A-LIOn - Alignment Learning through Inconsistency negatives of the aligned Ontologies

Sarah M. Alghamdi[1], Fernando Zhapa-Camacho[1] and Robert Hoehndorf[1,*]

[1]*Computational Bioscience Research Center, Computer, Electrical & Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology, 4700 KAUST, 23955 Thuwal, Saudi Arabia*

Abstract

Ontologies play important role in sharing and reusing knowledge. Several ontologies have been developed, describing a particular domain from a perspective of a community of developers and users. This has led to the existence of multiple ontologies covering the same or a different domain with varying degrees of variability. Ontology Alignment is typically used to identify correspondences between semantically related elements of two or more ontologies in order to address this problem.

We propose A-LIOn a system that learns alignments by combining lexical and semantic approaches as well as machine learning. The system utilizes OWL reasoning for negative sampling which is iteratively used to inform the correction of the learning of the alignments.

Keywords

Ontology Alignments, Ontology matching, Inconsistency negatives

1. Presentation of the System

Alignment **L**earning through **I**nconsistency negatives of the aligned **On**tologies (A-LIOn) is a system that discovers alignments between ontologies by combining various matching techniques, ranging from entity-level label matching and transformation learning to structure-level taxonomy learning and graph projection to logical reasoning and inconsistency detection and learning. This is the first participation of A-LIOn in the Ontology Alignment Evaluation Initiative (OAEI).

2. Proposed Methods

An ontology $\mathcal{O}$ can be defined over a signature $\mathcal{O} := (C, R, I; ax)$, where C is a set of concept names, R is a set of relation names, I is a set of individual names and ax is a set of axioms.

OAEI 2022

*Corresponding author.

‡These authors contributed equally.

sarah.alghamdi.1@kaust.edu.sa (S. M. Alghamdi); fernando.zhapacamacho@kaust.edu.sa (F. Zhapa-Camacho); robert.hoehndorf@kaust.edu.sa (R. Hoehndorf)

0000-0001-5544-7166 (S. M. Alghamdi); 0000-0002-0710-2259 (F. Zhapa-Camacho); 0000-0001-8149-5890 (R. Hoehndorf)

Given two ontologies $\mathcal{O}_s, \mathcal{O}_t$, the purpose of ontology alignment is to find the pairs of entities $(e_{\mathcal{O}_s i}, e_{\mathcal{O}_t j}) \in C_{\mathcal{O}_s} \times C_{\mathcal{O}_t}$ that are considered as being equivalent or standing in a subclass relation within certain contexts.

A graph is defined as a tuple $G = (E, R, T)$, where E is a set of entities names, R is a set of relations names and $T \subseteq E \times R \times E$ is a set of triples of the form (h, r, t).

A projection of an ontology into a graph is a mapping $f : \mathcal{O} \to G$ that usually transforms the ontology classes into graph nodes, ontology roles as graph relations and ontology axioms as graph triples following a particular set of rules.

Our method A-LIOn combines different matching techniques and consists of four main components:

- Learning lexical matching seeds.
- Graph construction from source and target ontology.
- Graph embedding and transformation learning.
- Consistency checking.

Those components cover element-wise, structure-wise, and formal semantics learning techniques. Element-wise techniques consider the entities in the ontology in isolation in order to find alignments, disregarding the fact that they are part of the ontology's structure. On the other hand, structure-wise techniques analyse the entities as part of their structure. Finally, there are systems that employ formal semantics learning techniques and logical inference to identify correspondences and repair inconsistencies.

2.1. Learning Lexical matching seeds

To begin learning ontology alignment, we need some known-to-be-positive seed alignments. We chose to align the classes of both ontologies with the same IRI, or lexically matched labels and relative IRIs. For lexical matching, we utilize fuzzy lexical matching, a method for finding approximate string matching with a retrieved score representing the similarity between one string to another. beginning with an exact matching score and then we decrease the threshold iteratively until a sufficient number of seeds are obtained or a minimal accepted threshold is reached.

