The Knowledge Engineering Review, Vol. 35:e1 (22pp.), 2020
DOI: 10.1017/S0269888919000249

ALIN: Improving Interactive Ontology Matching by Interactively Revising Mapping Suggestions

Jomar da Silva[1], Kate Revoredo[2], Fernanda Baião[3], Jérôme Euzenat[4]

[1] *Graduate Program in Informatics, Federal University of Rio de Janeiro (UFRJ), Brazil*
E-mail: jomar.silva@uniriotec.br
[2] *Graduate Program in Informatics, Federal University of Rio de Janeiro (UFRJ), Brazil*
E-mail: katerevoredo@ppgi.ufrj.br
[3] *Department of Informatics, Pontifical Catholic University of Rio de Janeiro (PUC-Rio), Brazil*
E-mail: fbaiao@inf.puc-rio.br
[4] *Univ. Grenoble Alpes, INRIA, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France*
E-mail: Jerome.Euzenat@inria.fr

Abstract

Ontology matching aims at discovering mappings between the entities of two ontologies. It plays an important role in the integration of heterogeneous data sources that are described by ontologies. Interactive ontology matching involves domain experts in the matching process. In some approaches, the expert provides feedback about mappings between ontology entities, i.e., these approaches select mappings to present to the expert who replies which of them should be accepted or rejected, so taking advantage of the knowledge of domain experts towards finding an alignment. In this paper, we present ALIN, an interactive ontology matching approach which uses expert feedback not only to approve or reject selected mappings, but also to dynamically improve the set of selected mappings, i.e., to interactively include and to exclude mappings from it. This additional use for expert answers aims at increasing in the benefit brought by each expert answer. For this purpose, ALIN uses four techniques. Two techniques were used in previous versions of ALIN to dynamically select concept and attribute mappings. Two new techniques are introduced in this paper: one to dynamically select relationship mappings and another one to dynamically reject inconsistent selected mappings using anti-patterns. We compared ALIN with state-of-the-art tools, showing that it generates alignment of comparable quality.

1 Introduction

An ontology is a formal and explicit artifact that represents a shared conceptualization on a particular domain, structurally consisting of a collection of interconnected entities (concepts, attributes of concepts and relationships between concepts). One advantage of the use of ontologies is to improve communication, not only among humans but also among application systems using these ontologies. This, in turn, fosters interoperability. However, due to the recent and steeply increasing advancements of the semantic technologies and the web, there are scenarios in which several ontologies exist for the same domain, each using different entities to refer to the same real-world object. These scenarios raise communication issues among people or application systems that use these different ontologies.

Ontology matching has successfully addressed these problems by discovering correspondences (mappings) between entities of different ontologies (Euzenat and Shvaiko, 2013). One of the possible approaches to the ontology matching process is the interactive one that involves domain experts in the matching process. This involvement can be used to improve the results over fully automatic approaches (Paulheim et al., 2013).

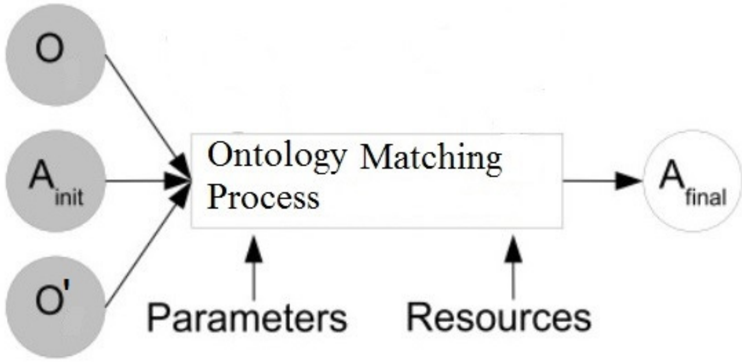

Figure 1 Ontology Matching Process - based on (Euzenat and Shvaiko, 2013)

In some interactive approaches, the domain expert provides feedback about mappings between ontology entities. These approaches select mappings to present to the expert who assesses which of them should be accepted or rejected. They take advantage of the domain expert knowledge to generate an alignment, and have, as their two main steps, the selection of the mappings to receive expert feedback and the propagation of this feedback. These approaches vary in techniques for these two steps.

Current approaches select mappings before beginning the propagation step, i.e., they do not select new mappings during the propagation step. The problem with these approaches is that they do not take advantage of the expert's knowledge to select new mappings.

We focused on this problem, the first time, with $ALIN_{Syn}$ (Da Silva et al., 2017), which interactively selected concept mappings. In a second time, we developed a new technique, which interactively selected attribute mappings, which gave rise to the other approach, $ALIN_{Attr}$ (Da Silva, Revoredo, Baião and Euzenat, 2018). Alin uses the two techniques implemented in $ALIN_{Syn}$ and $ALIN_{Attr}$.

The contributions of this research are the definition of a new structural technique to interactively select new relationship mappings and the interactive use of anti-patterns, so using the expert feedback to improve the set of selected mappings.

The Ontology Alignment Evaluation Initiative (OAEI) is a coordinated international initiative whose one of the goals is to assess the strengths and weaknesses of ontology matching systems. Alin has participated in OAEI 2018 and was evaluated in the interactive matching track, with the Conference and Anatomy data sets. When compared with the other tools, Alin got the highest precision in both data sets (Algergawy et al., 2018). In the Conference track, Alin achieved the highest recall among all tools. On the Anatomy track, Alin has not reached a prominent recall, for reasons this paper will explain.

The rest of this paper is organized as follows. Section 2 reviews Interactive Ontology Matching. In Section 3, we present the related works. Section 4 fully details Alin algorithms. Section 5 describes our evaluation methodology and discusses experimental results. Finally, Section 6 concludes the paper.

2 Interactive Ontology Matching

The ontology matching process starts with two ontologies, O and O', and returns an alignment A_{final} between the input ontologies. An alignment is a set of mappings between entities e and e' from O and O' respectively, defining pairs of entities. The ontology matching process can receive as input an initial alignment A_{init}. Besides, it can get input parameters and can access external resources, such as a dictionary (Figure 1).

A mapping asserts a semantic relationship between its entities, such as *disjunction*, *subsumption*, or *equivalence*. In this paper, we only consider the equivalence relationship. The ontology

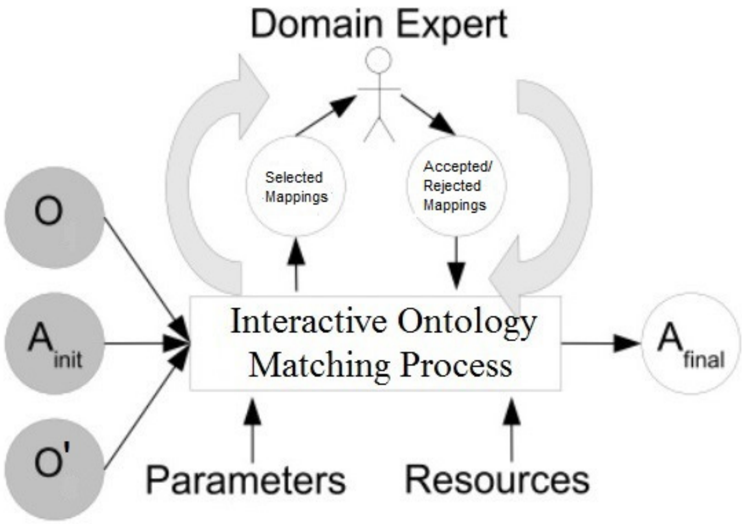

Figure 2 Interactive Ontology Matching Process - based on (Paulheim et al., 2013)

matching system is responsible to find, among all the possible mappings, the ones to be included in the alignment.

Given that two ontologies representing the same domain may present different types of heterogeneity (terminological, structural or semantic, among others), we need a combination of distinct techniques to correctly discover mappings between their entities. We can see an ontology matching tool as a collection of several matching components, each implementing a specific technique dealing with a specific heterogeneity type (Ngo et al., 2013).

Interactive ontology matching processes involve domain experts. The type of expert involvement used in this paper is through feedback on selected mappings. This type takes benefit from the domain expert knowledge by submitting selected mappings to the expert who assesses which of them should be accepted or rejected (Figure 2).

One common type of approach for this type of expert involvement is to divide the process into two main steps: the selection of mappings and the propagation of expert feedback. The current tools that follow this type of approach only select mappings in the selection step. They do not select any new mappings in the propagation step, thus not taking advantage of the expert feedback to select mappings.

