Ontology changes-driven semantic refinement of cross-language biomedical ontology alignments

Juliana Medeiros Destro[1], Julio Cesar dos Reis[1], Ricardo da Silva Torres[3], and Ivan Ricarte[2]

[1] Institute of Computing, University of Campinas (UNICAMP) – Brazil
{juliana.destro,jreis,}@ic.unicamp.br
[2] School of Technology, University of Campinas (UNICAMP) – Brazil
ricarte@unicamp.br
[3] Department of ICT and Natural Sciences, NTNU - Norwegian University of Science and Technology, Ålesund – Norway
ricardo.torres@ntnu.no

Abstract. Biomedical computational systems benefits from the use of ontologies. However, interconnectivity between these systems is a challenge, specially when the ontologies supporting each system are described in different natural languages. Ontology alignment plays a key role in data exchange. Existing ontology matching approaches usually provide only equivalent type of relation in the generated mappings. In this article, we propose a refinement technique to enable the update of the semantic type of the mapping beyond equivalence. Our approach relies on information from the ontology evolution. Our evaluation considered LOINC releases in different languages. The results demonstrate the usefulness of ontology evolution changes to support the process of mapping refinement.

Keywords: mapping refinement · ontology evolution · cross-language alignment.

1 Introduction

Advancements in biomedical research require relying upon vast arrays of voluminous, dynamic, heterogeneous and complex datasets, resulting in difficulties to use and reuse available data. This generates an ever greater demand for adequate computer-supported methods for automatically locating, accessing, sharing, analyzing and meaningfully integrating data. Biomedical information systems have intensively relied on semantic technologies such as ontologies to turn the semantic of information explicit for machines.

The number of ontologies created in different languages grows as their use increases [15]. Usually created by different authors and for different purposes, the heterogeneity of ontologies poses a challenge for system connectivity. Data exchange relies on finding correspondences, or mappings between concepts, a process called ontology matching. Cross-language ontology mappings (*i.e.*, ontology mappings generated between ontologies described in different natural languages) are crucial for enabling interconnectivity in multiple biomedical systems.

The semantic relations identified during the matching process can be expanded through mapping refinement. We differentiate *relation* from *relationship*, where the former represents a mapping, and the latter represents concept connections in an ontology. Refinement can modify or enrich semantic relations. For instance, during the refinement process, an equivalence ($\equiv$) relation (*i.e.*, a relation defining that two interrelated concepts are equivalent) can be modified to an *is-a* ($\sqsubseteq$) (*i.e.*, representing a relation in which one concept is a specialization of the other) [2].

Matching approaches are usually approximative and identify mappings based on relatedness between concepts. The challenges of mapping refinement are due to the difficulties in establishing semantic relations between concepts, beyond the relatedness identified by the traditional matching procedures. In this context, enriched semantic correspondences in ontology mapping might boost ontology merging [12].

Ontologies are constantly evolving, by adding and removing concepts and relationships over time. These changes indicate how concepts and their relationships with each other evolved. In this article, we investigate whether ontology changes are a valuable source of information to enhance the correspondences found between concepts beyond equivalences based on the type of semantic relation. We use the ontology change operation affecting ontology entities to refine the semantic relation in already established mappings. We believe that the use of this information might provide an understanding of how concepts were updated over time to support the decision and application of actions required to modify the type of semantic relation in cross-language mappings.

In this investigation, we define mapping refinement actions and explore them in refinement procedures based on categories of ontology changes. We propose and formalize mapping refinement for addition changes and for revision changes.

We conduct an evaluation to assess our proposal concerning releases of the *Logical Observation Identifiers Names and Codes* (*LOINC*) and alignments between English and Spanish language. We applied the defined refinement procedures based on the ontology changes computed from version to the other of *LOINC*. Our study reveals a promising approach on the use of ontology evolution changes to enhance semantic relations in mappings.

The remainder of this article is organized as follows: Section 2 presents related work. Afterwards, Section 3 presents a set of formal definitions including the research problem. Section 4 reports on our proposal for refinement of cross-lingual mappings. Section 5 presents the experimental evaluation conducted with biomedical ontologies and their alignments. Section 6 discusses our findings and lessons learned. Finally, Section 7 wraps up the article and points out future research.

2 Background

Several approaches were developed to address the ontology mapping problem. For instance, Trojahn *et al.* [15] presented an extensive survey on matching systems and techniques for accomplishing multilingual and cross-lingual ontology matching. Ontology matching techniques have considered the use of similarity methods relying on background knowledge. Similarity measures aim to calculate the degree of relatedness between concepts exploiting different knowledge sources (*e.g.*, ontologies, thesauri, and domain corpora).

Aleksovski *et al.* [1] proposed the use of external knowledge sources to align ontologies. They explored paths between the anchored matched concepts to find mapping between concepts. Mapping refinement relies on the existence of a previously calculated ontology mappings. In this context, *TaxoMap* [8] refers to an approach that brought together mapping and refinement by using *WordNet* lexical database as background knowledge and explored pattern-based refinement techniques. TaxoMap uses manually created patterns to refine mappings in the same domain. In contrast, Spiliopoulos *et al.* [13] presented the *Classification-Based Learning of Subsumption Relations* method for ontology alignment. This automated method relies on the exploration of patterns that describe the relation between concepts (*e.g.*, siblings at the same hierarchy level or attributes with same content). These patterns are identified by applying a classification task using machine learning methods.

