

Conversion Process As the original configuration files for both SILK and LIMES were in XML format, we built a specialized XML to RDF converter for each of them. The source code of the dataset converters is available at the project repository[8].

Provenance The LINKLION dataset reuses properties and classes from the PROV W3C recommendation[9] to keep track of data provenance.

Use Cases Having the LS of LION's DEN together with the links of LINKLION in a machine readable format and serving them from one portal offers a lot of opportunities, including, but not limited to: benchmarking link discovery algorithms, automatic linked data enrichment [6], key discovery [8], unification of LS, LS tansfer learning [2] and Link Discovery over *n* Knowledge Bases [4].

References

1. R. Isele, A. Jentzsch, and C. Bizer. Efficient Multidimensional Blocking for Link Discovery without losing Recall. In *WebDB*, 2011.
2. A.-C. N. Ngomo, J. Lehmann, and M. Hassan. Transfer learning of link specifications. In *Seventh IEEE International Conference on Semantic Computing (ICSC)*, 2013.
3. A. N. Ngomo. A time-efficient hybrid approach to link discovery. In *Proceedings of the 6th International Workshop on Ontology Matching, Bonn, Germany, October 24, 2011*, 2011.
4. A.-C. Ngonga Ngomo, M. A. Sherif, and K. Lyko. Unsupervised link discovery through knowledge base repair. In *Extended Semantic Web Conference (ESWC 2014)*, 2014.
5. G. Papadakis, E. Ioannou, C. Niederèe, T. Palpanasz, and W. Nejdl. Eliminating the redundancy in blocking-based entity resolution methods. In *JCDL*, 2011.
6. M. Sherif, A.-C. Ngonga Ngomo, and J. Lehmann. Automating RDF dataset transformation and enrichment. In *12th Extended Semantic Web Conference, Portoroz, Slovenia, 31st May - 4th June 2015*. Springer, 2015.
7. J. Sleeman and T. Finin. Computing foaf co-reference relations with rules and machine learning. In *Proceedings of the Third International Workshop on Social Data on the Web*, 2010.
8. T. Soru, E. Marx, and A.-C. Ngonga Ngomo. ROCKER – a refinement operator for key discovery. In *Proceedings of the 24th International Conference on World Wide Web*, 2015.

[4] `https://www.assembla.com/wiki/show/silk/Link_Specification_Language`
[5] `http://linkedgeodata.org/`
[6] `https://github.com/dbpedia/dbpedia-links/`
[7] `https://github.com/AKSW/LIMES`
[8] `https://github.com/AKSW/LionDen`
[9] `http://www.w3.org/ns/prov#`

Matching Instances in GeoLink

Michelle Cheatham, Reihaneh Amini, and Chandan Patel

DaSe Lab, Wright State University, Dayton OH 45435, USA,
{michelle.cheatham, amini.2, patel.383}@wright.edu

Abstract. We propose the use of the GeoLink data repository as an instance matching benchmark. The GeoLink project brings together seven datasets related to geoscience research. Both the T-box and the A-box of GeoLink are significantly larger than current benchmarks, and they have interesting challenges, such as geospatial and temporal data.

GeoLink is part of the NSF EarthCube initiative. Seven diverse geoscience datasets have been brought together into a single data repository. The ontology is documented at `http://schema.geolink.org`, and the triple store is accessible at `http://data.geolink.org`. There are currently 282 classes, 338 properties, 5,118,150 instances and 45,093,750 triples in the knowledge base. The are also owl:sameAs and skos:closeMatch links between instances of different types. The sameAs links were manually generated by the data providers, while the closeMatch links were generated by an automated coreference resolution system. We highlight three different classes within the GeoLink schema that pose different opportunities for evaluating and challenging coreference resolution systems: Person, Cruise, and Organization.