This paper presents an approach where various techniques are used to use the expert feedback to improve the set of selected mappings, including a new technique for interactively select relationship mappings, and also including the interactive use of mapping anti-patterns.

This paper shows that the new technique causes an increase in F-measure, and, if the approach uses expert feedback as the input of the technique, then the approach requires fewer interactions with the expert to do so. It also shows that the interactive use of mapping anti-patterns decrease the number of interactions with the expert.

3 Related Work

There are at least three areas in which users may be involved in a matching solution (Euzenat and Shvaiko, 2013):

- by providing initial alignments to the system (before matching),
- by configuring (which includes strategy and parameter selection) and tuning the system, and
- by providing feedback to matchers (during or after the automatic matching process) in order for them to adapt their results.

The type of user, a domain expert, involvement used in this paper is through feedback to matchers. They benefit from the domain expert knowledge by submitting selected mappings to the expert who assesses which of the mappings should be accepted or rejected. One common type of approach for this type of expert involvement is to divide the process into two main steps: the selection of mappings and the propagation of expert feedback.

In the propagation step, different approaches can use expert feedback in different ways. Some of these approaches automatically classify selected mappings using a threshold. The threshold is a value that indicates whether a mapping should be automatically accepted (or rejected) when its similarity values are greater (or smaller) than the threshold. Expert feedback is used to calculate this threshold (Paulheim and Hertling, 2013; Hertling, 2012; Duan et al., 2010; Shi et al., 2009; Chunhua et al., 2015). Some approaches automatically classify some selected mappings using a classifier. These approaches use expert feedback to create the training data set for learning the classifiers (Lopes et al., 2015; To H. et al., 2009). Others approaches use expert feedback to modify the weight of similarity metrics (Duan et al., 2010; Shi et al., 2009; Balasubramani et al., 2015) or to change the value of similarity metrics (Cruz et al., 2012, 2014).

Alin and its previous versions use the expert feedback to improve the set of selected mappings. Others approaches also use expert feedback to improve the set of selected mappings (Lambrix and Kaliyaperumal, 2016; Jiménez-Ruiz et al., 2012; Faria et al., 2013).

$ALIN_{Syn}$ (Da Silva et al., 2017), an earlier version of Alin, in the mapping selection phase, selects concept mappings through semantic and lexical similarity metrics. Afterward, it automatically accepts the selected concept mappings with the same names. Then, it suspends some of these mappings using a semantic technique. In the propagation phase, $ALIN_{Syn}$ uses expert feedback to unsuspend some of these suspended mappings, returning them to the set of selected mappings, thus improving it. Alin uses all the techniques used in $ALIN_{Syn}$.

$ALIN_{Attr}$ (Da Silva, Revoredo, Baião and Euzenat, 2018), another earlier version of Alin, has the same mapping selection phase as $ALIN_{Syn}$, but without suspending selected concept mappings. In the programming phase, $ALIN_{Attr}$ uses expert feedback to select attribute mappings. Alin uses the attribute mapping selection technique used in $ALIN_{Attr}$.

SAMBO (Lambrix and Kaliyaperumal, 2016) takes advantage of the expert feedback to improve the set of selected mappings, removing the selected mappings that conflict with the accepted mappings, according to a reasoner.

Logmap (Jiménez-Ruiz and Grau, 2011; Jiménez-Ruiz et al., 2012), in the selection step, select mappings based on similarity metrics seeking to achieve a high recall. Still, in this step, the approach also uses automatic, non-interactive, techniques for improving the precision of the set of selected mappings. In the propagation step, the approach takes advantage of the expert feedback by identifying mappings that are inconsistent with the accepted mappings, according to a reasoner, and removes them from the set of selected mappings.

AML (Faria et al., 2013) initially selects mappings based on lexical, semantic and structural similarities. AML places, in automatic mode, mappings that are above a given threshold in the alignment, which then goes through a repair process.

In the interactive mode, AML uses two thresholds. Those selected mappings that are below the lower threshold, are automatically rejected. Selected mappings above the higher threshold are automatically accepted unless they generate an inconsistency. In such a case, they are submitted to the expert. Concerning the selected mappings between the two thresholds, those which generate inconsistency are automatically rejected; the others are submitted to the expert.

AML uses the expert feedback to remove mappings from the set of selected mappings. If a selected mappings is accepted, all selected mappings that have an entity in common with it are rejected, as well as all those that are inconsistent with it.

In the current approaches, the impact of expert feedback on the set of selected mappings is the removal of mappings from it, to increase the precision and decrease the number of interactions.

ALIN and its previous versions also focus on using the expert feedback to select mappings to improve the set of selected mappings.

ALIN uses, besides the techniques of the earlier versions, another that selects relationships mappings associated with accepted concept mappings, and mapping anti-patterns.

In the approach of (Chunhua et al., 2015), there are formulas that associate relationships mappings to the concept mappings of the concepts associated with these relationships. The formulas serve to increase the weight of these concept mappings, which increases the likelihood of the approach automatically include them into the final alignment. ALIN selects relationships mappings, to receive the expert feedback, associated with accepted concepts mappings.

ALIN uses mapping anti-patterns to remove mappings from the set of selected mappings. ASMOV (Jean-Mary et al., 2009) uses concepts similar to mapping anti-patterns, which it calls semantic verification inferences. ASMOV chooses from the mappings found in one of these semantic verification inferences to take out of the alignment. ASMOV is an automatic matching tool, so unlike ALIN, it does not use expert feedback to identify which mappings to exclude from alignment.

Section 4 further explains the ALIN approach.

Several areas use user knowledge to improve their performance. In information search and retrieval area, for example, feedback data based on direct interaction (e.g., clicks, scrolling) as well as on user profiles/preferences have been proven valuable for personalizing the search process, e.g., from how queries are understood to how relevance is assessed (Belkin et al., 2010). However, in such cases, the feedback is provided by users and is used for answering the query of the same user (in a single session for relevance feedback, or across session for personalization). Ontology matching could take advantage of user feedback to improve alignments. However, interactive ontology matching is usually disconnected from the use of the resulting alignments, which may be used in different contexts. It relies on expert feedback, as opposed to user feedback, in order to find the correct alignment, not one that suits the need of a particular user. In addition, relevance feedback in information retrieval is rather used to give more or less weight to characteristics of the instances. Here it is used to revise the mappings based on more logical constraint propagation, e.g., anti-patterns.

4 The ALIN Approach

In this section, we describe our approach for interactively matching two ontologies. The next subsection presents the overall ALIN procedure. Section 4.2 presents the terminology used to explain our approach. Section 4.3 describe the ALIN algorithms.

4.1 General principles

ALIN handles three sets of mappings:

Accepted definitely to be retained in the alignment

Selected to be decided

Suspended selected but filtered out

The procedure of ALIN is:

1. Select mappings: select the first mappings;
2. Filter mappings: suspend some selected mappings, using semantic criteria for that;
3. Ask expert: accepts or rejects selected mappings
4. Propagate: select new mappings, reject some selected mappings or unsuspend some suspended mappings (depending on newly accepted mappings)
5. Go back to 3 as long as there are undecided selected mappings

All versions of ALIN follow this general scheme. They mainly differ on the way they implement the Propagation step.

There were two previous versions of ALIN: $ALIN_{Syn}$, and $ALIN_{Attr}$. The problem addressed by these versions was to improve the set of selected mappings using expert feedback to select better mappings to present to the expert. These versions differ from the type of mappings that is presented to the expert: $ALIN_{Syn}$, uses concept mappings only, and $ALIN_{Attr}$, uses attribute mappings in addition.

One contribution of this paper, concerning the previous ALIN versions, is a new structural technique to select relationship mappings, implemented by the *SelectRelationshipMappings* algorithm (Algorithm 9) used in the Propagate phase. The goal of the new technique is to increase the F-measure, and if the expert feedback is used to assist it, do it more efficiently (more increase per provided feedback).

In order to show the benefits of exploiting expert feedback to choose questions, we have developed two new versions of ALIN: $ALIN_{NoRel}$ and $ALIN_{Aut}$.

Another contribution is the use of mapping anti-patterns to reject the selected mappings that are inconsistent with the mappings accepted by the expert. ALIN uses various algorithms to avoid and undo these anti-patterns (Algorithms 10, 11, 12 and 13).

To identify the benefits of the interactive use of anti-patterns, we have developed another version of ALIN: $ALIN_{NoAP}$.

