A Category-theoretic Approach for the Detection of Conservativity Violations in Ontology Alignments

Cauã Roca Antunes[1], Alexandre Rademaker[2], Mara Abel[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

[2]Escola de Matemática Aplicada – Fundação Getúlio Vargas (FGV)
Rio de Janeiro – RJ – Brazil

crantunes@inf.ufrgs.br, arademaker@gmail.com, marabel@inf.ufrgs.br

Abstract. *Ontologies are formal specifications that enable inferential processes over shared knowledge. In distributed contexts, applications frequently need to access information from multiple ontologies. For this end, concepts of two different ontologies must be matched through an alignment. If the alignment is not semantically sound, however, the integration of the ontologies may lead to unintended consequences. One type of possible consequences is the introduction of new subsumption relations between concepts from one of the input ontologies, which violate the conservativity principle. We propose a method based on the mathematical formalism of Category Theory for detecting such violations.*

1. Introduction

The Semantic Web is an extension of the traditional World Wide Web where information is given well-defined meaning [Berners-Lee et al. 2001]. Such meaning is specified in ontologies, i.e., formal and explicit specifications of a shared conceptualization [Studer et al. 1998]. However, different people and groups may build distinct ontologies dealing with the same subject or domain. Applications frequently have to access multiple related ontologies in order to integrate all required information. In order to allow this, modelers must create alignments between the ontologies, either manually or automatically. Such alignments frequently match concepts imperfectly, causing inconsistencies.

The conservativity principle states that the merge of two ontologies through an alignment should not introduce new subsumption relations between concepts from the same source ontology. We say that a concept *c* subsumes a concept *d* if every instance of *d* is also an instance of *c*. If an alignment violates the conservativity principle, the merged ontology does not preserve the original meaning specified by the source ontology. For example, if an alignment matches concepts *person*, *client* and *company* from two ontologies *A* and *B*, where ontology *A* specifies that *person* subsumes *client* (that is, every client is a person) and ontology *B* states that *client* subsumes *company* (i.e., every company is a client), a query for *person* in the merged ontology will return every instance of *company* in the knowledge base, which is clearly not the expected result. We distinguish two types of conservativity principle violations:

1. **Violation of subsumption conservativity.** An alignment violates subsumption conservativity if it introduces new subsumption relations between concepts from the same input ontology.
2. **Violation of equivalence conservativity.** An alignment violates equivalence conservativity if it introduces new equivalence relations between concepts from the same input ontology. This may happen due to the introduction of a circular chain of subsumption relations or due to two concepts in one of the source ontologies being mapped to a single concept in the other ontology.

Category theory is a branch of mathematics that studies the structure present in systems of composable relations. These relations, called morphisms, are abstractions from several distinct mappings between mathematical objects, including functions, set-theoretic relations, graph homomorphisms, linear mappings between vector spaces, and others. Category theory and its morphisms provide a sound formal basis for the study of ontologies and their alignments. We use the formalisms of category theory to reduce the problem of detecting conservativity principle violations to the computation of two pullbacks[1] followed by a verification of the existence of a particular morphism between them.

The remainder of this paper is organized as follows. Section 2 introduces the fundamentals of category theory and describes the concepts that are relevant to this work. Section 3 presents works from the literature that deal with categories of ontologies and their constructions, and discuss other approaches to the problem of detecting conservativity violations. We present a category of ontologies in Section 3 and describe our method in Section 5. Section 6 contains a brief discussion of the merits of our approach and aspects for future improvement.

2. Category Theory Fundamentals

[Adámek et al., 1990] defines a category as a quadruple $C = (O, hom, id, \circ)$, consisting of:

- a class O whose members are C-objects,
- for each pair (A,B) of C-objects, a set $hom(A,B)$, whose members are C-morphisms from A to B,
- for each C-object A, a morphism $id_A{:}A \to A$, called the C-identity on A, and
- a composition law $\circ$ associating each pair of C-morphisms $f{:}A \to B$ and $g{:}B \to C$ to a C-morphism $g \circ f{:}A \to C$, called the composite of f and g.

Subject to the conditions that (1) composition is associative, that is, for any three morphisms $f{:}A \to B$, $g{:}B \to C$ and $h{:}C \to D$, $h \circ (g \circ f) = (h \circ g) \circ f$, (2) C-identities are neutral with respect to composition, i.e., for any morphism $f{:}A \to B$, $id_B \circ f = f = f \circ id_A$, and (3) the sets $hom(A,B)$ are pairwise disjoint.