2.2. Ontology Projection

We project each ontology as a graph in order to learn structure-level information from the source and the target ontologies. We evaluate two graph construction techniques:

- Subsumption hierarchy: in this method, we only utilized the subclass axioms asserted between the ontology classes to generate a directed graph for the source ontology and the target ontology. We evaluated this technique for Anatomy, Conference, Biodiversity and Ecology, and Material Sciences and Engineering tracks.
- OWL Projection: This method was proposed in [1], where OWL axioms are transformed directly into edges in the graph, and complex axioms are approximated in the graph to avoid the use of blank nodes. Despite the fact that this transformation method does not

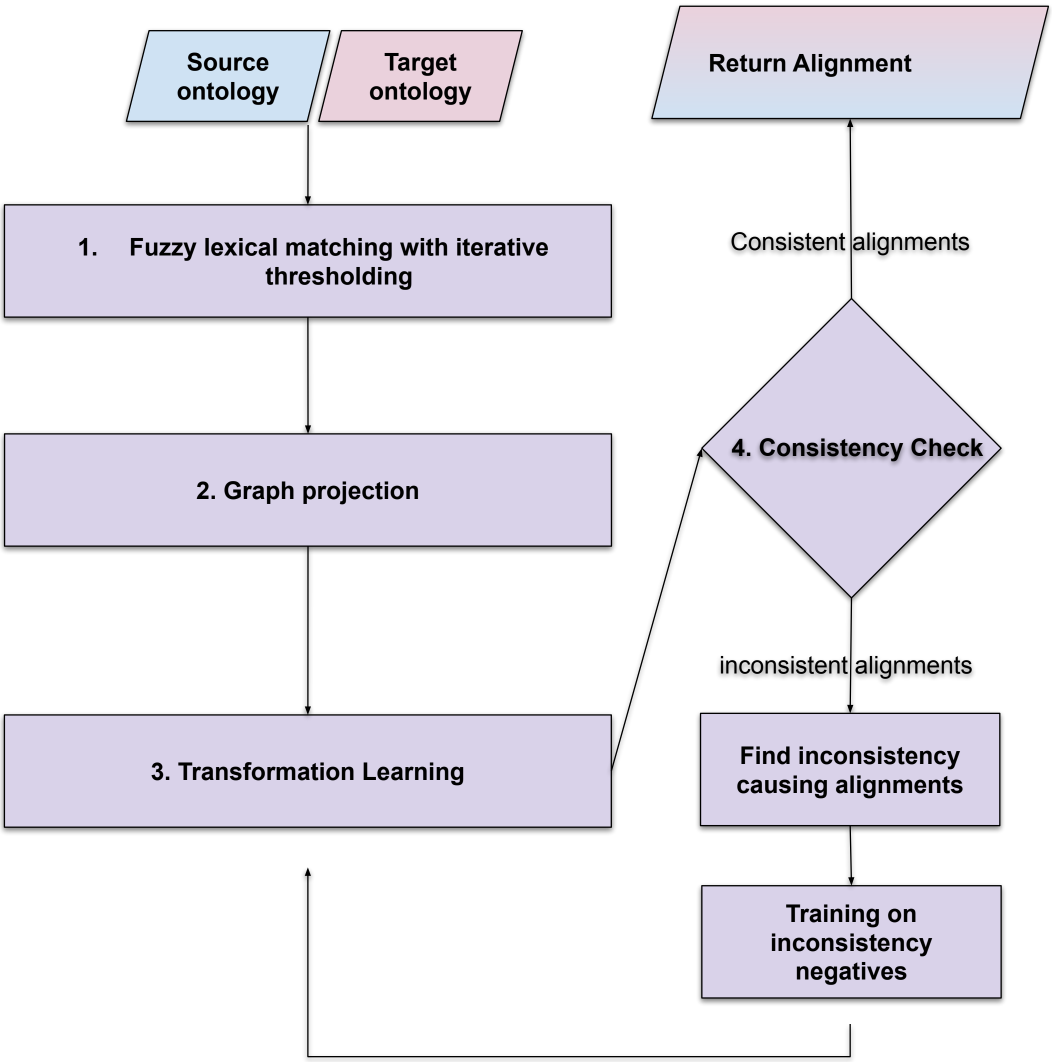

Figure 1: A-LIOn system component workflow

preserve exact logical relations, it enables correlation and learning alignments between classes of the source and target ontologies as well as within the same ontology. We evaluated this technique using the Phenotype ontology alignment.

2.3. Transformation Learning

After projecting an ontology, the result is a graph. Depending on the chosen projection method (Section 2.2), these graphs would encode the taxonomical structure or relational information found in the ontologies.