Person Instances of Person appear in a variety of contexts such as Chief Scientist on a cruise, Principal Investigator on a project, participant in a meeting, or creator of a dataset or paper. Key object properties related to the person class reflect these different contexts. Related data properties include name, email address, and ORCID.[1] GeoLink considers the NSF dataset to be "canonical" for the Person class, meaning that Person instances in each of the other datasets have been mapped to NSF instances. The NSF dataset contains 335,504 people, so it is not feasible to compare each person from one of the constituent datasets to every person in the NSF datset. This benchmark can therefore be used to encourage development of systems that employ effective filtering or other mechanisms to achieve scalablility. The triple store currently contains 15,660 people not in the NSF dataset. There are 790 sameAs and 1,405 closeMatch links between these people and those within the NSF data.

Cruise There are 12,070 cruises in the GeoLink repository, potentially allowing an m by n comparison. There are 1,356 sameAs links and 368 closeMatch links among cruises. The cruise coreference task is intriguing because cruises have geospatial and temporal elements, which are considered an important challenge

[1] `http://orcid.org`

for coreference resolution systems [3]. Two properties of particular interest are hasTrack and hasPortCall properties. A cruise's track is generally a series of latitude and longitude coordinates. The Cruise class also has properties hasStartPortCall, hasMidPortCall, and hasEndPortCall. The PortCall class is in the domain of the properties hasTimeStamp and hasPort, whereas the range of hasTimeStamp is a date time literal and the range of hasPort is Place. A place can be described in terms of its latitude and longitude, but it might also be identified using a gazetteer term.

Organization Compared to Person and Cruise, the GeoLink knowledge base contains relatively little information about instances of the Organization class. There is often little data other than the organization's title and the set of people who are affiliated with it in the knowledge base. Finding coreferences in this situation is likely to be difficult for approaches that rely on extensive schema information; however, approaches that rely on, for instance, the degree of overlap between the people affiliated with two organizations to measure their similarity, may perform quite well. Because there are nearly 300,000 organizations within GeoLink, this is again a task in which approaches that do not perform some type of filtering are unlikely to be feasible. There are currently no sameAs links between organizations, but 268 closeMatch links have been established.

There are several existing coreference resolution benchmarks. The dominant existing benchmark is that of the OAEI, which has included an instance matching track since 2009 [1]. Some tasks within this track are synthetic (generated via SPIMBENCH [2]) while others are real-world. The benchmark proposed here differs because it is less narrowly focused and involves a much larger schema and A-box. On the other hand, because the current set of links in GeoLink is likely not exhaustive, only recall (and not precision) can be evaluated.

Acknowledgments The authors sincerely thank the GeoLink team.[2] This work was supported by the National Science Foundation GeoLink project (1440202).

References

1. Ferrara, A., Nikolov, A., Noessner, J., Scharffe, F.: Evaluation of instance matching tools: The experience of OAEI. Web semantics: Science, services and agents on the World Wide Web 21, 49–60 (2013)
2. Saveta, T., Daskalaki, E., Flouris, G., Fundulaki, I., Herschel, M., Ngonga Ngomo, A.C.: Pushing the limits of instance matching systems: A semantics-aware benchmark for linked data. In: Proceedings of the 24th International Conference on World Wide Web. pp. 105–106. ACM (2015)
3. Shvaiko, P., Euzenat, J.: Ontology matching: State of the art and future challenges. IEEE Transactions on Knowledge and Data Engineering 25(1), 158–176 (2013)

[2] `http://www.geolink.org/team.html`

Toward Better Debugging Support on Extended SPARQL queries with On-the-fly Ontology Mapping Generation

Takuya Adachi[1] and Naoki Fukuta[2]

[1] Faculty of Informatics, Shizuoka University.
[2] Department of Informatics, Shizuoka University.
{cs13007@s.,fukuta@}inf.shizuoka.ac.jp

Abstract. SPARQLoid is an extended syntax of SPAQL to utilize reliability degrees in weighted ontology mappings as well as some controls of priorities to be searched based on the weights associated to the mappings. In this paper, we demonstrate a debugging support system for the use of such an extended SPARQL queries.

1 Introduction

SPARQLoid[3] extends a functionality to SPARQL queries for utilizing ontology mappings on the queries as well as the reliability degrees that are often supplied as "weights"[2] in the mappings.