A diagram in a category A is a selection of some of its objects and morphisms. A source for a diagram is a pair (x,f_i), consisting of an object x and a family of morphisms $f_i{:}x \to d_i$ with domain x and codomain indexed by the diagram, that is, a group of morphisms from x to each object in the diagram. If for any morphism $g{:}d_i \to d_j$ in the diagram the triangle formed by g, f_i and f_j commutes, i.e., $g \circ f_i = f_j$, then the source (x,f_i)

1 We describe pullbacks, morphisms and other category-theoretic concepts in Section 2.

is called a cone. If (x,f_i) is a terminal cone, that is, for every other cone (x',f_i') there exists a unique morphism $h:x'\rightarrow x$ such that the resulting diagram commutes, (x,f_i) is a limit. We show a cone and a limit for a diagram with the morphism $g:d_1\rightarrow d_2$ in Figure 1a.

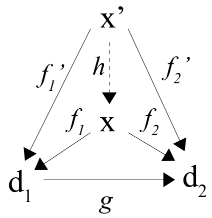

Figure 1a. A limit *x* and a cone *x'*.

If we reverse the direction of the morphisms in the previous definitions, that is, if we exchange each morphism's domain for its codomain, we arrive at the dual categorical constructions. Thus, the dual to a source is a sink, a pair (x,f_i) consisting of an object x and a family of morphisms $f_i:d_i\rightarrow x$ with codomain x and domain indexed by the diagram, that is, a group of morphisms from each object in the diagram to x. A commutative sink is a cocone, which is dual to a cone. An initial cocone is a colimit, i.e., the dual to a limit. We show a cocone and a colimit for a diagram with the morphism $g:d_1\rightarrow d_2$ in Figure 1b.

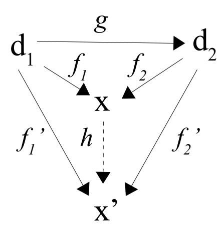

Figure 1b. A colimit *x* and a cocone *x'*.

In this work we are particularly interested in a specific type of limit and its dual colimit, which are respectively pullbacks and pushouts. Pullbacks are limits of diagrams containing two morphisms $f:A\rightarrow C$ and $g:B\rightarrow C$ with a shared codomain. Pushouts are colimits of diagrams containing two morphisms $f:A\rightarrow B$ and $g:A\rightarrow C$ that share their domain. Figure 2 depicts both constructions.

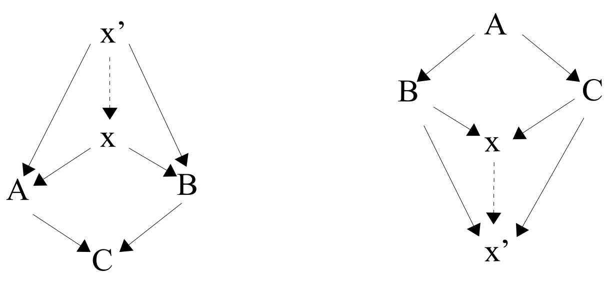

Figure 2. A pullback *x* with a cone *x'* (left) and a pushout *x* with a cocone *x'* (right).

3. Related Work

Traditionally, authors define categories of ontologies with total mappings as morphisms. Since alignments are rarely complete, dealing instead with only a subset of the ontology concepts and relations, they need to be formalized as more complex structures. [Bench-Capon and Malcom 1999] defines relations between two ontologies O_1 and O_2 as structures composed of a third ontology O and morphisms $x_i : O \rightarrow O_i$ for $i = 1, 2$. Thus, entities in O_1 and O_2 are "matched" by being mapped from the same entity in O by x_1 and x_2. Later, [Zimmerman et al. 2006] named such structures V-alignments due to their shape, and defined the operation of ontology merging as a pushout over the alignment, as well as three operations over alignments using limits and colimits, namely alignment composition, union and intersection. In addition to the category-theoretic constructions described in previous works, [Cafezeiro and Haeusler 2007] and [Cafezeiro et al. 2008] demonstrated that the pullback over two ontology mappings is the intersection of two ontologies in the context of a third, broader one. We shall build upon these constructions in the following sections.

