Semantic Interactive Ontology Matching: Synergistic Combination of Techniques to Improve the Set of Candidate Correspondences

Jomar da Silva[1], Fernanda Araujo Baião[1], Kate Revoredo[1], and Jérôme Euzenat[2]

[1] Graduated Program in Informatics, Department of Applied Informatics
Federal University of the State of Rio de Janeiro (UNIRIO), Brazil
{jomar.silva, fernanda.baiao,katerevoredo}@uniriotec.br
[2] Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP, LIG, F-38000 Grenoble France
Jerome.Euzenat@inria.fr

Abstract. Ontology Matching is the task of finding a set of entity correspondences between a pair of ontologies, i.e. an alignment. It has been receiving a lot of attention due to its broad applications. Many techniques have been proposed, among which the ones applying interactive strategies. An interactive ontology matching strategy uses expert knowledge towards improving the quality of the final alignment. When these strategies are based on the expert feedback to validate correspondences, it is important to establish criteria for selecting the set of correspondences to be shown to the expert. A bad definition of this set can prevent the algorithm from finding the right alignment or it can delay convergence. In this work we present techniques which, when used simultaneously, improve the set of candidate correspondences. These techniques are incorporated in an interactive ontology matching approach, called ALINSyn. Experiments successfully show the potential of our proposal.

Keywords: ontology matching, Wordnet, interactive ontology matching, ontology alignment, interactive ontology alignment

1 Introduction

Ontology matching seeks to discover correspondences between entities of different ontologies [1]. Ontology matching can be processed manually, semi-automatically or automatically [1]. Among the semi-automatic approaches, the ones that follow an interactive strategy stand out, considering the knowledge of domain experts through their participation [2]. The involvement of a domain expert is not always possible, as it is an expensive, scarce and time-consuming resource. However, when possible, better results have been achieved compared with automatic approaches.

An expert can be involved by giving his feedback to a correspondence, indicating whether or not it belongs to the alignment. Therefore, defining the set

of correspondences to show to the expert is one of the problems of these interactive techniques. If this set is not well defined, the final alignment may be imprecise or incomplete, or convergence to a good alignment can be delayed. Therefore, the scientific problem addressed in this paper is how to improve the set of correspondences to receive expert feedback.

This paper proposes ALINSyn, an approach that uses two techniques – a semantic and a structural – for the improvement of a given set of candidate correspondences. The semantic technique works by temporarily removing correspondences from the set of candidate correspondences. The structural technique interactively places part of the correspondences taken by the semantic technique back in the set of candidate correspondences. ALINSyn uses techniques used in the ALIN [13] system, that participated in OAEI 2016.

To evaluate ALINSyn, we defined ALINBasic, a basic ontology matching algorithm that generates and use a set of candidate correspondences to do the matching. Each of the two ALINSyn techniques was added to ALINBasic in order to modify the set of candidate correspondences generated by it, and the obtained alignments were compared. ALINSyn was also compared to state-of-the-art interactive ontology matching systems, showing the potential of our proposal.

This paper is structured as follows: Section 2 describes interactive ontology matching, Section 3 describes the ALINBasic algorithm, section 4 describes ALINSyn approach, by explaining its two steps, in section 5 the evaluation of the approach is made and the section 6 is the conclusion.

2 Interactive Ontology Matching

An ontology O is represented as a labeled graph G = (V, E, vlabel, elabel). The set of vertices V contains ontology entities such as concepts and properties. Edges in E (E $\subseteq$ V $\times$ V) represent structural relationships between entities. The edge labeling function elabel, which maps an edge (v, v) $\in$ E to a subset of the set SL of structural labels, which in turn specify the nature of the structural relationships between entities (e.g., subclassOf). Let LL denote the set of lexical labels associated with entities (e.g., name, documentation). Finally, the vertex labeling function, vlabel : V $\times$ LL $\rightarrow$ String, maps a pair (e, l) $\in$ V $\times$ LL to a string corresponding to the value of the lexical label l (e.g., name) associated with the entity e [3].

Given two ontologies O and O', an ontology matching is the process that aims to finding a set of correspondences (e, e'), where e and e' are entities in O and O', respectively. Interactive ontology matching takes advantage of user feedback to perform ontology matching.

Within the set of all possible correspondences between the entities of two ontologies, in the context of the interactive ontology matching, we distinguish two types of correspondences:

– Candidate correspondences are those possible correspondences that have been selected to be presented to the expert but have not yet received decision,

– Classified correspondences are those possible correspondences that have been selected to be presented to the expert and have received decision.