Here, we consider the case that a user is going to write and execute a SPARQLoid query shown in the above Listing with a certain ontology mapping.

```
 1: PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
 2: PREFIX own: <http://example/ownOntology/>
 3: SELECT DISTINCT ?person ?name
 4: WHERE {
 5: ?person rdf:type own:student .
 6: ?person own:belong own:ShizuokaUniversity .
 7: ?person own:hasName ?name .
 8: THRSHOLD { own:student >= 0.6 , own:hasName >= 0.4 }
 9: CRITERIA ?c { ( ( own:student ) * 60
10: + ( own:hasName ) * 40 ) }
11: RANKING ?score { ?c }
12: } limit 100
```

On SPARQLoid, users are allowed to make use of ontology mappings as well as make some filterings based on the reliability degrees of the associated ontology mapping data.

For example, on the above SPARQLoid query, the THRESHOLD is set and it means that the confidence value for $own : student$ should be equal or higher than 0.6, and the results of $?person$ will be some values associated with the mapped data such as $cs13000$. The same thresholding will be applied for $own : hasName$ and $?name$ will be NakoOkuhama and 70310000, for example. When the user is going to obtain the names of specified students such

as NakoOkuhama for the values of $?person$, then the user should set 0.4 to THRESHOLD for own : $hasName$, set the weight for $hasStudentID$ to 0.2, $hasFullName$ to 0.6, respectively. If we know the entire mappings and data associated with them, we could specify appropriate threshold values. However, such ontology mapping-specific values cannot be easily adjusted in case there are huge amount of mappings in the specified ontology mapping.

2 The Debugging Support System

We are implementing a query coding and debugging support system which is an extension of our previously implemented system[1]. The structure of the system is shown in Figure 1. Our system includes an "on-the-fly" ontology matcher to evaluate the used ontology mappings and suggests which mappings will be used in the query. Furthermore, the on-the-fly matcher can also be used to interactively add one-time mappings that could produce more instances in the results of the specified query, as well as adjusting weights of these mappings.

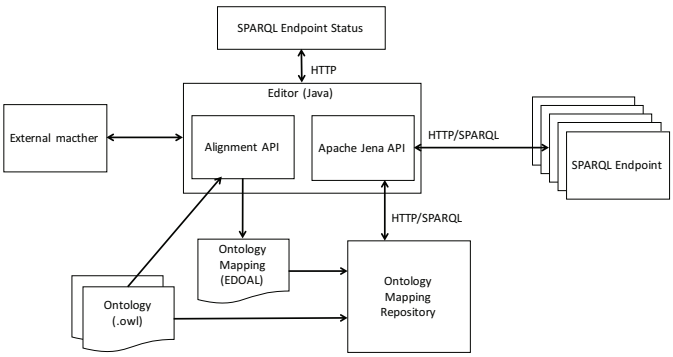


Fig. 1. Abstract System Configuration

References

1. Adachi, T., Yamada, N., Fukuta, N.: Toward better query coding support utilizing ontology mappings. In: Proc. of 1st International Workshop on Platforms and Applications for Social problem Solving and Collective Reasoning (PASSCR2016) (2016), (to appear)
2. Atencia, M., Borgida, A., Euzenat, J., Ghidini, C., Serafini, L.: A formal semantics for weighted ontology mappings. In: Proc. of the 11th International Semantic Web Conference (ISWC2012). pp. 17–33 (2012)
3. Fujino, T., Fukuta, N.: SPARQLoid - a querying system using own ontology and ontology mappings with reliability. In: Proc. of the 11th International Semantic Web Conference (Posters & Demos) (ISWC2012) (2012)

Quality Checking and Matching Linked Dictionary Data

Kun Ji, Shanshan Wang, and Lauri Carlson

University of Helsinki,
Department of Modern Languages
{kun.ji,shanshan.wang,lauri.carlson}@helsinki.fi

1 Introduction

The growth of web accessible dictionary and term data has led to a proliferation of platforms distributing the same lexical resources in different combinations and packagings. Finding the right word or translation is like finding a needle in a haystack. The quantity of the data is undercut by the redundancy and doubtful quality of the resources.