To the best of our knowledge, no other work has dealt with the problem of detecting conservativity violations in a category-theoretic context. However, several approaches based on different formalisms can be found in the literature. The approach proposed by [Jiménez-Ruiz et al. 2009] checks only for violations of equivalence conservativity (type 2 described above) by verifying if the alignment maps directly two different concepts from one source ontology to a single concept in the other, ignoring the cases when such violations arise indirectly from the inclusion of circular subsumption relations.

The method applied by [Ivanova and Lambrix 2013] and [Lambrix and Liu 2013] computes the integrated ontology over a network of alignments and use a reasoner to infer new subsumption relations. Nevertheless, they treat the introduction of new subsumption relations as evidence of incompleteness in the source ontologies, and not of an incorrect alignment.

[Solimando et al. 2014] reduce the problem of detecting conservativity violations to one of concept satisfiability. In order to do so, the authors follow the assumption of disjointness, which states that all concepts that do not share subsumees are disjoint. For many ontologies, however, this is not a reasonable assumption, since expliciting common subsumees for every pair of concepts frequently leads to a combinatorial explosion of concepts. The same authors later introduced a technique for the detection of equivalence conservativity violations by searching for loops in graphs where each node represent a concept and each arc a subsumption relation, using the two approaches together in a multi-strategy method for detecting conservativity violations of both types [Solimando et al. 2017].

4. A Category of Ontologies

We begin by defining our category *Ont* of ontologies. Objects in *Ont* are ontologies in the form of tuples (C, R, S, A), where C is a set of concepts, R is a set of relations between concepts, $S \subseteq C \times C$ is a transitive, reflexive and antisymmetric subsumption relation given by $S(c, d)$, i.e., *c subsumes d* $\leftrightarrow \forall x$, *instanceOf* $(x, d) \rightarrow$ *instanceOf* (x, c), and A is a set of axioms governing such concepts and relations. With this definition we intend to abstract from representational aspects and therefore we assume that every concept and relation is explicit in the ontology tuple. If the actual representation of the ontology (in some ontology representation language) contains implicit knowledge that needs to be inferred, the required reasoning tasks must be performed as a preparatory step. Morphisms in *Ont* are total ontology mappings $f{:}A\rightarrow B$ between ontologies *A* and *B* with components $f_C{:}C_A\rightarrow C_B$ and $f_R{:}R_A\rightarrow R_B$ which map respectively concepts and relations[2], such that the mappings preserve relations, that is:

(1)$\forall\ c, d \in C_A, S_A(c, d) \rightarrow S_B(f_C(c), f_C(d)),$ and

(2)$\forall\ r \in R_A, \forall\ c, d \in C_A, r(c, d) \rightarrow \exists\ r' \in R_B, f_R(r) = r' \wedge r'(f_C(c), f_C(d)).$

Composition in *Ont* is usual function composition on each component of the morphisms. Since function composition is associative and always exists for any two functions *f* and *g* with *Codomain(f) = Domain(g)*, in order to prove that the category laws for composition hold in *Ont*, we must prove that the composition of two morphisms is always a morphism and that identities exist and are neutral with respect to composition. This means that given two morphisms $f{:}A\rightarrow B$ and $g{:}B\rightarrow C$, which follow rules (1) and (2), $g \circ f$ must also follow such rules. Since *f* preserves relations, for any relation *r* in R_A that holds between concepts *c* and *d* in C_A, $f_R(r)$ must hold between $f_C(c)$ and $f_C(d)$. Since *g* is a total mapping on both concepts and relations, it maps $f_R(r)$ to a relation in R_C and both $f_C(c)$ and $f_C(d)$ to concepts in C_C. Additionally, since *g* also preserves relations, we have that $g_R(f_R(r))$ must hold between $g_C(f_C(c))$ and $g_C(f_C(d))$. Similarly, since both *f* and *g* preserve subsumption, for any pair of concepts *c* and *d* in C_A, if *c* subsumes *d*, then $f_C(c)$ subsumes $f_C(d)$ and $g_C(f_C(c))$ subsumes $g_C(f_C(d))$. Identities simply map each concept and relation to itself, i.e., $i_C(c) = c$ and $i_R(r) = r$. As required for identities, such mappings are neutral on composition, since for any *f, g* and *x*, $i(f(x)) = f(x)$ and $g(i(x)) = g(x)$.