There are similarity measures, denoted sim, which map the possible correspondence (e, e') ∈ O×O' to a real number in [0, 1].

According to Meilicke and Stuckenschmidt [4], ontology matching algorithms that are based on the analysis of entity names usually have two phases:

– In the first phase, there is the creation of a set of candidate correspondences. To reduce the need to classify all possible correspondences (all pairs of entities) between two ontologies as belonging or not to alignment, the algorithm selects a subset called set of candidate correspondences;
– In the second phase, each correspondence in the set of candidate correspondences is classified by the ontology matching algorithm. In an interactive strategy, at least part of these correspondences is classified by the expert, and the other part can be classified by some automatic technique.

3   ALINBasic Algorithm

When the ontology matching is done interactively, we have two quality measures that are conflicting: the number of interactions with the expert and the quality of the generated alignment. It is interesting that a technique to be used in an algorithm of ontology matching can improve one of these qualities without worsening the other in an accentuated way. That is, to decrease the number of interactions without decreasing proportionally the quality of the generated alignment, or to increase the quality of the generated alignment without increasing proportionally the number of interactions with the expert.

In this paper two techniques will be presented, which used alone, cannot increase one of the qualities without considerably worsening the other. The first one, the semantic technique, decreases the number of interactions with the expert, but greatly decrease the quality of the generated alignment. The other, structural technique, enhances the quality of the generated alignment, but increasing a lot the number of interactions with the expert. But when used together, they can mitigate the disadvantages of each other, reducing the number of interactions without dramatically decreasing the quality of the generated alignment.

To evaluate the results of the two proposed techniques, three algorithms will be compared. An algorithm without the inclusion of any of the two techniques, called ALINBasic, a second algorithm, with the inclusion of the semantic technique, called ALINSem, and a third one with the inclusion of both the semantic and structural techniques, called ALINSyn. The two techniques are included in the algorithms as steps of these algorithms, so ALINSem is equivalent to the ALINBasic algorithm plus a semantic step that implements the semantic technique, and the ALINSyn algorithm is equivalent to the ALINSem algorithm plus a structural step that implements the structural technique.

The ALINBasic algorithm has two phases, as described by Meilicke and Stuckenschmidt [4]. The first phase selects candidate correspondences to be presented to the user. The second phase presents the selected candidate correspondence to the user and assigns them to the classified correspondences. Hence, in the end there are no candidate correspondences left.

In the phase of generating the candidate correspondences, only class correspondences, not property correspondences, are chosen, therefore, the ALINBasic algorithm finds only class correspondences.

The first phase of ALINBasic (Algorithm 1) will use the stable marriage algorithm with size list limited to 1 [5][6], where the pair will be formed by classes of the two ontologies to be aligned. Correspondences will be ordered by decreased similarity.

The stable marriage algorithm will be executed six times, each time with a different similarity metric (Jaccard, Jaro-Winkler, n-Gram, Wu-Palmer, Jiang-Conrath and Lin) and the result of the six executions will form a set of correspondences by the union of the six formed sets (Steps 1 to 4 of Algorithm 1). The process of selecting the similarity metrics was based on two criteria: available implementations and the result of these metrics in assessments, such as those carried out in [7] and [8]. Wu-Palmer, Jiang-Conrath and Lin are metrics that require a taxonomy to be computed [7], this taxonomy being provided, in this algorithm, by Wordnet.

From the set of correspondences formed by the union of the six sets all correspondence whose classes have exactly the same name will be classified as true (Step 5 of Algorithm 1). The correspondences selected by the running of stable marriage algorithm and not automatically classified will be the candidate correspondences (Step 6 of Algorithm 1).

Algorithm 1 Candidate correspondence generation

Input: Two ontologies to be aligned
Output: Candidate correspondences
 1: **for** Each one of the similarity metrics: Jaccard, Jaro-Winkler, n-Gram, Wu-Palmer, Jiang-Conrath and Lin **do**
 2:     Run stable Marriage Algorithm forming the set A_{sim} (being sim the corresponding similarity metric)
 3: **end for**
 4: Let $A = A_{Jaccard} \cup A_{Jaro-Winkler} \cup A_{n-Gram} \cup A_{Wu-Palmer} \cup A_{Jiang-Conrath} \cup A_{Lin}$
 5: Let B = Correspondences, from A, automatically classified as true by the fact that their entities have the same name
 6: Set of candidate correspondences = A - B

Then begins the classification phase of the candidate correspondences of the ALINBasic. At this phase all the candidate correspondences will be presented to the expert to receive his feedback.