The main approaches available in literature for refinement are based on external resources or manual pattern definition. The work conducted by Arnold & Rahm *et al.* [2] defined a mapping refinement technique by using a set of equivalent mappings as input. They explored generic external resources and proposed a two-step enrichment technique to improve existing imprecise mappings. They used linguistic techniques and resources like *WordNet* to refine semantic relations between aligned concepts. Their work aimed to transform equivalence between concepts into an *is-a* or *part-of* relation, which may further reflect the real semantics of mapped concepts. The use of external resources influences the results and needs further research to determine their impact. The work investigated by Stoutenburg *et al.* [14] explored the use of upper ontologies (an ontology which consists of very general terms that are common across all domains) and linguistic resources to enhance the alignment process.

The literature has demonstrated the effects of ontology evolution in mappings [4]. Gross *et al.* [6] presented the impact of ontology changes in established mappings causing modifications in semantic relations between interrelated concepts. A detailed descriptive analysis of the impact of ontology changes on mappings were presented by Dos Reis *et al.* [4]. The method proposed by Dinh *et al.* [3] aimed at identifying the most relevant concept's attributes for supporting mapping adaptation when ontologies evolve, using differences identified among current and past versions of the ontologies.

The use of ontology evolution in mapping refinements has not been investigated in the literature. Our proposed approach in this investigation differs from the above mentioned proposals because we rely only on ontology change operations, an information obtained from the ontology itself, without depending on external tools or resources. We leverage the information obtained from ontology change operations to identify refinement actions applicable to mappings. To the best of our knowledge, this approach has not been investigated in the literature. We demonstrate how the evolution of concepts can be useful to enrich the semantics of correspondences already established.

3 Preliminary Formalizations

Ontology. An ontology $\mathcal{O}$ specifies a conceptualization of a domain in terms of concepts, attributes and relationships [7]. Formally, an ontology $O = (\mathcal{C}_{\mathcal{O}}, \mathcal{R}_{\mathcal{O}}, \mathcal{A}_{\mathcal{O}})$ consists of a set of concepts $\mathcal{C}_{\mathcal{O}}$ or $Concepts_{\mathcal{O}}$ interrelated by directed relationships $\mathcal{R}_{\mathcal{O}}$. For each concept $c_k \in \mathcal{C}_{\mathcal{O}}$, $\mathcal{L}(c_k)$ defines the value of the preferred label for c_k expressing its name denoted by a natural language string. For example, *"cardio vascular diseases"* describes the label of a concept. The labels can be defined by properties in RDF schema like *rdfs:label*, and SKOS (*Simple Knowledge Organization System*) like *skos:prefLabel*. Concept $c_k \in \mathcal{C}_{\mathcal{O}}$ is associated with a set of attributes $\mathcal{A}_{\mathcal{O}}(c) = \{a_1, a_2, ..., a_p\}$. Each relationship $relation(c_1, c_2) \in \mathcal{R}_{\mathcal{O}}$ is typically a triple (c_1, c_2, r), where r is the relationship (*e.g.*, "is_a", "part_of", and "advised_by") inter-relating c_1 and c_2. We define neighbour concepts of a given entity $e \in \mathcal{C}_{\mathcal{O}}$ or $e \in \mathcal{R}$ the set of concepts with a direct relation to e. Formally, the neighbourhood of e is the set $nbh = \{cpt | cpt \in \mathcal{C}_{\mathcal{O}} \wedge dist(e, cpt) = 1\}$, where $dist(e, cpt)$ is the distance (in terms of the number of edges) between 'e' and 'cpt'.
Similarity between concepts. Given two particular concepts c_1 and c_2, the similarity between them can be defined as the maximum similarity between each couple of attributes from c_1 and c_2. Formally:

$$sim(c_1, c_2) = \arg\max \ sim(a_{1x}, a_{2y}) \tag{1}$$

where $sim(a_{1x}, a_{2y})$ is the similarity between two attributes a_{1x} and a_{2y} denoting concepts c_1 and c_2 respectively. We can compute this similarity at different linguistic levels: *character*, *string*, and *semantic* level [3].
Mapping. Given two concepts c_1 and c_2 from two different ontologies, a mapping m_{12} can be defined as:

$$m_{12} = (c_1, c_2, semType, conf) \tag{2}$$

where $semType$ is the semantic relation connecting c_1 and c_2.

The following types of semantic relation are considered: *equivalent* [$\equiv$], *narrow-to-broad* [$\leq$], *broad-to-narrow* [$\geq$]. For example, concepts can be equivalent (*e.g.*, "cabeça"$\equiv$"head") – "cabeça" in Portuguese language – one concept can be less or more general than the other (*e.g.*, "thumb"$\leq$"dedo") –"dedo" in Portuguese language – or concepts can be somehow semantically related ($\approx$). The similarity between c_1 and c_2 indicates the confidence ($conf$) of their relation [5], a high similarity denotes a high confidence. $\mathcal{L}_{XY} = \{(m_{12})_k | k \in \mathbb{N}\}$ consists of the set of mappings between two ontologies O_X and O_Y as the result of an alignment process. Cross-lingual mapping is established between O_X and O_Y with concepts denoted by different natural languages, where $\mathcal{L}(c_1)$ is expressed in language α, and $\mathcal{L}(c_2)$ is expressed in language β such that $\alpha \neq \beta$. In a monolingual mapping, O_X and O_Y have concepts denoted by the same natural language. In this approach, we are considering only mappings where $\alpha \neq \beta$.