5. Our Approach

Given ontologies *A* and *B* and a V-alignment $(V, f_i{:}V\rightarrow i)$ for $i = A, B$, we wish to verify if the alignment leads to a violation of the conservativity principle. Since every concept and relation in *V* must be mapped by each f_i to a concept or relation in the corresponding ontology, the new subsumption relations introduced by the alignment cannot possibly be in *V*. Instead, they must hold in the other aligned ontology, between concepts which are mapped by f_j, where $j \neq i$. Thus, in order to answer our original question, we must find the sub-ontologies *A'* and *B'*, where *A'* contains only the concepts and relations in *V* plus every subsumption relation between the mapped concepts in *A* and *B'* contains only the concepts and relations in *V* plus every subsumption relation between the mapped concepts in *B*, and check if *A'* and *B'* share all their subsumption relations, i.e., there is no subsumption relation in *A'* that is not in *B'* and vice versa.

Therefore, first we must find *A'* and *B'*. We achieve this by computing the pullbacks $(i', g_i{:}i'\rightarrow i, h_i{:}i'\rightarrow V^*)$ of the diagrams containing the morphisms $j_i{:}i\rightarrow i^*$ and $k{:}V^*\rightarrow i^*$, where V^* is an ontology with the same concepts and relations from *V* but with additional subsumption relations such that each concept subsumes every other, and i^* is the merge of V^* and *i* computed through a pushout using *V* as alignment, along with f_i and the trivial inclusion $v{:}V\rightarrow V^*$. That is, i^* is the ontology *i* extended with additional subsumption relations so that each concept in the image of f_i subsumes every other. We propose that the ontology *i'* in the pullback just described is the smallest sub-ontology from *i* with every concept in *V* and every subsumption relation that holds between them in *i*.

2 Axioms are not yet in the scope of this definition

Proof. Suppose there exists a concept c in i to which a concept in V is mapped by f_i and there is no concept in i' that is mapped by g_i to c. Then, we may build an ontology i'' that contains every concept and relation in i' plus c, along with morphisms $q{:}i''\rightarrow i$ and $r{:}i''\rightarrow V^*$ which map every concept and relation to itself. Thus, (i'', q, r) is a cone for the diagram with morphisms j_i and k_i and there is no morphism $p{:}i''\rightarrow i'$ such that $g_i \circ p = q$ and $h_i \circ p = r$, since c is in the image of q but not of g_i. Therefore, i' is not the pullback, contradicting our initial assumption. Similarly, if we suppose that there exists a subsumption relation s in i that is not in i', we may construct a different i'' with every concept and relation in i' plus s, mapping concepts and non-subsumption relations exactly as for i'. Thus, we again have a cone for the diagram, and again no morphism $p{:}i''\rightarrow i'$ can be found, since such mapping would break the requirement that morphisms should preserve subsumption relations. Therefore, i' again is not the pullback. On the other hand, if there exists a concept c in i' that is not in V, then it either is mapped to a concept in i that is also not in V, and thus there exists no h_i such that the diagram commutes, or it is mapped to a concept in i to which another concept d is also mapped, and then there would exist an ontology i'' where c and d are collapsed in a single concept e, along with morphisms $q{:}i''\rightarrow i$ and $r{:}i''\rightarrow V^*$ such that $g_i(c) = g_i(d) = q(e)$ and $h_i(c) = h_i(d) = r(e)$. However, there are two morphisms p_1 and $p_2{:}i''\rightarrow i$, with $p_1(e) = c$ and $p_2(e) = d$, such that the diagram commutes, that is, $g_i \circ p_1 = q = g_i \circ p_2$ and $h_i \circ p_1 = r = h_i \circ p_1$, contradicting the uniqueness restriction from the definition of limit.