For this, the concept of interaction with the expert will be used. An interaction with the expert corresponds to a question asked about at most three correspondences, as long as they pair-wisely have at least one of the entities in common. This is compliant with the OAEI definition [10]. For example, if the following correspondences are shown to the expert at the same time (ConferenceChair,Chair), (Chairman,Chair) and (Chairman,AssociatedChair), they will be counted as only one interaction since each correspondence has at least one entity of another correspondence. The number of interactions will be used as a comparison criterion between the various executions shown in this paper.

The ALINBasic algorithm can be seen in Algorithm 2.

Algorithm 2 ALINBasic

Input: Two ontologies to be aligned
Output: Alignment between the two ontologies
 1: Run candidate correspondence generation (Algorithm 1)
 2: **for** Each candidate correspondence **do**
 3:     Receive feedback (the candidate correspondence is transformed to classified correspondence)
 4: **end for**

---

4 ALINSyn Algorithm

4.1 Improving the Set of Candidate Correspondences

The objective of the ALINSyn algorithm is to decrease the number of interactions with the expert without decreasing in the same proportion the quality of the generated alignment. To achieve this objective, two steps, one semantic step and one structural step, are added to the ALINBasic algorithm to improve the set of candidate correspondences.

We first introduce another type of correspondence:

- Temporarily suspended correspondences are correspondences that are no longer candidate correspondences because of the semantic step. These correspondences can once again be candidate correspondences after the structural step.

The semantic step transforms some candidate correspondences to temporarily suspended correspondences. The structural step can transform some temporarily suspended correspondences to candidate correspondences again.

At the end of the non-interactive phase, by the use of the semantic step, all candidate correspondences that are not semantically equivalent will be transformed to temporarily suspended correspondences. In the interactive phase, by the use of the structural step, after each interaction with the expert, the expert's feedback can transform temporarily suspended correspondences in candidate correspondences if they have a particular structural relationship with a candidate correspondence that received positive feedback.

4.2   Semantic Step

The action of this step is to transform all candidate correspondences with semantically different entity names to temporarily suspended correspondences. The step will be added to the ALINBasic algorithm at the end of the generation phase.

The semantic step uses Wordnet. Wordnet consists of synonym sets called synsets [9]. A synset denotes a group of terms with the same meaning. The same term may appear in various synsets, as long as it has several meanings.

Comparison of entity names A head noun of a phrase is a noun to which all other terms are dependent [11]. Only correspondences relating entities whose name head nouns are in the same Wordnet synset will remain in the set of candidate correspondences after the semantic step. Before comparing the two entity names, a pre-processing step is necessary in order to extract the correct terms to be compared. An entity name can be atomic or composed. In the latter case, our approach searches for the head noun, and only this head noun will be used to compare the two entities. The rule we used for detection of head noun can be summarized as follows:

1. If the name contains a preposition (e.g. HeadOfDepartment) then the head noun is the token before the preposition.

2. Otherwise the head noun is the last token in the name.

Algorithm 3 Semantic step

Input: Candidate correspondences
Output: Temporarily suspended correspondences (ex-candidate correspondences)
 1: **for** Each candidate correspondence **do**
 2:     Choose the head noun of each entity of the name of the correspondence
 3:     Put the head noun of each name in the canonical form
 4:     **if** The two head nouns are not in the same wordnet synset **then**
 5:         Transform the candidate correspondence to temporarily suspended correspondence
 6:     **end if**
 7: **end for**

Example of the semantic step The semantic step can be seen in the Algorithm 3. To illustrate the semantic step we assume that we have the candidate correspondences selected by Algorithm 1 shown in Table 1. The first correspondence to be analyzed will be (Author, Regular_Author) (step 1 of Algorithm 3). The head noun of Author is Author, since it has only one word. The head noun for Regular_Author is Author, because it does not have a preposition and the last word is Author (step 2). The two head nouns are already in canonical form (step 3) and as they are the same word they are in the same synset, so they are not transformed to temporarily suspended correspondences.