4.2 Terminology

In an ontology O, its entities e can be *concepts (c)*, which are concepts in a domain, organized in a subconcept-superconcept hierarchy, or *properties*, which describe attributes (a) of concepts and relationships (r) between concepts. Table 1 provides predicates that can be used to model an ontology.

Table 1 Predicates for ontology description (based on (Chunhua et al., 2015))

Predicate	Description
concept(c,O)	c is a concept of ontology O
attr(a,O)	a is an attribute of ontology O
relat(r,O)	r is a relationship of ontology O
entity(e,O)	e is an entity of ontology O
sub(c_1,c_2)	c_1 is subconcept of c_2
sub(c_1,c_2,n)	c_1 is subconcept of c_2 with a maximum of n depth levels
dis(c_1,c_2)	c_1 is disjoint of c_2
rconcept(r,c_1,c_2)	r is a relationship between c_1 and c_2
aconcept(a,c)	a is an attribute of concept c

Given two ontologies O and O', an ontology matching approach will search for mappings between the entities of these two ontologies. In this paper, all entities from O have no prime ($'$); all entities from O' have a prime ($'$). We use the notation $\langle e, e' \rangle$ to represent a mapping between two entities, $\langle c, c' \rangle$ between two concepts, and $\langle r, r' \rangle$ and $\langle a, a' \rangle$ between two relationships and two attributes.

4.3 Algorithms

The ALIN algorithms (Algorithms 1 to 14) use the variables described below:

- Selected - Set of selected mappings;
- Suspended - Set of suspended mappings;

- Accepted - Set of accepted mappings;
- SetofSimMet - Set of similarity metrics;
- SetofSemSim - Set of semantic similarity metrics, it is a subset of the set of similarity metrics;
- SimMet - one similarity metric;

ALIN uses the functions defined in Table 2 in its algorithms.

Table 2 Functions used in the ALIN algorithms

Function	Description
$ename(e_1, e_2)$	returns true if the name of entity e_1 is equal to the name of entity e_2, else return false.
$highsum(A)$	returns the mapping from the set of mappings A with the highest sum of similarity metrics.
$concepts(O)$	returns the set of all concepts of the ontology O.
$attributes(c)$	returns the set of all attributes of the concept c.
$multent(\langle e_1, e'_1\rangle, \langle e_2, e'_2\rangle)$	returns true if $\langle e_1, e'_1\rangle$ and $\langle e_2, e'_2\rangle$ are in a multiple-entity anti-pattern, else it returns false. See Definition 4.2.
$antipat(\langle e_1, e'_1\rangle, \langle e_2, e'_2\rangle)$	returns true if $\langle e_1, e'_1\rangle$ and $\langle e_2, e'_2\rangle$ are in any of the three anti-patterns, else it returns false. See Definitions 4.2 to 4.4.
$simvalue(\langle e, e'\rangle, SimMet)$	returns the value of the metric SimMet between the entities e and e'.
$isconcmap(\langle e, e'\rangle)$	returns true if e and e' are concepts, else it returns false.
$isattrmap(\langle e, e'\rangle)$	returns true if e and e' are attributes, else it returns false.
$isrelatmap(\langle e, e'\rangle)$	returns true if e and e' are relationships, else it returns false.
$isaccepted(\langle e, e'\rangle)$	returns true if the expert accepts the mapping $\langle e, e'\rangle$, else it returns false.
$trigger(\langle c_1, c'_1\rangle, \langle c_2, c'_2\rangle, \langle r, r'\rangle)$	returns true if there is a trigger, between $\langle c_1, c'_1\rangle$ and $\langle c_2, c'_2\rangle$, for the selection of $\langle r, r'\rangle$, else it returns false. See Definition 4.1.

ALIN (Algorithm 1) has two main steps: the selection step and the propagation step. The selection step is responsible for defining the initial selected, accepted and suspended mappings. The propagation step is where the expert provides feedback to the selected mappings and this feedback is propagated.

The following subsections detail these two steps.

4.3.1 Selection Step

The ALIN selection step (from line 1 to line 3 of Algorithm 1) describes the activities for defining the initial set of selected mappings, the initial set of suspended mappings and the initial alignment (accepted mappings). The algorithm has as input the two ontologies to be matched and, as parameters, a set of similarity metrics chosen by the user. The chosen similarity metrics can be semantic or lexical.

The algorithm starts selecting concept mappings (line 1 of Algorithm 1) using the *Select-ConceptMappings* algorithm (Algorithm 2). Inside this algorithm, the *SelectMappingsPerMetric* algorithm (Algorithm 3) selects mappings for each similarity metric. The *SelectMappingsPerMetric* algorithm treats the matching problem as a stable marriage problem with size list limited

Algorithm 1 *ALIN TopLevel*

Input: O, O', SetofSimMet
Output: Accepted
/*Selection step*/
1: Selected $\leftarrow$ *SelectConceptMappings* (O, O', SetofSimMet);
2: Accepted,Selected $\leftarrow$ *AcceptConceptMappings* (Selected);
3: Suspended,Selected $\leftarrow$ *SuspendConceptMappings* (Selected,SetofSemSim);
/*Propagation step*/
4: **for each** $\langle e, e' \rangle \in$ Accepted **do**
5: Selected,Suspended $\leftarrow$ *PropagateAcceptedMapping* ($\langle e, e' \rangle$, Accepted, Selected, Suspended, SetofSimMet);
6: **end for**
7: **while** Selected $\neq \emptyset$ **do**
8: $\langle e, e' \rangle$ = highsum(Selected);
9: receive expert feedback on $\langle e, e' \rangle$;
10: remove $\langle e, e' \rangle$ from Selected;
11: **if** *isaccepted*($\langle e, e' \rangle$) **then**
12: add $\langle e, e' \rangle$ to Accepted;
13: Selected,Suspended $\leftarrow$ *PropagateAcceptedMapping* ($\langle e, e' \rangle$, Accepted, Selected, Suspended, SetofSimMet);
14: **end if**
15: **end while**
16: **return** Accepted

to 1 (Gale and Shapley, 1962; Irving et al., 2009), i.e., the algorithm only selects one mapping if similarity value between the two entities of the mapping is the highest considering all the mappings with at least one of these entities.

The *SelectMappingsPerMetric* algorithm is executed once for each chosen metric, where it generates a set of mappings for each metric (lines 4 and 5 of Algorithm 2). The union of these sets defines the initial set of selected mappings (from line 6 to line 10 of Algorithm 2).

Algorithm 2 *SelectConceptMappings*

Input: O, O', SetofSimMet
Output: Selected
1: Selected = $\emptyset$;
2: $SCO \leftarrow concepts(O)$;
3: $SCO' \leftarrow concepts(O')$;
4: **for each** SimMet $\in$ SetofSimMet **do**
5: M $\leftarrow$ *SelectMappingsPerMetric* (SCO,SCO',SimMet);
6: **for each** $\langle e, e' \rangle \in$ M **do**
7: **if** $\langle e, e' \rangle \notin$ Selected **then**
8: add $\langle e, e' \rangle$ to Selected;
9: **end if**
10: **end for**
11: **end for**
12: **return** Selected

After defining an initial set of selected mappings, ALIN evaluates each of them to verify if some of them can already be automatically accepted (line 2 of Algorithm 1) using the *AcceptConceptMappings* algorithm (Algorithm 4). At first, ALIN automatically accepts concept mappings whose entity names are the same and moves them from the set of selected mappings

Algorithm 3 *SelectMappingsPerMetric*

Input: SE, SE', SimMet
Output: M

M $= \emptyset$;
for each $e \in SE$ **do**
  $bestmatch \leftarrow \underset{e' \in SE'}{\arg\max}\ simvalue(\langle e, e' \rangle, SimMet)$;
  if $e = \underset{e'' \in SE}{\arg\max}\ simvalue(\langle e'', bestmatch \rangle, SimMet)$ **then**
    add $\langle e, bestmatch \rangle$ to M;
  end if
end for
return M;

to the accepted ones (from line 1 to line 6 of Algorithm 4). After that, ALIN identifies all automatically accepted concept mappings that are in an anti-pattern with other automatically accepted concept mappings (line 7 of Algorithm 4) using the *UndoConceptAntiPatterns* algorithm (Algorithm 10) and moves them from the accepted mappings to the set of selected mappings. This use of anti-patterns is done to minimize the precision decrease that can occur because of the automatic acceptation of mappings. Subsection 4.3.3 explains anti-patterns.