Ontology change operations. An ontology change operation (OCO) is defined to represent a change in an attribute, in a set of one or more concepts or in a relationship between concepts. OCO is classified into two main categories: *atomic* and *complex* changes (*cf.* Table 1). Each OCO in the atomic category cannot be split into smaller operations, whereas each one in the complex category is composed of more than one atomic operation. For instance, the operation $chgA(c, a, v)$ is composed of two atomic operations, $delA(a, c)$ and $addA(a, v)$.

Table 1. Ontology change operations (OCOs) [9].

	Change operation	**Description**
A	$addC(c)$	Addition of a new concept $c \in O_X^j$
t	$delC(c)$	Deletion of an existing concept $c \in O_X^{j-1}$
o	$addA(a, c)$	Addition of a new attribute a to a concept $c \in O_X^{j-1}$
m	$delA(a, c)$	Deletion of an attribute a from a concept $c \in O_X^{j-1}$
i	$addR(r, c_1, c_2)$	Addition of a new relationship r between two concepts c_1 and c_2 which belongs to O_X^{j-1}
c	$delR(r, c_1, c_2)$	Deletion of an existing relationship r between two concepts c_1 and c_2 which belongs to O_X^{j-1}
	$chgA(c, a, v)$	Change of attribute a in concept c with the new value v
	$moveC(c, p_1, p_2)$	Moving of concept c (and its subtree) from concept p_1 to concept p_2
C	$substitute(c_i, c_j)$	Replacement of concept $c_i \in O_X^{j-1}$ by concept $c_j \in O_X^j$
o	$merge(C_k, c_j)$	Fusion of a set of multiple concepts $C_k \subset O_X^{j-1}$ into concept $c_j \in O_X^j$
m	$split(c_i, C_r)$	Split of concept $c_i \in O_X^{j-1}$ into a set of resulting concepts $C_r \subset O_X^j$
p	$toObsolete(c)$	Sets status of concept c to *obsolete* (c is no longer available)
l	$delInnerC(c_i, p_j)$	Deletion of concept c_i where $p_j \in sup(c_i)$ and $sub(c_i) \neq \emptyset$ from ontology O_X^{j-1}
e	$delLeafC(c_i, p_j)$	Deletion of leaf concept c_i where $p_j \in sup(c_i)$ and $sub(c_i) = \emptyset$ from ontology O_X^{j-1}
x	$addInnerC(c_i, p_j)$	Addition of a sub concept c_i under the concept $p_j \in sup(c_i)$ to the ontology O_X^j
	$addLeafC(c_i, p_j)$	Addition of leaf concept c_i where $p_j \in sup(c_i)$ and $sub(c_i) = \emptyset$ to the ontology O_X^j
	$revokeObsolete(c)$	Revokes obsolete status of concept c (*i.e.*, c becomes active)

We denote successive ontology versions derived from evolution by O^{j-1} and O^j to identify ontologies created in time $j-1$ and j. Changes may occur from one version to another, and we consider existing tools to automatically detect change operations [11].

Problem Statement. Consider two versions of the same source ontology O_X^{j-1} at time $j-1$ and O_X^j at time j, a target ontology O_Y^j, and a set of mappings $\mathcal{L}_{XY}^j$ between O_X^j, and O_Y^j at time j (mapping set already defined). Suppose that the frequency of new releases of O_X and O_Y is different and at time j only O_X has evolved. We assume that the evolution is likely to provide useful information for mapping refinement of $\mathcal{L}_{XY}^j$, to enrich semantic relations and obtain the refined mappings $\mathcal{L}'^j_{XY}$. All mappings in $\mathcal{L}_{XY}^j$ have initially the type of semantic relation *equivalent* [≡] or *overlapped* [≈] and we assume them as a mapping candidate set.

Given a mapping $m_{12} \in \mathcal{L}^j_{XY}$ associated with a concept c_1 affected by changes in the ontology, the challenging issue is to determine an exact and suited action of refinement to apply to m_{12}. To address this challenge, we define and formalize a set of *mapping refinement actions* (*cf.* Subsection 4.1).

The mapping refinement actions are part of refinement procedures, playing a key role to improve the quality of mappings. The objective is to enrich the mapping set by considering different semantic relations between concepts, for instance, equivalence relations can be refined to *is-a* or *part-of*.

In this investigation, we study how $\mathcal{L}^j_{XY}$ can be refined (*e.g.*, new mapping relations derived) based on ontology changes related to ontology evolution. The refined output consists of the $\mathcal{L}'^j_{XY}$. In particular, we address the following research questions:

- How to exploit ontology change operations for mapping refinement?
- Is it possible to reach mapping refinement without applying a new matching operation in the whole target ontology?
- What is the impact of using evolution information on the mapping refinement effectiveness?

4 Refinement of Biomedical Ontology Mappings across Languages

We propose and formalize a set of refinement actions aiming at refining mapping sets (Section 4.1) and how these actions are applicable in a refinement procedure (Section 4.2).