In this paper, we develop ways to assess the quality of multilingual lexical web and linked data resources by internal consistency. Concretely, we deconstruct Princeton WordNet [1] to its component word senses or word labels, with the properties they have or inherit from their synsets, and see to what extent these properties allow reconstructing the synsets they came from. The methods developed should then be applicable to aggregation of term data coming from different term sources - to find which entries coming from different sources could be similarly pooled together, to cut redundancy and improve coverage and reliability. The multilingual dictionary BabelNet [2] can be used for evaluation.

We restrain our current research to dictionary data and improving language models rather than introducing external sources.

2 Methodology

In ([3]) we canvassed our sample of large dictionary and term databases for the descriptive fields/properties of entries to see what sources for matching entries they provide. The following types of properties susceptible to matching could be found in different combinations:

```
1) languages and labels
2) translations / synonyms
3) thematic (subject field) classifications
4) hypernym (genus/superclass) lattice
5) other semantic relations (antonym, meronym, paronym)
6) textual definitions / examples / glosses
7) source indications
8) grammatical properties (part of speech, head word, etc)
9) distributional properties (frequency, register etc)
10) instance data (text or data containing or labeled by terms)
```

Normally, these data sources in a dictionary vary in terms of coverage, unambiguity and information value. Labels are exact, but polysemous; semantic properties are informative, but scarce; distributional properties have a large information potential, but hard to make precise. Subject field classifications are potentially powerful, but alignments between them are often open as well.

Based on the above information sources, we have developed string-based distributional distance measures. Many of them are variations of Levenshtein edit distance [4]. Distributional measures are adaptable by machine learning, language independent and cheap, but fall short with unconstrained natural language (witness MT). A reason to hope that purely distributional methods work better on dictionary data is that dictionaries are a constrained language and self-documenting. For example, the following glosses for "green card" (culled from web-based glossaries) seem at first sight lexically unrelated, but WordNet synsets and the hypernym lattice allow relating many label pairs in them: "permit" is a "document", "immigrant" is a "person", "US" is "United States".

```
GREEN CARD OR PERMANENT RESIDENT CARD: A green card is a document which demonstrates that a
person is a lawful permanent resident allowing a non-citizen to live and work in the United
States indefinitely. A green card/lawful permanent residence can be revoked if a person does
not maintain their permanent residence in the United States travels outside the country for
too long or breaks certain laws.

GREEN CARD:  A permit allowing an immigrant to live and work indefinitely in the US.
```

We have tested and trained measures for some of the properties listed above on WordNet data. Although they find definite correlations, they are too weak taken singly to predict the synsets. The task is to develop a synset distance measure that combines individual measures on the different criteria above to one similarity vector and train it on WordNet data for optimal aggregation of WordNet senses back to WordNet synsets.

3 Conclusion

In this paper, we only use information available in the dictionary entries themselves in the criteria 1-9 above, to see how far dictionary internal information goes to reconstruct synset structure. Use of external information (instance data, language specific parsers, MT) will follow in subsequent papers.

References

1. Princeton University "About WordNet." WordNet. Princeton University. 2010. <http://wordnet.princeton.edu>.
2. R. Navigli and S. Ponzetto. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. Artificial Intelligence, 193, Elsevier, 2012, pp. 217-250.
3. Wang. S and L. Carlson 2016. Linguistic Linked Open Data as a Source for Terminology - Quantity versus Quality. Proceedings of NordTerm 2015 (to appear).
4. Levenshtein, Vladimir I. (February 1966). "Binary codes capable of correcting deletions, insertions, and reversals". Soviet Physics Doklady. 10 (8): 707–710.