Algorithm 4 *AcceptConceptMappings*

Input: Selected
Output: Accepted, Selected

1: Accepted $= \emptyset$;
2: **for each** $\langle e, e' \rangle \in$ Selected **do**
3:   **if** ename(e, e') **then**
4:     move $\langle e, e' \rangle$ from Selected to Accepted;
5:   **end if**
6: **end for**
7: Accepted,Selected $\leftarrow$ *UndoConceptAntiPatterns*(Accepted,Selected);
8: **return** Accepted, Selected

Algorithm 5 *SuspendConceptMappings*

Input: Selected,SetofSemSim
Output: Suspended,Selected

Suspended$=\emptyset$;
for each $\langle e, e' \rangle \in$ Selected **do**
  if $\forall$ SimMet $\in$ SetofSemSim, $simvalue(\langle e, e' \rangle, SimMet) \leq 0.9$ **then**
    move $\langle e, e' \rangle$ from Selected to Suspended;
  end if
end for
return Suspended,Selected

In the selection step, selected mappings less likely to be correct are suspended (line 3 of Algorithm 1), using the *SuspendConceptMappings* algorithm (Algorithm 5). The *SuspendConceptMappings* algorithm identifies all selected mappings in which the two concepts have all semantic similarity metric values under a threshold (here 0.9, see Section 5.2) and suspends them. The similarity metrics used in this technique are chosen by the user, e.g., Resnick, Jiang-Conrath or Lin. The suspended mappings can become selected again after interaction with the expert, during the propagation step (Subsection 4.3.2), by the use of the *UnsuspendConceptMappings* algorithm (Algorithm 7).

Subsection 4.3.2 describes the propagation of the accepted mappings.

4.3.2 Propagation Step

The ALIN propagation step (from line 4 to line 15 of Algorithm 1) describes the activities to choose the selected mapping to receive expert feedback, the feedback itself, and the feedback propagation.

At the beginning, ALIN propagates the automatically accepted mappings (line 5 of the Algorithm 1) in the same way as it does with the mappings accepted by the expert.

At each iteration in the propagation step (from line 7 to line 15 of the Algorithm 1), ALIN chooses among all the selected mappings, the one that has the highest sum of their similarities (line 8 of Algorithm 1). After that, this mapping receives the expert feedback (line 9 of Algorithm 1), where the expert can accept or reject the selected mapping. If the expert accepts the mapping, ALIN moves it to the set of accepted mappings (line 12 of Algorithm 1) and propagates its effects (line 13 of Algorithm 1) through the *PropagateAcceptedMapping* algorithm (Algorithm 6).

Algorithm 6 *PropagateAcceptedMapping*

Input: $\langle e, e' \rangle$,Accepted,Selected,Suspended,SetofSimMet
Output: Selected,Suspended
if *isconcmap*($\langle e, e' \rangle$) **then**
  Selected ← *SelectAttributeMappings* ($\langle e, e' \rangle$, SetofSimMet, Selected);
  Selected ← *SelectRelationshipMappings* ($\langle e, e' \rangle$, Accepted, Selected);
  Suspended,Selected ← *UnsuspendConceptMappings* ($\langle e, e' \rangle$, Suspended, Selected);
  Selected ← *AvoidConceptAntiPatterns* ($\langle e, e' \rangle$, Selected);
end if
if *isattrmap*($\langle e, e' \rangle$) **then**
  Selected ← *AvoidAttributeMultipleEntityAntiPattern* ($\langle e, e' \rangle$, Selected);
end if
if *isrelatmap*($\langle e, e' \rangle$) **then**
  Selected ← *AvoidRelationshipMultipleEntityAntiPattern* ($\langle e, e' \rangle$, Selected);
end if
Return Selected,Suspended

In the propagation step, ALIN only accepts mappings that have received positive feedback from the expert. So, if the expert does not make mistakes, the monotonic growth of the recall and the precision in the propagation step is guaranteed.

The techniques used to propagate the effects of the expert feedback are described in this subsection. The first three techniques described are structural ones that select new mappings. A structural technique takes, as input, one or more mappings, and it generates, as output, mappings structurally related to them.

The *UnsuspendConceptMappings* algorithm (Algorithm 7) implements the first structural technique. This algorithm assumes that concept mappings are more prone to be true if the superconcepts of both concepts in the mapping are in an accepted mapping. This algorithm, at each iteration, identifies all suspended mappings that are formed by subconcepts of the concepts of the accepted mappings and unsuspend them, i.e., they become again selected mappings. We implemented this technique in a prior version of ALIN: $ALIN_{Syn}$ (Da Silva et al., 2017).

So far, the ALIN approach has only taken into account concept mappings. With the second structural technique, implemented by the *SelectAttributeMappings* algorithm (Algorithm 8), attribute mappings of the accepted mappings were selected, i.e., once the expert accepts a concept mapping, ALIN selects some mappings between the attributes of its concepts.

Algorithm 7 *UnsuspendConceptMappings*

Input: $\langle c_x, c'_x \rangle$, Suspended, Selected
Output: Suspended, Selected

for each $\langle c_y, c'_y \rangle \in$ Suspended **do**
  if sub(c_y, c_x) $\wedge$ sub(c'_y, c'_x) **then**
    move $\langle c_y, c'_y \rangle$ from Suspended to Selected;
  end if
end for
return Suspended, Selected

To develop the *SelectAttributeMappings* algorithm, we have assumed that attribute mappings are more prone to be a mapping between attributes of concepts if their concepts are in an accepted mapping. To reduce the search space to the *SelectAttributeMappings* algorithm, ALIN (line 4 of the Algorithm 8) used the *SelectMappingsPerMetric* algorithm (Algorithm 3). We implemented this technique in a prior version of ALIN: $ALIN_{Attr}$ (Da Silva, Revoredo, Baião and Euzenat, 2018).

Algorithm 8 *SelectAttributeMappings*

Input: $\langle c, c' \rangle$,SetofSimMet,Selected
Output: Selected

1: SA $\leftarrow attributes(c)$;
2: SA' $\leftarrow attributes(c')$;
3: **for each** $SimMet \in SetofSimMet$ **do**
4:   M $\leftarrow$ *SelectMappingsPerMetric*(SA,SA',SimMet);
5:   **for each** $\langle a, a' \rangle \in$ M **do**
6:     **if** $\langle a, a' \rangle \notin$ Selected **then**
7:       add $\langle a, a' \rangle$ to Selected;
8:     **end if**
9:   **end for**
10: **end for**
11: **return** Selected

To develop the third structural technique, implemented by the *SelectRelationshipMappings* algorithm (Algorithm 9), we have assumed that relationships are more prone to be in a mapping if they are relationships between the concepts of two accepted concept mappings. We have also assumed that relationships between subconcepts of these concepts are more prone to be in a mapping too. These situations in which the *SelectRelationshipMappings* algorithm selects relationship mappings are called triggers for the selection of relationship mappings (Definition 4.1).

Algorithm 9 *SelectRelationshipMappings*

Input: $\langle c_x, c'_x \rangle$, Accepted, Selected
Output: Selected

for each $\langle c_y, c'_y \rangle \in$ Accepted such that $\langle c_y, c'_y \rangle \neq \langle c_x, c'_x \rangle$ **do**
  if trigger($\langle c_x, c'_x \rangle$,$\langle c_y, c'_y \rangle$,$\langle r, r' \rangle$) **then**
    add $\langle r, r' \rangle$ to Selected;
  end if
end for
return Selected

Definition 4.1 (Triggers) *Given $\langle c_1, c'_1\rangle$ and $\langle c_2, c'_2\rangle$, the following conditions:*

- *rconcept(r,c_1,c_2) $\wedge$ rconcept(r',c'_1,c'_2)*
- *sub(c_3,c_1,2) $\wedge$ rconcept(r,c_3,c_1) $\wedge$ rconcept(r',c'_1,c'_2)*
- *sub(c'_3,c'_1,2) $\wedge$ sub(c_3,c_1,2) $\wedge$ rconcept(r',c'_3,c'_1) $\wedge$ rconcept(r,c_3,c_1)*

are called triggers between $\langle c_1, c'_1\rangle$ and $\langle c_2, c'_2\rangle$ for the selection of the relationship mapping $\langle r, r'\rangle$.

4.3.3 Mapping Anti-patterns

An ontology may have construction constraints, such as a concept cannot be equivalent to its superconcept. An alignment may have other constraints like, for example, an entity of ontology O cannot be equivalent to two entities of the ontology O'. A mapping anti-pattern is a combination of mappings that generates a problematic alignment, i.e., a logical inconsistency or a violated constraint.