4.1 Refinement Actions

We present an approach to refine ontology mappings based on different types of ontology changes (Table 1). The proposal explores OCOs for refining mappings individually. For this purpose, we define actions as pre-defined behaviours of mapping refinement into algorithms designed to enrich ontology mappings according to ontology evolution (*cf.* Section 4.2).

The distinct actions representing different possibilities for refining mappings include: *mapping movement*, *mapping derivation*, *semantic relation modification* and *no action*. In the following, we formally describe each action. To this end, let $m_{12} \in \mathcal{L}^j_{XY}$ be the mapping between two particular concepts $c_1 \in O^j_X$ and $c_2 \in O^j_Y$. The actions are illustrated in Figure 1

Mapping derivation source. This is an action for which an existing mapping from $\mathcal{L}^j_{XY}$ derives a new mapping with the same target concept and different source concept. This action results in addition of a new mapping m_{k2} to $\mathcal{L}'^j_{XY}$.

$$\begin{aligned}&deriveS(m_{12}, c_k) \longrightarrow m_{12} \in \mathcal{L}^j_{XY} \wedge m_{k2} \notin \mathcal{L}^j_{XY} \wedge \\ &(\exists c_k \in O^j_X, m_{k2} \in \mathcal{L}'^j_{XY} \wedge sim(c_1, c_k) \geq \sigma) \wedge \\ &m_{12} \notin \mathcal{L}'^j_{XY}\end{aligned} \tag{3}$$

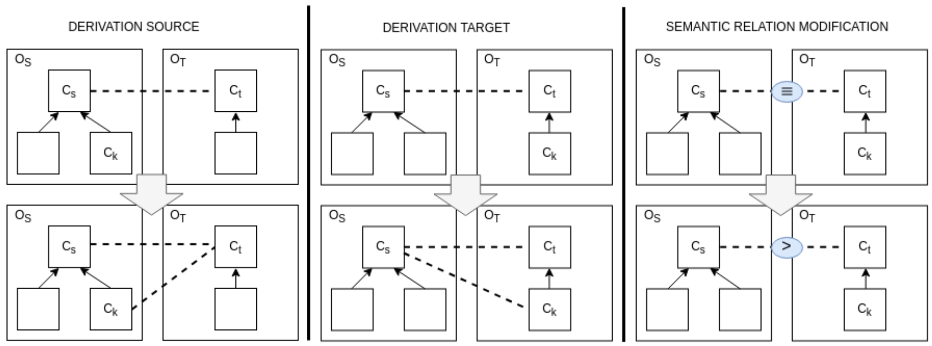

Fig. 1. Refinement actions.

where $sim(c_1, c_k)$ denotes the similarity between c_1 and $c_k \in neighborhood(c_1)$, and σ denotes the threshold used to compare the derived mapping.

Mapping derivation target. This is an action for which an existing mapping m_{12} in $\mathcal{L}^j_{XY}$ derives a new mapping with the same source and a different target. This action results in addition of a new mapping m_{1v} to $\mathcal{L}'^j_{XY}$.

$$\begin{aligned} &deriveT(m_{12}, c_v) \longrightarrow m_{12} \in \mathcal{L}^j_{XY} \wedge m_{1v} \notin \mathcal{L}^j_{XY} \wedge \\ &(\exists c_v \in O^j_Y, m_{1v} \in \mathcal{L}'^j_{XY} \wedge sim(c_1, c_v) \geq \sigma) \wedge \\ &m_{12} \in \mathcal{L}'^j_{XY} \end{aligned} \tag{4}$$

Semantic relation modification. This is an action in which the type of the semantic relation of a given mapping is modified. This action is designed for supporting the refinement of mappings with different types of semantic relations rather than only considering the type of equivalence relation ($\equiv$).

$$\begin{aligned} &modSemType(m_{12}, new\_semType_{12}) \longrightarrow m_{12} \in \mathcal{L}'^j_{XY} \wedge \\ &new\_semType_{12} \in \{\perp, \equiv, \leq, \geq, \approx \wedge \\ &semType_{12} \neq new\_semType_{12}\} \end{aligned} \tag{5}$$

The action for the modification of semantic relation can be applied in conjunction with the actions of move of mapping and derivation of mapping. That is when moving a mapping, it is also possible to modify the type of the semantic relation of such mapping. The same applies for derivation of mapping.

No Action. This action does not modify any aspect of a mapping m_{12}.

$$NoAction(m_{12}) \longrightarrow m_{12} \in \mathcal{L}^j_{XY} \wedge m_{12} \in \mathcal{L}'^j_{XY} \tag{6}$$

4.2 Refinement Procedure

The mapping refinement phase takes into account concepts from one version of the source ontology to another (O^{j-1}_X and O^j_X) to refine a candidate mapping set (suggests modifications in the set of mappings). The necessary instances of OCOs are identified from one ontology version at time $j-1$ to another at time j with a *diff* computation [9]. It generates a *diff*, which is basically a set of changes identified between two versions of the same ontology. This article considers only the changes affecting O_X^j, *i.e.*, $diff_{(O_X^{j-1}, O_X^j)}$.