Exploiting Ontology Matching to Support Reuse in PURO-started Ontology Development

Marek Dudáš, Ondřej Zamazal, and Vojtěch Svátek

Department of Information and Knowledge Engineering,
University of Economics, W. Churchill Sq.4, 130 67 Prague 3, Czech Republic,
{marek.dudas|ondrej.zamazal|svatek}@vse.cz

1 Introduction

We have recently proposed an innovative method of ontological engineering [1]: easing OWL ontology development by first creating a model in a less constrained language called PURO, allowing the engineer to focus on defining what is to be described by the ontology in an example real-world situation. The model is then automatically transformed to OWL, while allowing the user to choose the OWL encoding style.[1] The result is then finalized in a common tool like Protégé. We present an experimental implementation demonstrating how the above-described proposal can be enhanced by exploiting ontology matching, allowing to *reuse entities* from existing ontologies (as best practice in ontology design) or searching for a *combination of existing ontologies* that could cover the desired domain. We used ontology matching to search for relevant entities in existing ontologies for the given PURO model, visualize them and enable their easy reuse in our PURO-started ontology development approach. The approach is similar to vocabulary reuse tools like LOVER [4], however, the coupling with PURO is novel.

2 Design and Implementation

To find out whether there are OWL entities in existing ontologies that could cover some part of the PURO model, we simply take the OWL fragment generated from the model, match it to as many ontologies as possible, and present the results in a user-friendly way. To increase the chance of finding a match, the matching is run for four different encoding style OWL variants generated from the PURO model. We match the OWL fragment to Linked Open Vocabularies (LOV).[2] The matching is implemented as a RESTful service and executed in two steps. First, to speed up the process, we select candidate ontologies from LOV using vocabulary term search available from the LOV API, considering all terms from the OWL fragment. Second, the ontology candidates are matched to the OWL fragment using the state-of-the-art ontology matching tool LogMap [3]. We use a cached LOV snapshot to gain speed and reliability.

[1] E.g., whether a relationship will be represented by a property or class membership.
[2] http://lov.okfn.org/

The PURO-to-OWL transformation is made via a series of SPARQL update queries. To allow that, the PURO model is first serialized into RDF. The queries use RDF annotations to keep track of which PURO entities have been transformed to which OWL entities. This allows translating the OWL-to-OWL correspondences produced by the matching service to PURO-to-OWL mappings and visualize the latter in the original PURO model. Namely, entities (represented by nodes in the PURO model node-link visualization) with available mappings to OWL are highlighted by lines encircling them and labeled with the ontology IRI where the entity has been found. This way the user can see which parts of the PURO model are potentially covered by which existing ontologies. The visualization can therefore be useful on its own, simply suggesting relevant ontologies that can cover a given situation modeled in PURO.[3]

A list of the mappings found for a PURO entity is displayed when the PURO entity node is selected. By selecting an OWL entity from the list, the mapping is stored in the RDF PURO serialization and visualized in the PURO model as a separate node of different color. When the PURO-to-OWL transformation is run, the mapped entities are used by the SPARQL queries for transformation of corresponding PURO entities, instead of creating new OWL entities.

3 Results so Far and Future Work

The proof-of-concept implementation, available online as part of the web-based tool OBOWLMorph,[4] suggests that the basic idea is valid. We have yet to evaluate whether matching to several OWL encoding style variants brings any improvement compared to using just a default one. Compared to manual visualization of local ontology coverage [2], automated visualization is less complete and imprecise, but could serve to bootstrap the manual approach.

The research is supported by UEP IGA F4/28/2016. Ondřej Zamazal is supported by CSF 14-14076P.

References

1. Dudáš, M., Hanzal, T., Svátek, V., Zamazal, O.: OBOWLMorph: Starting ontology development from PURO background models. In: International Experiences and Directions Workshop on OWL. pp. 14–20. Springer (2015)
2. Dudáš, M., Hanzal, T., Svátek, V.: What can the ontology describe? visualizing local coverage in PURO modeler. In: VISUAL@EKAW (2014)
3. Jiménez-Ruiz, E., Grau, B.C.: Logmap: Logic-based and scalable ontology matching. In: International Semantic Web Conference. pp. 273–288. Springer (2011)
4. Schaible, J., Gottron, T., Scheglmann, S., Scherp, A.: LOVER: support for modeling data using linked open vocabularies. In: Proceedings of the Joint EDBT/ICDT 2013 Workshops. pp. 89–92. ACM (2013)

[3] We presented such a use case earlier [2], but only as an aid for the user allowing to highlight parts of the PURO model by hand. Now the same is done automatically.

[4] `http://goo.gl/IjTwlX`, please use login and password "om2016".