ALIN uses three mapping anti-patterns empirically identified by Guedes (Guedes et al., 2014*b*) (Definitions 4.2 to 4.4), extracted from the results of ontology matching tools evaluated by OAEI (Guedes et al., 2014*a*).

The multiple-entity anti-pattern (Definition 4.2) applies when a single entity does not participate in two mappings (Figure 3 (a)). The cross mapping anti-pattern (Definition 4.3) applies when no subconcept can be equivalent to its superconcept. This mapping anti-pattern applies when a concept c_1 is a subconcept of the concept c_2 and c'_1 is subconcept of the concept c'_2 and c_1 is in a mapping with c'_2 and c_2 is in a mapping with th c'_1 (see Figure 3 (b)). The disjunction and generalization anti-pattern (Definition 4.4) applies when a pair of concepts of an ontology that are subconcept and superconcept are in mappings with two disjoint concepts of the other ontology (Figure 3 (c)).

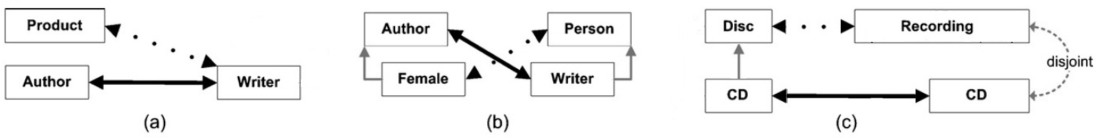

Figure 3 The three mapping anti-pattern used in ALIN. (a) multiple-entity anti-pattern. (b) cross mapping anti-pattern. (c) disjunction and generalization anti-pattern. Based on (Jean-Mary et al., 2009)

Definition 4.2 (Multiple-entity anti-pattern) *If the entities e_1, e'_1 and e'_2 of mappings $\langle e_1, e'_1\rangle$ and $\langle e_1, e'_2\rangle$ occur in the situation described by the formula $entity(e_1, O) \wedge entity(e'_1, O') \wedge entity(e'_2, O')$, then $\langle e_1, e'_1\rangle$ and $\langle e_1, e'_2\rangle$ are said to be in a multiple-entity anti-pattern.*

Definition 4.3 (Cross mapping anti-pattern) *If the concepts c_1, c_2, c'_1 and c'_2 of mappings $\langle c_2, c'_1\rangle$ and $\langle c_1, c'_2\rangle$ occur in the situation described by the formula $concept(c_1, O) \wedge concept(c'_1, O') \wedge concept(c_2, O) \wedge concept(c'_2, O') \wedge sub(c_1, c_2) \wedge sub(c'_1, c'_2)$, then $\langle c_2, c'_1\rangle$ and $\langle c_1, c'_2\rangle$ are said to be in a cross mapping anti-pattern.*

Definition 4.4 (Disjunction and Generalization anti-pattern) *If the concepts c_1, c_2, c'_1 and c'_2 of mappings $\langle c_1, c'_1\rangle$ and $\langle c_2, c'_2\rangle$ occur in the situation described by the formula $concept(c_1, O) \wedge concept(c'_1, O') \wedge concept(c_2, O) \wedge concept(c'_2, O') \wedge sub(c_1, c_2) \wedge dis(c'_1, c'_2)$, then $\langle c_1, c'_1\rangle$ and $\langle c_2, c'_2\rangle$ are said to be in a disjunction and generalization mapping anti-pattern.*

ALIN takes advantage of the fact that if a correct mapping is in a mapping anti-pattern with another mapping, the other mapping is wrong. It uses (line 7 of Algorithm 4) the *UndoConceptAntiPatterns* algorithm (Algorithm 10) to identify automatically accepted concept mappings that are in mapping anti-patterns with others automatically accepted concept mappings and moves these mappings from the set of accepted mappings to the set of selected mappings. ALIN (line 5 of Algorithm 6) uses the *AvoidConceptAntiPatterns* algorithm (Algorithm 11) to identify those selected concept mappings that are in a mapping anti-patterns with an accepted concept mapping and removes them from the set of selected mappings. ALIN (lines 8 and 11 of Algorithm 6) uses the *AvoidAttributeMultipleEntityAntiPattern* algorithm (Algorithm 12) and the *AvoidRelationshipMultipleEntityAntiPattern* algorithm (Algorithm 13) to identify selected attribute and relationship mappings that are in the multiple-entity anti-pattern with an accepted mapping and removes them from the set of selected mappings.

Algorithm 10 *UndoConceptAntiPatterns*

Input: Accepted, Selected
Output: Accepted, Selected

for each $\langle c_x, c'_x \rangle \in$ Accepted **do**
  for each $\langle c_y, c'_y \rangle \in$ Accepted **do**
    if $antipat(\langle c_x, c'_x \rangle, \langle c_y, c'_y \rangle)$ **then**
      move $\langle c_x, c'_x \rangle$ from Accepted to Selected;
      move $\langle c_y, c'_y \rangle$ from Accepted to Selected;
    end if
  end for
end for
return Accepted, Selected

Algorithm 11 *AvoidConceptAntiPatterns*

Input: $\langle c_x, c'_x \rangle$, Selected
Output: Selected

for each $\langle c_y, c'_y \rangle \in$ Selected **do**
  if $antipat(\langle c_x, c'_x \rangle, \langle c_y, c'_y \rangle)$ **then**
    remove $\langle c_y, c'_y \rangle$ from Selected;
  end if
end for
return Selected

Algorithm 12 *AvoidAttributeMultipleEntityAntiPattern*

Input: $\langle a_x, a'_x \rangle$, Selected
Output: Selected

for each $\langle a_y, a'_y \rangle \in$ Selected **do**
  if $multent(\langle a_x, a'_x \rangle, \langle a_y, a'_y \rangle)$ **then**
    remove $\langle a_y, a'_y \rangle$ from Selected;
  end if
end for
return Selected

5 Evaluation overview

ALIN is implemented in Java using the following APIs: Stanford CoreNLP API (Manning et al., 2014) to put a word in canonical form; Simmetrics API (Surhone et al., 2010), for string-based similarity metrics; HESML API (Lastra-Díaz et al., 2017), for Wordnet (Fellbaum, 1998) based linguistic metrics; the OWL API for handling ontologies written in OWL, and the Alignment API (David et al., 2011) to deal with alignments.

Algorithm 13 *AvoidRelationshipMultipleEntityAntiPattern*

Input: $\langle r_x, r'_x \rangle$, Selected
Output: Selected
for each $\langle r_y, r'_y \rangle \in$ Selected **do**
  if $multent(\langle r_x, r'_x \rangle, \langle r_y, r'_y \rangle)$ **then**
    remove $\langle r_y, r'_y \rangle$ from Selected;
  end if
end for
return Selected

In the executions described in this section, ALIN receives as input parameters six similarity metrics. We based the process of choosing the similarity metrics used by ALIN on the result of these metrics in assessments (Petrakis et al., 2006; Cheatham and Hitzler, 2013). We chose Jaccard, Jaro-Wrinkler, and n-gram lexical metrics and the Resnick, Jiang-Conrath and Lin semantic metrics. Resnick, Jiang-Conrath, and Lin are metrics that require a taxonomy to be computed, this taxonomy is provided by Wordnet (Fellbaum, 1998).

The Ontology Alignment Evaluation Initiative (OAEI) is a coordinated international initiative, which organizes the evaluation of ontology matching systems. Its main goal is to compare systems and algorithms openly and on the same basis, to allow anyone to draw conclusions about the best matching strategies. OAEI provides ontologies for various domains (data sets). The ontologies of data sets can have three types of entities: concepts (classes), data properties (attributes), and object properties (relationships). OAEI provides reference alignments, which are alignments that contains the mappings that are believed to be correct, for each ontology pair in a data set.

To evaluate ALIN, we followed the same protocol as the OAEI 2018 interactive matching track (Algergawy et al., 2018) in which ALIN participated (Da Silva, Revoredo and Baião, 2018). An interactive matcher is run and the reference alignment is used to simulate the expert answering. At each interaction, up to three selected mappings can be submitted to the expert, as long as each selected mapping has one entity in common with another selected mapping in the interaction (Faria, 2016). The quality of an alignment generated by a matching approach is generally measured by the F-measure, which is the harmonic mean between recall and precision. When the ontology matching process is interactive another quality metric occurs, it is the number of interactions with the expert. In Tables 4 to 11 hereafter, 'Total Requests' is the number of interactions with the expert.