In our method, we start with two ontologies $\mathcal{O}_s$, $\mathcal{O}_t$, which, after applying the graph projection, will become two graphs $G_{\mathcal{O}_s}, G_{\mathcal{O}_t}$, respectively. When we deal with two graphs, there are several graph alignment methods that can align two graphs from a small number of seed alignments; we follow the method in [2].

To learn representations of the two graphs $G_{\mathcal{O}_s}, G_{\mathcal{O}_t}$, we define two vector spaces $V_{\mathcal{O}_s}, V_{\mathcal{O}_t}$, where the entities (nodes and edges) of each graph will be processed separately. To learn the graph embeddings we rely on knowledge graph embeddings methods such as TransR [3], optimizing the following loss function:

$$O_{s_{loss}} = \|Mr_s.c1 + r1 - Mr_s.c2\| \tag{1}$$

$$O_{t_{loss}} = \|Mr_t \cdot c1 + r1 - Mr_t \cdot c2\| \tag{2}$$

Simultaneously, we use a transformation $M : E_{G_{\mathcal{O}_s}} \rightarrow E_{G_{\mathcal{O}_t}}$ that takes the entities from the seeds we found earlier (Section 2.1) from the source embedding space to the target space, using the following loss:

$$A_{loss} = \|M \cdot c_s - c_t\| \tag{3}$$

2.4. Inconsistency negatives learning

OWL ontologies are based on Description Logics and facilitate the use of automated reasoners, which in turn facilitate computing entailments of statements from the asserted ontology axioms. In addition, these inferences can be investigated to determine if a class in an ontology is satisfiable or unsatisfiable. A class is unsatisfiable if it cannot have any instances (i.e., the axioms constrain the class in a contradictory way); an ontology is inconsistent if it has at least one instance of a logical contradiction [4, 5]. To determine unsatisfiable classes, we utilize the ELK reasoner [6] to find alignments that lead to inconsistencies. In order to find inconsistency in aligning $\mathcal{O}_s$ and $\mathcal{O}_t$, we first merge both ontologies (i.e., we combine their axioms into a new ontology) and add all alignments predicted by our model as equivalence class axioms to the merged ontology $\mathcal{O}_{merged}$. We define this ontology as follows $\mathcal{O}_{merged} := (C_{\mathcal{O}_s} \cup C_{\mathcal{O}_t}, R_{\mathcal{O}_s} \cup R_{\mathcal{O}_t}, I_{\mathcal{O}_s} \cup I_{\mathcal{O}_t}, ax_{\mathcal{O}_s} \cup ax_{\mathcal{O}_t}, A)$, where C_i is a set of concepts from ontology i, R_i is a set of relations form ontology i, I_i is a set of individuals form ontology i, and ax_i is a set of axioms from ontology i, A is the predicted alignments.

Then we use the ELK reasoner [6] to identify unsatisfiable classes in the merged ontology. If we identify an unsatisfiable class we generate explanations for the entailment generated by Elk; an explanation consists of a small set of axioms from which the unsatisfiability follows directly; we specifically identify any of the equivalence class axioms we have added within the generated explanations, as these are likely causing the class to become unsatisfiable. We remove the equivalence class axioms from the merged ontology and iterate. Finally, we return to the transformation learning step with an updated loss to optimize for alignment learning as follows:

$$A_{loss} = \|(M \cdot c_s - c_t) - (M \cdot c_{ns} - c_{nt})\| \tag{4}$$

where c_s, c_t are positive class pairs form source ontology and target ontology, respectively, c_{ns}, c_{nt} are pairs of classes which gave rise to unsatisfiable classes and which we removed in the repair step. The new iteration of our method now uses these pairs as negatives during training the alignment of both ontologies.

Table 1
Phenotype use case test results on last year 5-consensus. We show the results for different variations of A-LIOn, starting with the use of subsumption hierarchy graph(SH), projected graph (P)

	number of alignments	Precision	Recall	F-score
LogMap	2,136	0.767	0.908	0.831
AML	2,029	0.810	0.910	0.857
ATMatcher	769	0.968	0.412	0.578
TOM	306	0.101	0.140	0.117
A-LIOn - (SH)	700	0.986	0.382	0.551
A-LIOn - (P)	1078	0.8223	0.7318	0.7744

3. Results

For this year's evaluation, we tested A-LIOn in three tracks: Anatomy, Conference and Material Sciences and Engineering (MSE). We have also tested our system on the phenotypes track using last year's evaluation tests.