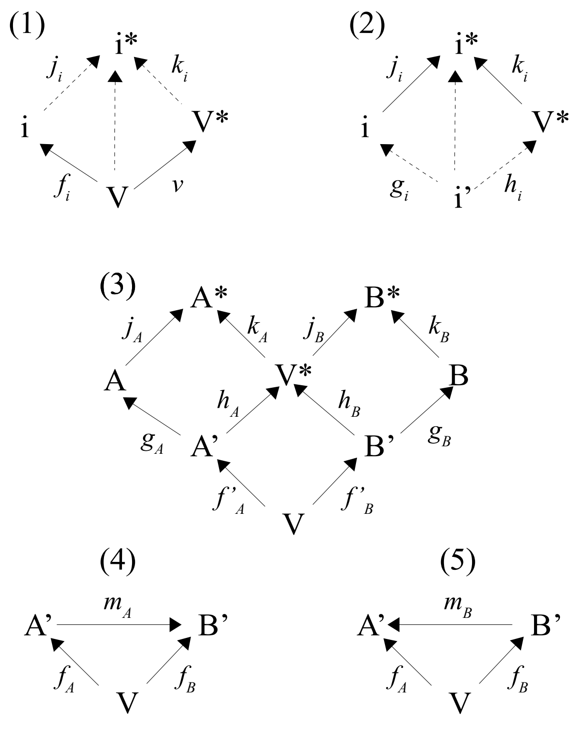

Figure 3. Our approach step by step.

Since (i', g_i, h_i) is the pullback and (V, f_i, v) is a cone for the diagram, depicted respectively in (2) and (1) in Figure 3, then there exists a single morphism $f_{i'}{:}V\rightarrow i'$ such that $g_i \circ f_{i'} = f_i$ and $h_i \circ f_{i'} = v$. Considering that A' and B' have exactly every concept and relation in V plus any subsumption relation between those concepts in A or B respectively, if there is a morphism $m_A{:}A'\rightarrow B'$ such that $m_A \circ k_A = k_B$, then every subsumption relation between concepts in V that holds in A also holds in B. Otherwise, i.e., if there is no such morphism, the alignment introduces at least one new subsumption relation between concepts from B and thus violates the conservativity principle. Symmetrically, if there is no $m_B : B' \rightarrow A'$ such that $m_B \circ k_B = k_A$, a new subsumption relation is introduced between concepts from A.

Figure 3 depicts our approach step by step. Step (1) is the construction of i^* trough a pushout. Step (2) is the computation of i' as a pullback. Then, (3) shows the complete diagram after both sides of the alignment have been analyzed. (4) and (5) show the commutative triangles formed with m_A and m_B respectively. We note that this approach is also enough to detect equivalence conservativity violations, since if two concepts c and c' in A are mapped to a single concept d in B, then it is impossible to build a mapping $m_B : B' \rightarrow A'$ such that $m_B(d) = c \wedge m_B(d) = c'$ unless $c = c'$.

From the category-theoretic constructions previously described, we build the Algorithm 1 to find the sub-ontologies A' and B'. Taking advantage of the knowledge that i^* is the pushout over f_i and v, and that therefore the only concepts and non-subsumption relations in i' are those in the image of f_i, the algorithm constructs i' directly from the mapping and then includes the subsumption relations found in i.

Algorithm 1. findSubOntology algorithm for finding minimum sub-ontology

```
Input: V, i: ontologies; f_i: V→i: mapping.
Output: i': minimum sub-ontology containing all subsumption relations in i between concepts in V.
        f_i': V→i': ontology mapping.
C_i' ← Ø          // initialize the set of concepts of the sub-ontology as an empty set
S_i' ← Ø          // initialize the subsumption relation in the sub-ontology as an empty relation
f_Ci' ← Ø
for each c_1 ∈ C_V do                 // for every concept in V
    if f_Ci(c_1) ∉ C_i' do                    // if it is not yet in i'
        C_i' ← C_i' ∪ {f_Ci(c_1)}                     // add it to i'
        for each f_Ci(c_2) ∈ C_i' do                  // then, for every concept already in i'
            if (f_Ci(c_1), f_Ci(c_2)) ∈ S_i do                // check if they should subsume
                S_i' ← S_i' ∪ {(f_Ci(c_1), f_Ci(c_2))}        // each other and add the relation
            end if
            if (f_Ci(c_2), f_Ci(c_1)) ∈ S_i do
                S_i' ← S_i' ∪ {(f_Ci(c_2), f_Ci(c_1))}
            end if
        end for
    end if
    f_Ci' ← f_Ci' ∪ {(c_1, f_Ci(c_1))}          //update f_i' with the new concept mapping
end for
R_i' ← Ø
f_Ri' ← Ø
for each r ∈ R_V do                   // for every relation in V
    if f_Ri(r) ∉ R_i' do                      // if it is not yet in i'
        R_i' ← R_i' ∪ {f_Ri(r)}         // add it
    end if
    f_Ri' ← f_Ri' ∪ {(r, f_Ri(r))}      //update f_i' with the new relation mapping
end for
i' ← (C_i', R_i', S_i', Ø)
f_i' ← (f_Ci', f_Ri')
return (i', f_i')
```