The OAEI 2018 interactive matching track (Algergawy et al., 2018) is based on two data sets: Conference and Anatomy.

The Conference data set consists of 7 ontologies, resulting in a total of 21 ontology pairs. There are 125,860 possible mappings when we take into account only the mappings of the same entity type among all 21 ontology pairs. ALIN takes into account only the mappings of the same entity type. Among the 21 ontology pairs of the Conference data set there are 305 mappings in the reference alignments.

The Anatomy data set that contains two ontologies, one with the mouse anatomy and another with the human anatomy. The number of possible mappings between the two ontologies is 9,066,176. There are 1,516 mappings in the reference alignment.

5.1 Evaluation Criteria

We measure efficiency in the use of expert feedback as the ratio between the F-measure gain over the baseline ($ALIN_{NoRel}$) and the additional number of expert answers.

We raise the following research questions to evaluate if the techniques used in ALIN help to achieve the objective to generate a better alignment:

RQ1: Does the *SelectRelationshipMappings* algorithm increases the F-measure of the generated alignment?

RQ2: Is the expert feedback more efficiently used when it assists the *SelectRelationshipMappings* algorithm?

RQ3: Does the interactive use of mapping anti-patterns decrease the number of interactions with the expert?

RQ4: Does the use of the techniques described in this paper generates a final alignment with quality and number of interactions compatible with the state-of-the-art proposals?

Table 3 Alin versions

ALin version	Use the *SelectRelationshipMappings* algorithm	Use the expert feedback to assist the *SelectRelationshipMappings* algorithm	Use anti-patterns
$ALIN_{NoRel}$	No	-	Yes
$ALIN_{Aut}$	Yes	No	Yes
$ALIN_{NoAP}$	Yes	Yes	No
ALIN	Yes	Yes	Yes

In order to evaluate the benefit of the proposed solutions, we developed several versions of ALIN:

ALIN is the full version of ALIN. It uses only the accepted mappings as input of the *SelectRelationshipMappings* algorithm, i.e., it uses the expert feedback to assist the *SelectRelationshipMappings* algorithm.

$ALIN_{NoAP}$ is the full ALIN without using anti-patterns. Comparing its results with the results of full ALIN allows answering to RQ3.

$ALIN_{Aut}$ uses all the automatically selected concept mappings as input to the *SelectRelationshipMappings* algorithm, i.e., the line 3 of the *PropagateAcceptedMapping* algorithm (Algorithm 6) is removed and the *AutomaticallySelectRelationshipMappings* algorithm (Algorithm 14) is called between the lines 1 and 2 of Algorithm 1. Since it uses the automatically selected concept mappings as input, it does not use the expert feedback to the *SelectRelationshipMappings* algorithm. Comparing the results of its execution with the results of $ALIN_{NoRel}$ allows answering to RQ1. Comparing the results of its execution with the the results of $ALIN_{NoRel}$ and the results of full Alin allows answering to RQ2.

$ALIN_{NoRel}$ does not use the *SelectRelationshipMappings* algorithm, i.e., line 3 and lines from 10 to 12 are removed from the *PropagateAcceptedMapping* algorithm (Algorithm 6). Comparing the results of its execution with the results of $ALIN_{Aut}$ allows answering to RQ1. Comparing the results of its execution with the the results of $ALIN_{Aut}$ and the results of full Alin allows answering to RQ2.

Table 3 shows the characteristics of ALIN versions.

Algorithm 14 *AutomaticallySelectRelationshipMappings*

Input: Selected
Output: Selected

for each $\langle c, c' \rangle \in$ Selected **do**
    Selected $\leftarrow$ *SelectRelationshipMappings* ($\langle c, c' \rangle$,Selected,Selected);
end for
Return Selected

5.2 Parameter Tunning

In the *SelectRelationshipMappings* algorithm, we searched for the relationship mappings of the subconcepts of the concepts of two accepted concept mappings. During the creation of this algorithm, we found out that if we fixed a maximum depth of the subconcepts that we searched, the number of relationship mappings that the algorithm would select decreased. So the number of interactions with the expert. The depth of 2 was where we found the maximum recall, and we set this value for ALIN. From this value, we found only relationships mappings that were not accepted by the expert (Table 4).

Table 4 Comparison between ALIN executions with different maximum depths of the subconcepts using Conference data set

Maximum Depth of the Subconcepts	Total Requests	Precision	F-measure	Recall
With no Relationships Mappings	246	0.918	0.793	0.698
0	251	0.919	0.798	0.705
1	254	0.919	0.802	0.711
2	276	0.921	0.809	0.721
3	295	0.921	0.809	0.721
4	310	0.921	0.809	0.721
5	321	0.921	0.809	0.721

We modified the *SuspendConceptMapping* algorithm in ALIN. We removed, in $ALIN_{Syn}$ (Da Silva et al., 2017), all mappings from the set of selected mappings in which the two mapping concepts were not semantically equivalent, i.e., if all of the semantic similarity metrics were less than 1. In ALIN, using the Conference data set, we did several executions changing the minimum value of similarity, and we found that the value of 0.9 reached the best result, so in ALIN, we fixed the value in 0.9 (Table 5). Although we have found the value by running ALIN with the Conference data set, we used the same value with the Anatomy data set.

Since the parameter values were empirically set, further study would be required to make them more general.

Table 5 Comparison between ALIN executions with different minimum semantic similarity values of the entities using the Conference data set

Minimum Semantic Similarity between the Entities	Total Requests	Precision	F-measure	Recall
1.0	257	0.917	0.789	0.692
0.9	276	0.921	0.809	0.721
0.8	300	0.921	0.809	0.721
0.7	310	0.920	0.805	0.715

5.3 Analysis of the results

We made executions with the three versions of ALIN for both the Conference data set and the Anatomy data set. The execution of $ALIN_{Aut}$, for the Conference data set, using the *SelectRelationshipMappings* algorithm generates an alignment with higher F-measure relatively to $ALIN_{NoRel}$. Thus, this execution responds positively to RQ1.

With all the selected mappings as input to the *SelectRelationshipMappings* algorithm in $ALIN_{Aut}$, there is an increase of 1.02% in F-measure, but with 29.26% more interactions

Table 6 Comparison between different ontology matching executions with Conference data set

	Total Req	Prec	F	Rec	Increase in Total Requests relative to $ALIN_{NoRel}$	Increase in F-measure relative to $ALIN_{NoRel}$	Efficiency relative to $ALIN_{NoRel}$
$ALIN_{NoRel}$	246	0.918	0.793	0.698	-	-	-
$ALIN_{Aut}$	318	0.921	0.811	0.725	29.26	1.02	0.011
ALIN	276	0.921	0.809	0.721	12.19	1.02	0.027

Table 7 Comparison between different ontology matching executions with Anatomy data set

	Total Req	Prec	F	Rec	Increase in Total Requests relative to $ALIN_{NoRel}$	Increase in F-measure relative to $ALIN_{NoRel}$	Efficiency relative to $ALIN_{NoRel}$
$ALIN_{NoRel}$	602	0.994	0.902	0.826	-	-	-
$ALIN_{Aut}$	602	0.994	0.902	0.826	-	-	-
ALIN	602	0.994	0.902	0.826	-	-	-

with the expert relative to $ALIN_{NoRel}$. With the use of accepted mappings as input to the *SelectRelationshipMappings* algorithm in Full ALIN, we got almost the same increase in F-measure, but with only 12.19% more interactions relative to $ALIN_{NoRel}$. So, concerning $ALIN_{NoRel}$, the Full ALIN allows more efficiency in the use of the expert feedback than $ALIN_{Aut}$. Thus, this execution responds positively to RQ2.

With the use of the Anatomy data set, all versions achieved the same alignment, and thus the same F-measure (Table 7). This is because there are no relationships in the ontologies of the Anatomy data set, hence the *SelectRelationshipMappings* algorithm brings no benefit.

Thus, the executions show that we can only respond positively to both RQ1 and RQ2 if that involved ontologies have relationships.

The interactive use of mapping anti-patterns by the ALIN approach decreases the number of interactions with the expert without decreasing the F-measure (Tables 8 and 9), which respond positively to the RQ3.