3.1. Phenotype matching

We tested the OWL projection method in the problem of aligning phenotype ontologies. To test this approach, we utilized the datasets provided last year [7] for aligning Human phenotype ontology (HP) [8] and Mammalian Phenotype Ontology (MP) [9]. The seed alignments we used are exactly matching IRIs of classes, as well as lexical alignments for HP and MP classes only. We tested two different approaches for generating the graphs from source and target ontologies (Section 2.2). Results are shown in Table 3.1 where we included the results for some of the participating systems from last year for comparison [10, 11, 12, 13]. Comparing the results of the various graph generation techniques, we found that using the OWL projection in the problem of phenotype mappings allows for the discovery of more mappings, whereas the subsumption hierarchy produces alignments with high precision but finds fewer alignments, thereby decreasing the recall.

4. Conclusion

A-LIOn is a system that incorporates both entity-level and structure-level information in learning alignments between two ontologies; A-LIOn also uses logical reasoning to correct alignments that are likely faulty because they lead to unsatisfiable classes, and incorporates the results of this symbolic step in the learning process to generate new negatives. In the future, we plan to make our system able to learn better parameters based on the input ontologies features and self-evaluate the predicted alignment. For example, using a different set of parameters for the first task on Material Sciences and Engineering track allowed us to increase the F-score by 3.3%. A further improvement will be the use of language models in seed selection.

References

[1] J. Chen, P. Hu, E. Jimenez-Ruiz, O. M. Holter, D. Antonyrajah, I. Horrocks, Owl2vec*: Embedding of owl ontologies, Machine Learning 110 (2021) 1813–1845.

[2] M. Chen, Y. Tian, M. Yang, C. Zaniolo, Multilingual knowledge graph embeddings for cross-lingual knowledge alignment, arXiv preprint arXiv:1611.03954 (2016).

[3] Y. Lin, Z. Liu, M. Sun, Y. Liu, X. Zhu, Learning entity and relation embeddings for knowledge graph completion, in: Twenty-ninth AAAI conference on artificial intelligence, 2015.

[4] L. T. Slater, G. V. Gkoutos, R. Hoehndorf, Towards semantic interoperability: finding and repairing hidden contradictions in biomedical ontologies, BMC Medical Informatics and Decision Making 20 (2020) 1–13.

[5] J. Martinez-Gil, S. Yin, J. Küng, F. Morvan, Matching large biomedical ontologies using symbolic regression, in: The 23rd International Conference on Information Integration and Web Intelligence, 2021, pp. 162–167.

[6] Y. Kazakov, M. Krötzsch, F. Simančík, Elk: a reasoner for owl el ontologies, System Description (2012).

[7] M. Pour, A. Algergawy, F. Amardeilh, R. Amini, O. Fallatah, D. Faria, I. Fundulaki, I. Harrow, S. Hertling, P. Hitzler, et al., Results of the ontology alignment evaluation initiative 2021, in: CEUR Workshop Proceedings 2021, volume 3063, CEUR, 2021, pp. 62–108.

[8] S. Köhler, M. Gargano, N. Matentzoglu, L. C. Carmody, D. Lewis-Smith, N. A. Vasilevsky, D. Danis, G. Balagura, G. Baynam, A. M. Brower, et al., The human phenotype ontology in 2021, Nucleic acids research 49 (2021) D1207–D1217.

[9] C. L. Smith, J. T. Eppig, The mammalian phenotype ontology as a unifying standard for experimental and high-throughput phenotyping data, Mammalian genome 23 (2012) 653–668.

[10] E. Jiménez-Ruiz, Logmap family participation in the oaei 2021, in: CEUR Workshop Proceedings, volume 3063, 2021, pp. 175–177.

[11] D. Faria, B. Lima1, F. Couto, M. Silva, C. Pesquita, Aml and amlc results for oaei 2021, in: The 23rd International Conference on Information Integration and Web Intelligence, 2021, pp. 131–136.

[12] S. Hertling, H. Paulheim, Atbox results for oaei 2021, in: CEUR Workshop Proceedings, volume 3063, RWTH Aachen, 2021, pp. 137–143.

[13] D. Kossack, N. Borg, L. Knorr, J. Portisch, Tom matcher results for oaei 2021, in: CEUR Workshop Proceedings, volume 3063, RWTH, 2022, pp. 193–198.