Algorithm 2 takes two ontologies and a V-alignment as input and checks if the alignment is conservative. We use the Algorithm 1 to find the corresponding sub-ontologies and then build mappings between them by matching each entity in each ontology to the concept or relation in the other to which it is aligned. Then, the algorithm checks if the mappings are functional (i.e., no entity may be mapped to more than one entity in the target ontology) and if they preserve the subsumption relations, as required for morphisms in our category *Ont* of ontologies. If they do, we have a morphism between the sub-ontologies and, as previously discussed, the V-alignment does not violate the principle of conservativity. We note that the operator ⊕ here denotes exclusive logical disjunction, i.e., $p \oplus q$ is true if and only if p is true or q is true, but not both.

Algorithm 2. isConservative algorithm for detecting conservativity violations

Input: A, B: ontologies; $(V, f_A{:}V \rightarrow A, f_B{:}V \rightarrow B)$: V-alignment.
Output: **true** if the alignment is conservative, **false** otherwise.

```
(A', f_A') ← findSubOntology(V, A, f_A)          // find sub-ontology A'
(B', f_B') ← findSubOntology(V, B, f_B)          // find sub-ontology B'
m_A ← Ø
m_B ← Ø
flag ← true
for each e ∈ C_V ∪ R_V do          // for each entity in the alignment
    if f_A'(e) ∈ Dom(m_A) do          // if its match in A' is already mapped to something by m_A
        if m_A(f_A'(e)) ≠ f_B'(e) do          // and it does not match the mapping by f_B'
            flag ← false          // then the alignment is not conservative
        end if
    else do:          // if it is not mapped to anything by m_A
        m_A ← m_A ∪ {(f_A'(e), f_B'(e))}          // map it to the same entity to which e is mapped
    end if          // by f_B'
    if f_B'(e) ∈ Dom(m_B) do          // if its match in B' is already mapped to something by m_B
        if m_B(f_B'(e)) ≠ f_A'(e) do          // and it does not match the mapping by f_A'
            flag ← false          // then the alignment is not conservative
        end if
    else do:          // if it is not mapped to anything by m_B
        m_B ← m_B ∪ {(f_B'(e), f_A'(e))}          // map it to the same entity to which e is mapped
    end if          // by f_A'
    if e ∈ C_V do          // if the entity is a concept
        for each c ∈ Dom(m_A) do          // for each concept already mapped
            if (f_A'(e), c) ∈ S_A' ⊕ (f_B'(e), m_A(c)) ∈ S_B' do
                flag ← false          // if subsumption is different in A' and B'
            end if          // then the alignment is not conservative
            if (c, f_A'(e)) ∈ S_A' ⊕ (m_A(c), f_B'(e)) ∈ S_B' do
                flag ← false
            end if
        end for
    end if
end for
return flag
```

Complexity Analysis. First, we note that variable assignments, equality checks and logical operations such as exclusive disjunction all have constant time complexity, i.e. O(1). With the right choice of data structure, checking if an element is in a set and inserting a new element also present the same complexity. This is the case if we use a presence-absence array for the concepts and an adjacency matrix for the subsumption relation, for example. For Algorithm 1, we have two nested conditional loops, Loop 1.1 in lines 4-17 and Loop 1.2 in lines 7-14, followed by another loop, Loop 1.3, in lines 20-25. Loop 1.1 runs for n_V iterations, where n_V is the number of concepts in the alignment V. Loop 1.2 runs a single iteration in the first iteration of Loop 1.1, two in the second, and so forth, up to n_V iterations in the last. This sum is the result of the formula $n_V*(n_V+1)/2$. Loop 1.3 runs m_V iterations, where m_V is the number of relations in V. Since many different relations may hold between any pair of concepts in V, m_V may be greater than n_V^2. Every other operation has constant complexity, as previously noted. Therefore, the time complexity of Algorithm 1.1 is $O(n_V^2 + m_V)$. With the data structures we have described, auxiliary space complexity is n_V for the concepts, m_V for the relations and n_V^2 for the adjacency matrix for subsumption relations.