Table 8 Comparison between ALIN executions with and without mapping anti-patterns using the Conference data set

	Total Requests	Precision	F-measure	Recall
$ALIN_{NoAP}$	329	0.921	0.809	0.721
ALIN	276	0.921	0.809	0.721

Table 9 Comparison between ALIN executions with and without mapping anti-patterns using the Anatomy data set

	Total Requests	Precision	F-measure	Recall
$ALIN_{NoAP}$	679	0.993	0.903	0.828
ALIN	602	0.994	0.902	0.826

Table 10 Comparison between OAEI 2018 interactive matching track tools using Conference data set with 100% hit rate (Algergawy et al., 2018)

	Total Requests	Precision	F-measure	Recall
ALIN	276	**0.921**	**0.809**	**0.721**
AML	270	0.912	0.799	0.711
LogMap	82	0.886	0.723	0.61
XMap	**16**	0.719	0.666	0.62

Table 11 Comparison between OAEI 2018 interactive matching track tools using Anatomy data set with 100% hit rate (Algergawy et al., 2018)

	Total Requests	Precision	F-measure	Recall
ALIN	602	**0.994**	0.902	0.826
AML	240	0.964	**0.956**	**0.948**
LogMap	388	0.982	0.909	0.846
XMap	**35**	0.929	0.897	0.867

5.4 Comparison between tools that participated in the OAEI 2018 interactive matching track

ALIN participated in the OAEI 2018 interactive matching track. OAEI provides a comparison between tool performance in the interactive matching track each year, and it uses the Conference and the Anatomy data sets (Algergawy et al., 2018), as we can see in the tables 10 and 11.

The tools AML, LogMap, and XMAP (Tables 10 and 11) are interactive ontology matching tools which select attribute and relationship mappings, but in a non-interactive way, not taking into account the expert feedback.

In relation of the F-measure, the results show that ALIN generated a high-level result when running the Conference data set (Table 10) and ALIN got a result close to other tools when running the Anatomy data set (Table 11) when the expert hit 100% of the answers, which answers positively to RQ4.

In relation to number of interactions with the expert, ALIN, when running with the Conference data set, has a number of interactions more compatible with the other tools than with the Anatomy data set. That is because the ontologies of the Conference dataset, contrary to those of Anatomy, contain attributes and relationships, which allows the use of the *SelectRelationshipMappings* and the *SelectAttributeMappings* algorithms to improve the set of selected mappings.

When the expert make mistakes, ALIN suffers a sharp fall, relative to other tools (Figures 4 and 5), of the F-measure. This fall is because ALIN uses expert feedback as input to the used techniques, which causes these techniques to generate erroneous results when the expert fails. Besides, these outputs of the techniques serve to modify the set of selected mappings, which causes the set of selected mappings to get worse, that is, correct mappings can come out, and wrong mappings can enter.

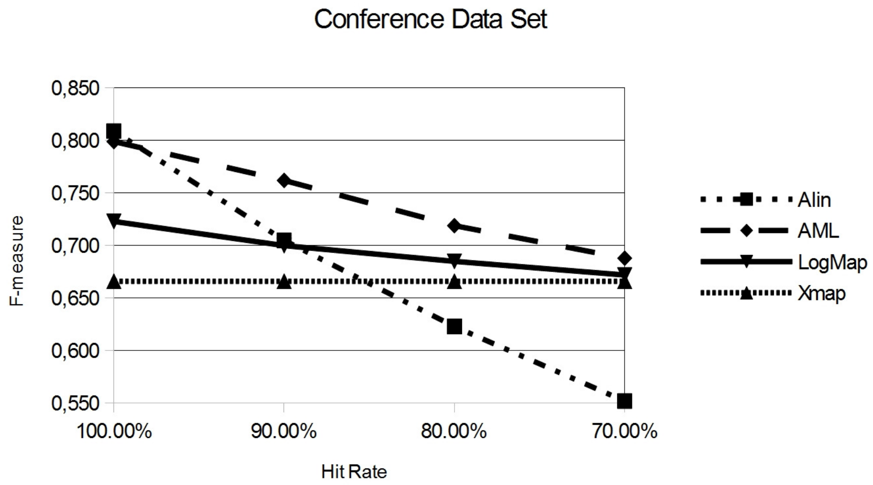

Figure 4 Comparison of execution of several interactive tools with different hit rates – Conference data set

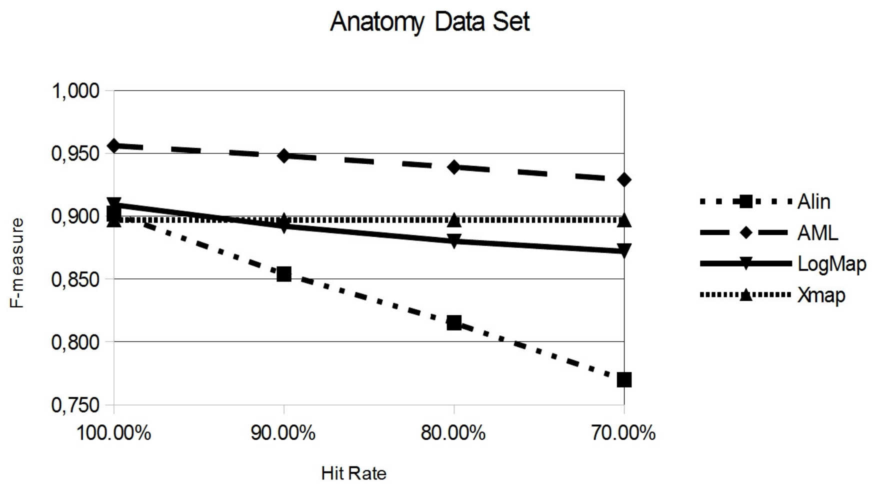

Figure 5 Comparison of execution of several interactive tools with different hit rates – Anatomy data set

6 Conclusions and Future Works

Ontology matching plays an important role in the integration of heterogeneous data sources that are described by ontologies. Its purpose is to discover mappings between the entities of at least two ontologies.

ALIN contributes to interactive ontology matching techniques through the use of expert feedback for the improvement of the set of selected mappings. It does this by using expert feedback to select relationship mappings and to reject selected mappings through the use of anti-patterns.

We evaluated the various new features of the ALIN approach by comparing different versions of the system. The evaluation performed showed that the use of the expert feedback to improve the set of selected mappings increased the F-measure gain per interaction.

ALIN participated in the interactive ontology matching track of OAEI 2018. The results showed that ALIN generates an alignment with good quality in comparison to other tools, in precision, recall and F-measure when the expert never makes mistakes. Hence, in spite of good quality results, the proposed technique is no silver bullet for ontology matching.

The Alin approach had good results when the expert does not make mistake, but as the approach uses expert feedback to select and reject mappings, expert mistakes generate noise for the structural techniques and for the use of mapping anti-patterns. As future work, one interesting direction is to explore how to reduce the negative effects of expert mistakes. In this case, the question of using several experts simultaneously can occur and require specific treatments.

A more informative evaluation to assess the benefits of using the expert feedback to assist the selection of mappings would consist of applying it to the mappings generated by other systems participating in OAEI.

Acknowledgments

Jérôme Euzenat was partially funded by a CNPq Special Visiting Researcher grant (314782/2014-1).

References

Algergawy, A., Cheatham, M., Faria, D., Ferrara, A., Fundulaki, I., Harrow, I., Hertling, S., Jiménez-Ruiz, E., Karam, N., Khiat, A., Lambrix, P., Li, H., Montanelli, S., Paulheim, H., Pesquita, C., Saveta, T., Schmidt, D., Shvaiko, P., Splendiani, A., Thiéblin, E., Trojahn, C., Vataščinová, J., Zamazal, O. and Zhou, L. (2018), 'Results of the ontology alignment evaluation initiative 2018', *CEUR Workshop Proceedings* **2288**, 76–116.
`http://www.dit.unitn.it/~pavel/om2018/papers/oaei18_paper0.pdf`

Balasubramani, B., Taheri, A. and Cruz, I. (2015), User involvement in ontology matching using an online active learning approach, *in* 'CEUR Workshop Proceedings', Vol. 1545, CEUR-WS, pp. 45–49.

Belkin, N. J., Bierig, R., Buscher, G., Jose, J. and Teevan, J. (2010), 'Understanding the user - logging and interpreting user interactions in information search and retrieval', *Information Retrieval* (October 2015), 2009–2010.

Cheatham, M. and Hitzler, P. (2013), String similarity metrics for ontology alignment, *in* 'Proceedings of the 12th International Semantic Web Conference - Part II', ISWC '13, Springer-Verlag New York, Inc., New York, NY, USA, pp. 294–309.