The candidate mapping set $\mathcal{L}_{XY}^j$ undergoes the mapping refinement procedure. We describe the procedure in two phases:

1. The output of executed ontology change detection tools is used to identify mappings with potential of refinement. The identification is based on the type of ontology evolution operations that affected the concepts. For instance, the addition of a concept to an ontology may indicate a specialization of another concept (*e.g.*, in O_X^j, the concept "*Eagle*" was added as child of the concept "*Bird*", being the former a specialization of the latter). Therefore, any candidate mapping involving the concepts "*Eagle*" and "*Bird*" are identified with possibility of refinement.
2. After the selection of mappings for refinement, for each selected mapping from $\mathcal{L}_{XY}^j$, an action is executed based on the type of ontology change. The action may include a direct decision to perform modification in the semantic relation of the candidate mapping (*e.g.*, a $\equiv$ relationship may be replaced with a $\sqsubseteq$), or other appropriate action.

Algorithm 1 presents the main procedure to refine $\mathcal{L}_{XY}^j$. The input is the candidate mappings $\mathcal{L}_{XY}^j$ and the $diff_{(O_X^{j-1}, O_X^j)}$. For each mapping $m_{12} \in \mathcal{L}_{XY}^j$, the algorithm verifies if the concept $c_1 \in O_X^j$ was affected by change operations with the use of the $diff_{(O_X^{j-1}, O_X^j)}$. The algorithm then invokes the appropriate procedure for each case by considering addition change operations and revision change operations. If the concept was not affected by change operations from the $diff_{(O_X^{j-1}, O_X^j)}$, then *noAction* is applied to (m_{12}). The output is the refined mapping $\mathcal{L}'^j_{XY}$.

We grouped the OCOs into two categories: (i) **AdditionOCO** adds concepts or information to concepts into the ontology. It consists of OCOs by including: $addC(c)$, $addInnerC(c_s, p_s)$, $addLeafC(c_s, p_s)$, $revokeObsolete(c)$, $addA(a, c_s)$ and $addR(r, c_{s1}, c_{s2})$; (ii) The **RevisionOCO** group of ontology changes revise existing concepts. It consists of OCOs such as: $merge(C_k, c_s)$ and $split(c_i, C_s)$. In the following, we explain the procedures involved by Algorithm 1.

AdditionProcedure. This procedure is invoked when c_1 was affected by some OCO in the *AdditionOCO* group. Algorithm 2 presents the proposed strategy for refining mappings associated to addition changes. For each mapping m_{12}, the neighborhood of the both c_1 and c_2 is retrieved to perform a local rematch. The *rematch* function receives a set of source concepts C_1 and a set of target concepts C_2 and returns a similarity matrix (*simMatrix*). The objective in applying a local rematch is to compare the similarities between the neighborhood of the source and target concepts. The similarity values found then drive modifications to the semantic relation established in m_{12}.

Algorithm 1 Mapping refinement procedure.

Require: $\mathcal{L}^j_{XY}; diff_{(O^{j-1}_X, O^j_X)}$

for all $m_{12} \in \mathcal{L}^j_{XY}$ **do**
  for $c_1 \in m_{12}$ **do**
    if $AdditionOCO(c_1) \in diff_{(O^{j-1}_X, O^j_X)}$ **then**
      $AdditionProcedure(m_{12})$
    else if $RevisionOCO(c_1) \in diff_{(O^{j-1}_X, O^j_X)}$ **then**
      $RevisionProcedure(m_{12}; diff_{(O^{j-1}_X, O^j_X)})$
    else
      $noAction(m_{12})$;
    end if
  end for
end for
return $\mathcal{L}'^j_{XY}$

Algorithm 2 Mapping refinement for addition changes.

Require: m_{12}

for $c_1 \in m_{12}$ **do**
  $neighC_1 \leftarrow neighborhood(c_1)$;
  $neighC_2 \leftarrow neighborhood(c_2)$;
  $simMatrix(C_1, C_2) \leftarrow rematch(neighC_1, neighC_2)$;
  for all $(c_{1i}, c_{2i}) \in simMatrix(C_1, C_2)$ **do**
    if $c_{1i} = sup(c_1)$ **and** $(sim(c_{1i}, c_2) > sim(c_1, c_2))$ **then**
      $semType \leftarrow relation(c_{1i}, c_1)$;
      $modSemTypeM(m_{12}, semType)$;
      $deriveS(m_{12}, c_{1i})$;
    end if
    if $(c_{2i} = sup(c_2)$ **or** $c_{2i} = sub(c_2))$ **and**
    $sim(c_1, c_{2i}) => sim(c_1, c_2)$ **then**
      $deriveT(m_{12}, c_{2i})$;
    end if
  end for
end for

For example, if $sim(sup(c_1), c_2) > sim(c_1, c_2)$, the algorithm modifies the semantic relation in m_{12} to the same semantic relation of $sup(c_1)$ and c_1 and add a new mapping between $sup(c_1)$ and c_2. The local rematch also helps establishing a derivation of mapping when the $sim(c_1, sub(c_2)) \geq sim(c_1, c_2)$ or $sim(c_1, sup(c_2)) \geq sim(c_1, c_2)$.