Table 1. Correspondences before and after the semantic step, and after the first run of structural step

e	e'	before semantic step	after semantic step	after the first run of structural step
Author	Regular_author	Candidate correspondence	Candidate correspondence	Candidate correspondence
Chairman	Chair	Candidate correspondence	Temporarily suspended correspondence	Temporarily suspended correspondence
Co-author	Regular_author	Candidate correspondence	Temporarily suspended correspondence	Candidate correspondence
Paper	Paper	Candidate correspondence	Candidate correspondence	Candidate correspondence
Paper_Abstract	Abstract	Candidate correspondence	Candidate correspondence	Candidate correspondence
Person	Person	Candidate correspondence	Candidate correspondence	Classified as true
Subject_Area	Abstract	Candidate correspondence	Temporarily suspended correspondence	Temporarily suspended correspondence
Subject_Area	Program_Committee	Candidate correspondence	Temporarily suspended correspondence	Temporarily suspended correspondence

The second correspondence in the table is the correspondence (Chairman,Chair) (step 1). Chairman is considered a word because a term is only divided into words if it has hyphen, white space or is in camelcase (step 2). Since the two are in the canonical form (step 3) of the word their synsets are compared in Wordnet, and they are different. It is important to note that the most common meanings of words are searched for in wordnet, so Chair is the object of sitting and not Boss. Therefore this correspondence will be transformed to temporarily suspended correspondences (step 5).

The result after following these steps for all correspondences is shown in Table 1, in the column 'after the semantic step'.

Algorithm 4 ALINSyn

Input: Two ontologies to be aligned
Output: Alignment between the two ontologies
1: Run candidate correspondence generation (Algorithm 1)
2: Run semantic step (Algorithm 3)
3: **for** Each candidate correspondence **do**
4:     Receive feedback (the candidate correspondence is transformed to classified correspondence)
5:     Run structural Step (Algorithm 5)
6: **end for**

With the inclusion of the semantic step, the algorithm will be called ALINSem. As an illustration, this algorithm is the same as the algorithm ALINSyn (Algorithm 4) without the inclusion of step 5 (Run structural step). The results of ALINSem will be compared to the results of ALINSyn with the objective of verifying if the combined use of the semantic step and the structural step improves the result achieved by the use of the semantic step alone.

Algorithm 5 Structural Step
___
Input: Temporarily suspended Correspondences, Classified correspondences
Output: Candidate Correspondences (ex-temporarily suspended correspondences)
  1: **for** Each temporarily suspended correspondence **do**
  2:    **if** The two classes of the temporarily suspended correspondence are subclasses of classes of a correspondence classified as true **then**
  3:       Transform the temporarily suspended correspondence to candidate correspondence
  4:    **end if**
  5: **end for**
___

4.3 Structural Step

When only the semantic step is applied, experiments showed that the number of interactions with the expert were reduced, i.e. convergence was reached faster, however the final alignment lost in quality. This is because some true correspondences have been taken from the set of candidate correspondences because of semantic step. The main goal of the structural step is to recover part of the quality lost through the use of the semantic step by transforming some temporarily suspended correspondences again to candidate correspondences.

At each iteration, all temporarily suspended correspondences that are formed by subclasses of the classes of the correspondences that received positive feedback from the expert are transformed again to candidate correspondences. Tests were performed again using the two techniques, which showed that the use of both techniques makes the number of interactions decrease considerably, but with a much lower quality loss, in relation to the results obtained with the ALINBasic algorithm. The structural step can be seen in Algorithm 5.
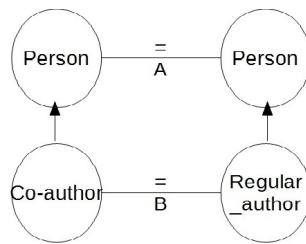

Fig. 1. Correspondences with classes that are subclasses of other correspondence classes

To illustrate the technique let us assume the situation described in Figure 1, where Co_author is a subclass of Person in the cmt ontology and Regular_author is a subclass of Person in the Conference ontology. Let us assume that the correspondence A (Person, Person) is a candidate correspondence and correspondence B (Co_author, Regular_author) is a temporarily suspended correspondence. If the

correspondence A receive positive feedback, the correspondence B by having its classes that are subclasses of the classes of A is transformed to candidate correspondence. The result of the structural step can be seen in Table 1 in the column 'after the first run of the structural step'. With the inclusion of the structural step in the interactive phase, the algorithm is called ALINSyn and can be seen in the Algorithm 4.