Chunhua, L., Zhiming, C., Pengpeng, Z., Jian, W., Jie, X. and Tianxu, H. (2015), Improving ontology matching with propagation strategy and user feedback, *in* 'Seventh International Conference on Digital Image Processing (ICDIP 2015)', Vol. 9631, p. 6.
`https://doi.org/10.1117/12.2197167`

Cruz, I. F., Loprete, F., Palmonari, M., Stroe, C. and Taheri, A. (2014), Pay-as-you-go multi-user feedback model for ontology matching, *in* K. Janowicz, S. Schlobach, P. Lambrix and E. Hyvönen, eds, 'Knowledge Engineering and Knowledge Management', Springer International Publishing, Cham, pp. 80–96.

Cruz, I. F., Stroe, C. and Palmonari, M. (2012), Interactive user feedback in ontology matching using signature vectors, *in* '28th IEEE International Conference on Data Engineering', pp. 1321–1324.

Da Silva, J., Revoredo, K. and Baião, F. A. (2018), ALIN results for OAEI 2018.
`http://www.dit.unitn.it/~pavel/om2018/papers/oaei18_paper1.pdf`

Da Silva, J., Revoredo, K., Baião, F. A. and Euzenat, J. (2017), Semantic interactive ontology matching: Synergistic combination of techniques to improve the set of candidate correspondences, *in* 'OM 2017 - 12th ISWC workshop on ontology matching', Vol. 2032, pp. 13–24.
`http://disi.unitn.it/~pavel/om2017/papers/om2017_Tpaper2.pdf`

Da Silva, J., Revoredo, K., Baião, F. A. and Euzenat, J. (2018), Interactive Ontology Matching: Using Expert Feedback to Select Attribute Mappings.
`http://disi.unitn.it/~pavel/om2018/papers/om2018_LTpaper3.pdf`

David, J., Euzenat, J., Scharffe, F. and Trojahn dos Santos, C. (2011), 'The Alignment API 4.0', *Semant. web* **2**(1), 3–10.
`http://dl.acm.org/citation.cfm?id=2019470.2019474`

Duan, S., Fokoue, A. and Srinivas, K. (2010), One size does not fit all: Customizing ontology alignment using user feedback, *in* P. F. Patel-Schneider, Y. Pan, P. Hitzler, P. Mika, L. Zhang, J. Z. Pan, I. Horrocks and B. Glimm, eds, 'The Semantic Web – ISWC 2010', Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 177–192.

Euzenat, J. and Shvaiko, P. (2013), *Ontology Matching - Second Edition*, Springer-Verlag.

Faria, D. (2016), 'Using the SEALS Client's Oracle in Interactive Matching'.
`https://github.com/DanFaria/OAEI{_}SealsClient/blob/master/OracleTutorial.pdf`

Faria, D., Pesquita, C., Santos, E., Palmonari, M., Cruz, I. F. and Couto, F. M. (2013), The AgreementMakerLight ontology matching system, *in* R. Meersman, H. Panetto, T. Dillon, J. Eder, Z. Bellahsene, N. Ritter, P. De Leenheer and D. Dou, eds, 'On the Move to Meaningful Internet Systems: OTM 2013 Conferences', Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 527–541.

Fellbaum, C., ed. (1998), *WordNet: An electronic lexical database*, MIT Press.

Gale, D. and Shapley, L. S. (1962), 'College admissions and the stability of marriage', *The American Mathematical Monthly* **69**(1), 9–15.
`https://doi.org/10.1080/00029890.1962.11989827`

Guedes, A., Baião, F. and Revoredo, K. (2014*a*), Digging Ontology Correspondence Antipatterns, *in* 'Proceedings of the 5th Workshop on Ontology and Semantic Web Patterns (WOP2014) co-located with the 13th International Semantic Web Conference (ISWC 2014)', Vol. 1032, pp. 38—-48.

Guedes, A., Baião, F. and Revoredo, K. (2014*b*), On the Identification and Representation of Ontology Correspondence Antipatterns, *in* 'Proceedings of the 8th International Workshop on Modular Ontologies co-located with the 8th International Conference on Formal Ontology in Information Systems (FOIS 2014), CEUR Workshop Proceedings', Vol. 1248.

Hertling, S. (2012), Hertuda results for OAEI 2012, *in* 'Proceedings of the 7th International Conference on Ontology Matching - Volume 946', OM'12, CEUR-WS.org, Aachen, Germany, Germany, pp. 141–144.
`http://dl.acm.org/citation.cfm?id=2887596.2887607`

Irving, R. W., Manlove, D. F. and O'Malley, G. (2009), 'Stable marriage with ties and bounded length preference lists', *Journal of Discrete Algorithms* **7**(2), 213 – 219. Selected papers from the 2nd Algorithms and Complexity in Durham Workshop ACiD 2006.
`http://www.sciencedirect.com/science/article/pii/S1570866708000683`

Jean-Mary, Y. R., Shironoshita, E. P. and Kabuka, M. R. (2009), Ontology matching with semantic verification.

Jiménez-Ruiz, E. and Grau, B. C. (2011), LogMap: Logic-based and scalable ontology matching, *in* L. Aroyo, C. Welty, H. Alani, J. Taylor, A. Bernstein, L. Kagal, N. Noy and E. Blomqvist, eds, 'The Semantic Web – ISWC 2011', Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 273–288.

Jiménez-Ruiz, E., Grau, B. C., Zhou, Y. and Horrocks, I. (2012), Large-scale interactive ontology matching: Algorithms and implementation, *in* 'ECAI 2012 - 20th European Conference on Artificial Intelligence', Vol. 242, pp. 444–449.

Lambrix, P. and Kaliyaperumal, R. (2016), 'A Session-based Ontology Alignment Approach enabling User Involvement', *Semantic Web* **1**, 1–28.

Lastra-Díaz, J. J., García-Serrano, A., Batet, M., Fernández, M. and Chirigati, F. (2017), 'HESML: A scalable ontology-based semantic similarity measures library with a set of reproducible experiments and a replication dataset', *Inf. Syst.* **66**(C), 97–118.
`https://doi.org/10.1016/j.is.2017.02.002`

Lopes, V., Baião, F. and Revoredo, K. (2015), Alinhamento Interativo de Ontologias Uma Abordagem Baseada em Query-by-Committee, Master's thesis, UNIRIO.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J. and McClosky, D. (2014), The Stanford CoreNLP natural language processing toolkit, *in* 'Association for Computational Linguistics (ACL) System Demonstrations', pp. 55–60.
`http://www.aclweb.org/anthology/P/P14/P14-5010`

Ngo, D., Bellahsene, Z. and Todorov, K. (2013), Opening the black box of ontology matching, *in* P. Cimiano, O. Corcho, V. Presutti, L. Hollink and S. Rudolph, eds, 'The Semantic Web: Semantics and Big Data', Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 16–30.

Paulheim, H. and Hertling, S. (2013), WeSeE-match results for OAEI 2013, *in* 'Proceedings of the 8th International Conference on Ontology Matching - Volume 1111', OM'13, CEUR-WS.org, Aachen, Germany, Germany, pp. 197–202.
`http://dl.acm.org/citation.cfm?id=2874493.2874513`

Paulheim, H., Hertling, S. and Ritze, D. (2013), 'Towards Evaluating Interactive Ontology Matching Tools', *Lecture Notes in Computer Science* **7882**, 31–45.

Petrakis, E. G. M., Varelas, G., Hliaoutakis, A. and Raftopoulou, P. (2006), 'Design and Evaluation of Semantic Similarity Measures for Concepts Stemming from the Same or Different Ontologies object instrumentality', *Proceedings of the 4th Workshop on Multimedia Semantics (WMS)* **4**, 233–237.

Shi, F., Li, J., Tang, J., Xie, G. and Li, H. (2009), Actively learning ontology matching via user interaction, *in* A. Bernstein, D. R. Karger, T. Heath, L. Feigenbaum, D. Maynard, E. Motta and K. Thirunarayan, eds, 'The Semantic Web - ISWC 2009', Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 585–600.

Surhone, L. M., Timpledon, M. T. and Marseken, S. F. (2010), *SimMetrics*, VDM Publishing.

To H., Ichise, R. and Le, H. (2009), An Adaptive Machine Learning Framework with User Interaction for Ontology Matching, *in* 'Proc. Twenty-first International Joint Conference on Artificial Intelligence'.