We present an example to illustrate the AdditionProcedure. Ontology O_X evolved over time by generating different versions from time $j-1$ to time j. Figure 2(A) illustrates the changes. A set of candidate mappings $\mathcal{L}^j_{XY}$ between O^j_X and O^j_Y, at time j, is given as input for the refinement procedure. Figure 2(B) illustrates the mapping $m_12 \in \mathcal{L}^j_{XY}$ between concepts c_1 "*Angina*" and c_2 "*Cardiopatia*". The refinement procedure requires as input the list of change operations (OCOs) detected from one version of the ontology to another. Similarity values between concept c_1 "*Angina*" and the concepts of the neighborhood of the target concept "*Cardiopatia*" at time j are calculated via local rematch (cf. Figure 2(C)). If the similarity value between the concepts c_1 "*Angina*" and some neighbor c_{2i} of c_2 is higher than the original similarity value given by $sim(c_1, c_2)$, *i.e.* $sim(c_1, c_{2i}) \geq sim(c_1, c_2)$, the algorithm derives a mapping between c_1 and c_{2i} to reflect this finding (cf. Figure 2(D)).

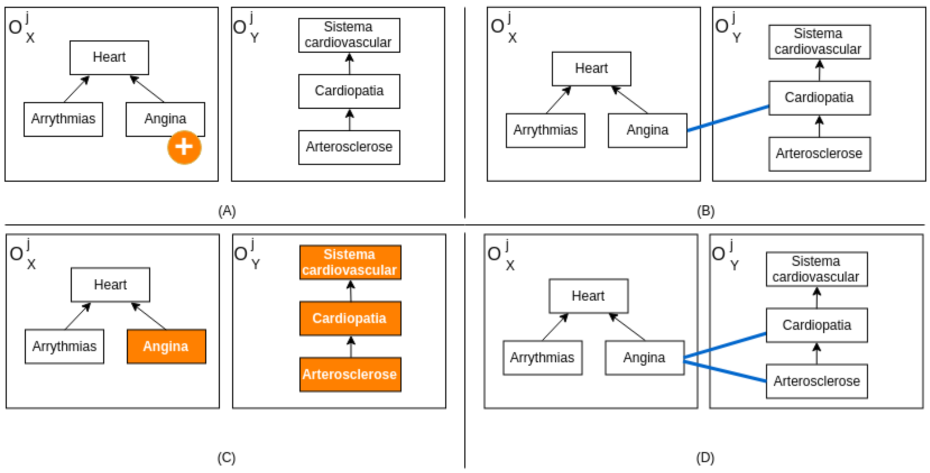

Fig. 2. (A) Ontology change operations (OCO) on O_X. (B) Illustration of the mapping $m_{12} \in \mathcal{L}^j_{XY}$ candidate for refinement (C) Computing similarity values between c_1 and the $neighborhood(c_2)$ (D) Resulting $\mathcal{L}'^j_{XY}$ after our refinement procedure (application of the derivation action).

RevisionProcedure. This procedure is used to refine mappings when c_1 was affected by some OCO in the *RevisionOCO* group. Algorithm 3 describes the proposed strategy for the refinement. For each input mapping m_{12}, the algorithm retrieves the concepts from O^{j-1}_X involved in merge or split ontology change operations. In the *merge* operation, an initial set of concepts $C_k \subset O^{j-1}_X$ gives place to a concept $c_1 \in O^j_X$. On the other hand, in a *split* operation, an initial concept $c_1 \in O^{j-1}_X$ is split in a set of concepts $C_s \subset O^j_X$.

The algorithm extracts the before evolution concepts c_1 (in the split) and the set of concepts C_k (in the merge) and computes the similarity between them with $c_2 \in m_{12}$. The algorithm explores the similarity values between c_2 and $\{c_1, C_k\}$ to extract information and refine m_{12}. For example, an useful information for refinement is the similarity value between the concept $c_i \in O^{j-1}_X$ involved in the

Algorithm 3 Mapping refinement for revision changes.

Require: $m_{12}; diff_{(O_X^{j-1}, O_X^j)}$
for $c_1 \in m_{12}$ **do**
  if $split(c_i, C_s) \in diff_{(O_X^{j-1}, O_X^j)}$ **and** $(c_1 \in C_s)$ **then**

    if $sim(c_i, c_2) > sim(c_1, c_2)$ **and** $semType(m_{12}) == \equiv$ **then**
      $modSemTypeM(m_{12}, \leq);$
    end if
  end if
  if $merge(C_k, c_1) \in diff_{(O_X^{j-1}, O_X^j)}$ **then**
    $neighC_2 \leftarrow neighborhood(c_2);$
    $simMatrix(C_k, C_2) \leftarrow rematch(C_k, neighC_2);$
    for all $(c_{ki}, c_{2i}) \in simMatrix(C_k, C_2)$ **do**
      if $(c_{2i} = sup(c_2)$ **or** $c_{2i} = sub(c_2))$ **and**
      $sim(c_{ki}, c_{2i}) \geq sim(c_1, c_2)$ **then**
        $deriveT(m_{12}, c_{2i});$
      end if
    end for
  end if
end for

split of $c_1 \in m_{12} \wedge c_1 \in C_s$ and c_2. If $sim(c_i, c_2) > sim(c_1, c_2)$, we can infer that c_1 and c_2 do not hold an $\equiv$ relation because c_1 is the result of the split operation of c_i into more specific concepts, not equivalent concepts. We can use this information to refine the semantic relation of m_{12}.