In Algorithm 2, there are two nested loops, Loop 2.1 in lines 4-30 and Loop 2.2 in lines 21-28. Loop 2.1 runs for $n_V + m_V$ iterations, and Loop 2.2 runs an increasing number of iterations up to n_V, but only when the element selected in Loop 2.1 is a concept. Thus, the time complexity is $n_V*(n_V+1)/2 + m_V$, which is bound by $O(n_V^2 + m_V)$, the same time complexity from both calls of Algorithm 1 in lines 1 and 2. We highlight here that the complexity of both algorithms depends only on the size of the alignment, and not of the aligned ontologies, which are usually much larger.

6. Conclusion

We have described a method based on category theory for the detection of conservativity violations in ontology alignments. The foundation in category theory allows the integration of our approach with other techniques and procedures defined in the same mathematical formalism. Further, our proposal allows the detection of both types of conservativity violations, i.e., subsumption conservativity violations as well as equivalence conservativity violations, without requiring the merging of the aligned ontologies, the execution of reasoners, or the assumption of disjointness, all of which lead to great complexity when the input ontologies are large, as discussed in Section 3.

The technique described in this work still needs to be extended to repair the discovered violations, as well as implemented and evaluated against reference datasets, such as the ones provided by the Ontology Alignment Evaluation Initiative [Thiéblin et al. 2018]. The evaluation would allow us to check (1) the number of violations detected, (2) repaired and (3) the execution time of the algorithm. Another aspect that needs to be further investigated is the preservation of conservativity over the alignment operations defined by [Zimmerman et al. 2006]. Intuitively, we expect that the intersection of two conservative alignments should also be conservative – however, this proposition still needs to be proved.

Acknowledgemments

This work was partially funded by brazilian agencies CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) and CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

References

Adámek, J., Herrlich, H., and Strecker, G. E. (1990). *Abtract and concrete categories: The joy of cats*. John Wiley & Sons, Inc., New York.

Bench-Capon, T. and Malcolm, G. (1999). Formalising ontologies and their relations. In *International Conference on Database and Expert Systems Applications*. Springer.

Berners-Lee, T., Hendler, J., and Lassila, O. (2001) The semantic web. *Scientific American*, 284(5):34-43.

Cafezeiro, I. and Haeusler, E. H. (2007). Semantic interoperability via category theory. In *International Conference on Conceptual Modeling*. Australian Computer Society, Inc.

Cafezeiro, I., Haeusler, E. H., and Rademaker, A. (2008). Ontology and Context. In *IEEE International Conference on Pervasive Computing and Communications.* IEEE Computer Society.

Ivanova, V. and Lambrix, P. (2013). A Unified Approach for Aligning Taxonomies and Debugging Taxonomies and Their Alignments. In *Extended Semantic Web Conference*. Springer.

Jiménez-Ruiz, E., Grau, B. C., Horrocks, I. and Berlanga, R. (2011). Logic-based assessment of the compatibility of UMLS ontology sources. *Journal of Biomedical Semantics*. 2(1):S2.

Lambrix, P. and Liu, Q. (2013). Debugging the missing is-a structure within taxonomies networked by partial reference alignments. *Data & Knowledge Engineering*, 86:179-205.

Solimando, A., Jiménez-Ruiz, E. and Guerrini, G. (2014). Detecting and correcting conservativity principle violations in ontology-to-ontology mappings. In *International Semantic Web Conference*. Springer.

Solimando, A., Jiménez-Ruiz, E. and Guerrini, G. (2017). Minimizing conservativity violations in ontology alignments: algorithms and evaluation. *Knowledge and Information Systems*, 51(3):775-819.

Studer, R., Benjamins, V. R., and Fensel, D. (1998). Knowledge engineering: principles and methods. *Data and Knowledge Engineering*, 25(1-2):161-197.

Thiéblin, E., Cheatham, M., Trojahn, C., Zamazal, O. and Zhou, L. (2018). The First Version of the OAEI Complex Alignment Benchmark. In *International Semantic Web Conference*. Springer-Verlag.

Zimmermann, A., Krötzsch, M., Euzenat, J., and Hitzler, P. (2006). Formalizing ontology alignment and its operations with category theory. In *International Conference on Formal Ontology in Information Systems*.