5 Evaluation Overview and Designed Analysis

The goal of the ALINSyn approach is to reduce the number of interactions with the expert without greatly diminishing the quality of the generated alignment. Thus a first research question is:

RQ1: Does the semantic step allow the ontology matching strategy to decrease the number of interactions with the expert? This question is answered with the use of the semantic step in the ALINBasic algorithm, as we see in the section "Analysis of the Results", which shows that the number of interactions with the expert has been reduced, but with a great drop in quality. That is why it is important to address other research questions.

RQ2: Can the expert feedback reduce the quality loss by the use of the semantic step?

RQ3: Does the use of both, semantic step and structural step together, generate an alignment with quality and number of interactions compatible with the state of the art proposals?

5.1 Conference dataset

Results obtained in the interactive matching of OAEI 2016 using the conference dataset were used to compare with the state of the art.

The OAEI interactive track is performed with percentages of expert correctness, from 70% to 100%. This paper has taken into consideration, for the evaluation of the execution of the ALINSyn and of other tools, 100% of correctness by the expert.

5.2 Analysis of the Results

After using the semantic step the results presented in Table 2 (ALINSem row) were reached, which shows that the use of the semantic step decreases the number of expert interactions, which responds to 'RQ1: Does the semantic step allow the ontology matching strategy to decrease the number of iterations with the expert ?', but there has been a sharp drop in quality, which shows the need to answer the question 'RQ2: Can the expert feedback reduce the quality loss by using the semantic step?'.

The recovery in the quality of the generated alignment was attempted by the use of structural step. After the inclusion of this new step the results shown in

Table 2 (ALINSyn row) were reached. That shows that the goal of the ALIN-Syn was achieved using the two techniques. The number of interactions with the expert decreased greatly, from 619 to 219, with the quality decreasing proportionally much less, the f-measure was from 0.79 to 0.75, what responds to RQ2: Can the expert feedback reduce the quality loss by the use of the semantic step ?. The result achieved is due to the combined effect of the joint use of the two techniques.

If we use only the semantic step we have a good decrease in the number of interactions with the expert, but with a sharp drop in quality. The subsequent use of the structural step, interactively, causes some of the lost quality to be recovered.

If we use only the structural step, without using the semantic step before, with all possible correspondences, not only the temporarily suspended correspondences, we would have an increase in quality, but a large number of correspondences would be added to the set of candidate correspondences, which would make the number of interactions with the expert too large (Table 2, ALINStr row). The transformation of candidate correspondences into temporarily suspended correspondences, through the semantic step, and the search, by the structural step, only among the temporarily suspended correspondence reduces the search space, which means that the number of interactions with the expert do not go up explosively.

The combined use of the two techniques results in a more balanced result, with a reduction in the number of interactions without a big loss of quality (Table 2, ALINSyn row).

Table 2. Comparison between different matching executions

	NI	Precision	F-measure	Recall
ALINBasic	619	0.92	0.79	0.70
ALINSem	152	0.90	0.69	0.57
ALINStr	3539	0.93	0.84	0.78
ALINSyn	219	0.91	0.75	0.65

5.3 Comparison among Tools that Participated in the OAEI Interactive Conference Track

OAEI provides a comparison among tool performance in the ontology matching process each year, and one of the ontology groups used is the conference dataset used in this paper [12].

Table 3 shows a comparison of some tools that participated in the OAEI 2016 interactive conference track. NI means number of interactions. In each interaction there can be up to three questions. "%" is the ratio of the number of interactions to the number of possible correspondences among all the alignments of the conference dataset.

Table 3 compares the performance of ALINSyn with some interactive tools that participated in OAEI 2016, with the expert hitting 100% of the answers in relation to the conference dataset. The use of the techniques shown in this work generates a high quality alignment, in cases where the expert does not make errors, what responds to 'RQ3: Does the use of the two techniques, semantic step and structural step together, generate an alignment with quality and number of interactions compatible with the state of the art ?'. The use of the two techniques combined puts ALINSyn among the best tools in the evaluation of OAEI 2016, when the expert hits 100% of the interactions.

Table 3. Comparison between some OAEI 2016 conference dataset interactive tracking tools and ALINSyn

	Number of questions	NI	%	Precision	F-measure	Recall
AML	270	271	0.215	0.912	0.799	0.711
ALINSyn	483	219	0.174	0.915	0.754	0.652
LogMap	142	142	0.113	0.886	0.723	0.610
XMap	4	4	0.003	0.837	0.681	0.574

6 Conclusion

Progress in information and communication technologies has made a large number of data repositories available, but with a great deal of semantic heterogeneity, which makes it difficult to integrate. A process that has been used to solve this problem is the ontology matching, which tries to discover the existing correspondences between the entities of two distinct ontologies, which in turn structures the concepts that define the data stored in each repository.