5 Evaluation

We evaluate our approach to ontology alignment refinement by applying it to ontologies in the biomedical domain.

We considered the *Logical Observation Identifiers Names and Codes* (*LOINC*) ontology published in English and its linguistic variant in Spanish. *LOINC* provides a standard for identifying clinical information (laboratory and clinical test results) in electronic reports. *LOINC* is freely available and widely used in 175 countries[4]. The English variant of LOINC contains 89,271 entities and the Spanish variant contains 54,599 entities.

LOINC presents a regular update schedule of twice a year, providing an amount of ontology changes in every new version available. Update changes in the ontology entities are provided in every release in a separate document, specifying the change operations undergone by the entities. The version selected for this evaluation was the 2.65, released in December 2018.

Our proposed technique requires an initial mapping set as input. For this purpose, we used the mapping set already established between the two linguistic

[4] As of September 12, 2019.

variants of *LOINC* (one ontology in English language and the other in Spanish language). Each entity has a unique permanent identifier named *LOINC code* (in the sense that it cannot be reused even if the entity is deprecated). This code is invariable across linguistic variants. We use LOINC code to identify equivalent entities between the two selected ontologies. In the release version 2.65, the changes available in the document of updates were **AdditionOCO** and minor changes in labeling. In particular, we focused our evaluation in **AdditionOCO** actions. Only the updates performed from version 2.64 to version 2.65 of the ontology in the English variant are considered for this experiment.

Based on this candidate mapping set, we applied our defined Algorithm 1 to invoke the appropriate refinement actions based on the change operations undergone by the entities $e_k \in LOINC_{en}^{2.65}$ participating in the alignment. We employed the Levenshtein edit-distance [10] as similarity measure, aided by automatic translation from Spanish to English by the Google Translate API. The automatic translation was required to enable the comparison of the label of entities in the same language (in this case, English language).

Our results present real-life examples of the outcome in applying our technique to LOINC entities. Table 2 presents the results as the effect of applying the refinement actions. The original input mapping set $\mathcal{L}_{en-es}^{2.65}$ contains 54599 correspondences. In the last version update, from release version 2.64 to 2.65, over 8000 entities have suffered some **OCO**, with 1408 entities undergoing **AdditionOCO**. All of these entities were involved as part of mappings, thus allowing the application of the **AdditionProcedure** described in Subsection 4.2. The technique has refined the mapping set and the refinement actions performed generated 1513 new semantically enriched mappings, by increasing the number of mappings to a total of 56113.

Table 2. Evaluation results.

Input mapping size	Total of changes of **AdditionOCO** type	Total of changes affecting mappings	Mapping size after applying refinement actions
54,599	1,408	1,408	56,113

Figure 3(A) presents an example of an entity added during the ontology evolution from the release version 2.64 to 2.65 of LOINC. Concept "*Zika virus Ab.IgG*" is added as a sub concept of "*Zika virus*" concept.

The refinement of the candidate mapping set relies on the similarity values computed between the concept "*Zika virus Ab.IgG*" and the concepts of the neighborhood of the target concept "*Virus Zika IgG*". To this end, the algorithm 2 performed a cross-lingual local rematch defined in its step 2. As a result of this operation, the algorithm applies refinement action *deriveT* and derives a mapping between "*Zika virus Ab.IgG*" and "*Virus zika*" (cf. Figure 3(B)).

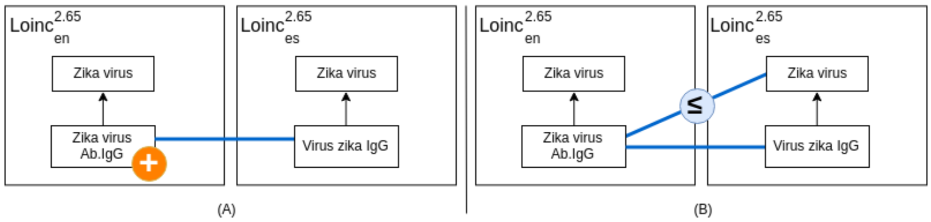

Fig. 3. (A) Addition of the concept "*Zika virus Ab.IgG*" from $LOINC_{en}^{2.64}$ release version to $LOINC_{en}^{2.65}$ (B) Resulting mapping set after refinement action application

6 Discussion

In this investigation, we assumed that ontology evolution is useful to decide on the application of mapping refinement actions and improve the mapping quality outcome. To the best of our knowledge, the use of ontology change operations for mapping refinement has never been proposed in literature. This aspect refers to the key originality of this article. We demonstrated the usefulness of ontology changes to aid the process of ontology mapping refinement in a case of aligned biomedical ontologies.

The actions performed during the refinement procedure enrich the candidate mapping set with semantic context, which is beneficial for ontology merging and system integration. Our proposal defined algorithms that reach mapping refinement without applying a new matching operation with the whole target ontology. In addition, our technique enables the update of the semantic type of mappings. The current approach focuses only on *is-a* and *part-of* relationships. Other relationship types will be addressed in future work.

The main advantage of using evolution information is the possibility of refinement without the need of an external resource. The information required to refine is available in the ontology itself or can be computed based on a early version of the ontology, by using ontology diff computation tools to calculate the change history. This is particularly useful when external resources are unavailable to aid in the refinement task. Due to the lack of experimental results concerning mapping refinement in literature, we were unable to compare our method to others approaches.