This work presented an interactive approach for ontology matching, based on manipulation of the set of candidate correspondences with techniques to decrease the number of interactions with the expert, without greatly reducing the quality of the alignment.

Two techniques were combined, one semantic and the other structural. The goal of the semantic technique was to decrease the number of interactions with the expert. The structural technique came in support of the semantic technique, and its objective was to decrease the quality loss resulting from the decrease in the number of interactions with the expert.

In order to evaluate if the techniques generated a decrease in the number of interactions without significantly lowering the quality, the executions of a basic algorithm with and without the techniques were compared, which showed that the techniques, when combined, reach their goal.

In addition, the quality of the alignment provided by the ALINSyn approach was compared to state of the art tools that have participated in the track of interactive ontology matching in OAEI 2016. The results obtained show that ALINSyn generates an alignment with a good quality in comparison to other

tools, with regard to precision, recall and f-measure, when the expert never makes mistakes, keeping the number of interactions within the range achieved by the other tools.

References

1. J. Euzenat and P. Shvaiko, Ontology Matching - Second Edition, 2. Springer-Verlag, 2013.
2. H. Paulheim, S. Hertling, and D. Ritze, Towards Evaluating Interactive Ontology Matching Tools, Lect. Notes Comput. Sci., vol. 7882, pp. 31-45, 2013.
3. S. Duan, A. Fokoue, and K. Srinivas, One Size Does Not Fit All: Customizing Ontology Alignment Using User Feedback, in Lecture Notes in Computer Science (LNCS), 2010, pp. 177-192.
4. C. Meilicke and H. Stuckenschmidt, A New Paradigm for Alignment Extraction, CEUR Workshop Proc., vol. 1545, pp. 1-12, 2015.
5. D. Gale and L. S. Shapley, College Admissions and the Stability of Marriage, Am. Math. Mon., vol. 69, no. 1, pp. 9-15, 2014.
6. R. W. Irving, D. F. Manlove, and G. OMalley, Stable marriage with ties and bounded length preference lists J. Discret. Algorithms, vol. 7, no. 2, pp. 213-219, 2009.
7. E. G. M. Petrakis, G. Varelas, A. Hliaoutakis, and P. Raftopoulou, Design and Evaluation of Semantic Similarity Measures for Concepts Stemming from the Same or Different Ontologies object instrumentality, Proc. 4th Work. Multimed. Semant., vol. 4, pp. 233-237, 2006.
8. M. Cheatham and P. Hitzler, String similarity metrics for ontology alignment, Lect. Notes Comput. Sci., vol. 8219 LNCS, no. PART 2, pp. 294-309, 2013.
9. F. Lin and K. Sandkuhl, A survey of exploiting WordNet in ontology matching, IFIP Int. Fed. Inf. Process., vol. 276, pp. 341-350, 2008.
10. D. Faria, Using the SEALS Client s Oracle in Interactive Matching, 2016. [Online]. Available: https://github.com/DanFaria/OAEI_SealsClient/blob/master/OracleTutorial.pdf
11. O. Svab-Zamazal and V. Svatek, Analysing ontological structures through name pattern tracking, Lect. Notes Comput. Sci., vol. 5268 LNAI, pp. 213-228, 2008.
12. M. Achichi, M. Cheatham, Z. Dragisic, J. Euzenat, D. Faria, A. Ferrara, G. Flouris, I. Fundulaki, I. Harrow, V. Ivanova, E. Jimenez-Ruiz, E. Kuss, P. Lambrix, H. Leopold, H. Li, C. Meilicke, S. Montanelli, C. Pesquita, T. Saveta, P. Shvaiko, A. Splendiani, H. Stuckenschmidt, K. Todorov, C. Trojahn, and O. Zamazal, Results of the Ontology Alignment Evaluation Initiative 2016, Proc. 11th Int. Work. Ontol. Matching co-located with 15th Int. Semant. Web Conf. (ISWC 2016) Kobe, Japan, Oct. 18, 2016., 2016.
13. J. Silva, F. A. Baião, and K. Revoredo, ALIN Results for OAEI 2016, CEUR Workshop Proc., vol. 1766, 2016.