The use of OCOs for refinement purposes is limited to mappings with at least one participant ontology with multiple versions available to calculate history changes; or a list of updates between versions must be available. Our proposed procedure depends on the set of ontology changes, thus only mappings with entities associated with ontology change(s) are eligible for the procedure. This limits the amount of mappings that can be refined with this technique. For example, in the conducted evaluation we were able to refine mappings where one of the participants has undergo **AdditionOCO** action, because that was the only change action available for the entities participating in mappings.

Our main goal with this evaluation was to assess the usefulness of evolution change information in mapping refinement, by verifying if the semantic relationships in mappings are expanded beyond equivalence in a meaninful way. The correctness of generated output will be evaluated in future work.

Biomedical ontologies usually are syntactically regular and this condition might not be found in other domains. Further investigations are required to verify the applicability of our proposed technique to different domains.

The procedure uses similarity measures for local rematch. The selection of the applicable similarity measure depends on the addressed problem, because there are similarity measures which depends on background knowledge, such as, semantic similarity measures relying on semantic networks of a specific domain. In our experimental evaluation, we chosen a simple, widely used and domain neutral similarity measure. Nevertheless, any similarity measure appropriated for the problem can be used.

7 Conclusion

Ontology mapping refinement remains an open research problem. The result of mapping refinement increases the usefulness of mapping sets, benefiting the semantic data integration of systems. This article proposed an original approach with the use of ontology change operations detected during ontology evolution to leverage mapping refinement. We contributed with the formalization of refinement actions and defined algorithms to apply them based on ontology evolution operations. We demonstrated the evaluation of the technique using biomedical real-world ontologies across different languages. Future work involves to include domain specialists to evaluate the correctness proposed concept mappings and their specific type of semantic relation. We also plan to investigate this approach in monolingual mappings and evaluate the impact of other similarity measures in the quality of the refinement.

Acknowledgment

This work is supported by CNPq (grant #307560/2016-3), the São Paulo Research Foundation (FAPESP) (Grant #2017/02325-5). This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.[5]

References

1. Aleksovski, Z., Klein, M., ten Kate, W., van Harmelen, F.: Matching unstructured vocabularies using a background ontology. In: Proceedings of the 15th International Conference on Managing Knowledge in a World of Networks (EKAW 2006). pp. 182–197 (2006)

[5] The opinions expressed in this work do not necessarily reflect those of the funding agencies.

2. Arnold, P., Rahm, E.: Enriching ontology mappings with semantic relations. Data & Knowledge Engineering **93**, 1–18 (2014)
3. Dinh, D., Dos Reis, J.C., Pruski, C., Da Silveira, M., Reynaud-Delaître, C.: Identifying relevant concept attributes to support mapping maintenance under ontology evolution. Journal of Web semantics **29**, 53–66 (2014)
4. Dos Reis, J.C., Pruski, C., Da Silveira, M., Reynaud-Delaître, C.: Understanding semantic mapping evolution by observing changes in biomedical ontologies. Journal of biomedical informatics **47**, 71–82 (2014)
5. Euzenat, J., Shvaiko, P.: Ontology matching, pp. 42,71–84,306. Springer, Heidelberg, 2nd edn. (2013)
6. Gross, A., Hartung, M., Thor, A., Rahm, E.: How do computed ontology mappings evolve?-a case study for life science ontologies. In: 2nd Joint Workshop on Knowledge Evolution and Ontology Dynamics. vol. 890 (2012)
7. Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing. International Journal of Human-Computer Studies **43**, 907 – 928 (1995)
8. Hamdi, F., Safar, B., Niraula, N.B., Reynaud, C.: Taxomap alignment and refinement modules: Results for oaei 2010. In: Proceedings of the 5th International Workshop on Ontology Matching (OM-2010). vol. 689, pp. 212–219 (2010)
9. Hartung, M., Gross, A., Rahm, E.: COnto-Diff: Generation of Complex Evolution Mappings for Life Science Ontologies. J. Biomedical Informatics **46**, 15–32 (2013)
10. Levenshtein, V.I.: Binary codes capable of correcting deletions, insertions, and reversals. Soviet Physics Doklady **10**, 707–710 (1966)
11. Noy, N.F., Musen, M.A., et al.: Promptdiff: A fixed-point algorithm for comparing ontology versions. AAAI/IAAI **2002**, 744–750 (2002)
12. Raunich, S., Rahm, E.: Atom: Automatic target-driven ontology merging. In: 27th International Conference on Data Engineering. pp. 1276–1279. IEEE (2011)
13. Spiliopoulos, V., Valarakos, A., Vouros, G.: Csr: discovering subsumption relations for the alignment of ontologies. The Semantic Web: Research and Applications pp. 418–431 (2008)
14. Stoutenburg, S.K.: Acquiring advanced properties in ontology mapping. In: 2nd PhD Workshop on Information and Knowledge Management. pp. 9–16 (2008)
15. Trojahn, C., Fu, B., Zamazal, O., Ritze, D.: State-of-the-art in multilingual and cross-lingual ontology matching. In: Towards the Multilingual Semantic Web, pp. 119–135. Springer (2014)