Model-based Merging of Open-Domain Ontologies

Zied Bouraoui, Sébastien Konieczny, Truong-Thanh Ma, Ivan Varzinczak
CRIL, Université d'Artois & CNRS, Lens, France
{bouraoui, konieczny, ma, varzinczak}@cril.fr

***Abstract*—Conceptual knowledge, encoded in ontologies or knowledge graphs, plays an essential role in many areas, including Semantic Web, Information Retrieval, and Natural Language Processing. Considerable attention has recently been devoted to the problem of unifying and linking available ontologies. While the vast majority of existing work focuses on matching or aligning resources, in this paper, we investigate the application of belief merging theory to ontology merging to obtain a unique perspective. We consider the setting where different ontologies share the same terminology (i.e., assuming that they are already mapped to each other). However, they express knowledge in different and potentially conflicting ways. In order to get a unified view of the knowledge conveyed by the different ontologies, we start by providing a semantic-based merging model. Our method retrieves all the interpretations in which the outcome can be found. We support demonstrating the method's effectiveness by an experimental evaluation of the method on existing open-domain ontologies.**

***Index Terms*—Belief Merging, Ontology, Semantic Conflicts, Open-Domain, Description Logics.**

I. Introduction

Structured knowledge about concepts and properties commonly uses in fields such as natural language processing (NLP) [1], and Semantic Web [2]. They are typically encoded using ontologies or knowledge graphs. The critical difference between these two frameworks is that knowledge graphs are considerably less expressive than ontologies, and therefore can be thought of as a more straightforward form of ontologies. [1]

An important point, however, is that many ontologies sharing the same knowledge are available. Several open domain ontologies, such as SUMO, OpenCyc, Wikidata, Babelnet, and others, are available on the Web. Therein, they often express knowledge of a particular domain such as food, sport, dance, and so on, using different terminology. This observation has led to several methods aiming at unifying and link ontologies to each other. Therefore, ontology integration approaches, such as *ontology matching, mapping, and alignment*, have emerged (e.g., [3]–[7]). Alignment methods offer the different correspondences between ontologies. Ontology matching is the process of automatically generating correspondences between terms of different ontologies. Some recent studies, including [3], [8], [9], focus on these aspects.

Naturally, when expressing knowledge of a given domain and assuming that the same terminology (i.e., individual, concept and role names are the same), many points of view are possible. Therefore many types of conflicts may arise. By conflict, they do not only refer to logical inconsistency but also to semantic conflicts that might appear in the knowledge structured in different ways. Consider, for instance, the concept "*Process*" as defined by the following three ontologies: SUMO, WikiData, and BabelNet. i.e., Wikidata: "Procedure" is a "Technique", and "Technique" is a "Process". SUMO: "Technique" is a Procedure, and "Technique" is a "Process". Babelnet: "Technique" and "Procedure" are a "Process". In this example, the three ontologies are already matched to each other. While there is no logical inconsistency between these ontologies, the emerging problem is that they structure the knowledge about the domain in different and potentially conflicting ways. (i.e., WikiData says that "*Procedure*" is a *"Technique"* while SUMO states that "*Technique*" is a "*Procedure*". Then, one issue that occurs is whether the statements of Wikidata or SUMO should be selected in terms of merging). Hence, ontology merging combines two (or more) ontology sources into a target ontology while solving (semantic and logical) conflicts between them. In this paper, we focus on the disputes/conflicts that arise when the ontologies sources express, using the same terminology, the knowledge about a particular domain in different incompatible ways. In the example mentioned above, one possible solution to deal with semantic conflicts would be to consider all the concepts as equivalent (e.g., *"Procedure"* $\equiv$ *"Technique"*). However, we would prefer to avoid this solution as it leads to flattening the hierarchy of the output ontology (i.e., the result of merging), which means lose all the information provided by the ontologies. Recall that the primary purpose of an ontology is to structure conceptual knowledge hierarchically. A solution one can consider the majority, for instance, if five ontologies agree that "*Procedure*" is a *"Technique"* while the only one says that "*Technique*" is a "*Procedure*", then the majority will be followed.

In order to solve those problems, we propose an ontology merging method that relies on the solid theoretical foundations of propositional belief merging (e.g., [10], [11]). Therein, several merging operators have been proposed (e.g., [12]–[15]). Recently, several studies related to ontologies merging have been proposed [16]–[21]. In this paper, we introduce semantic-based ontology-merging operators as an approach to solving conflicts. Notice that we focus crucially on handling the semantic conflicts between sources, and we assume that these sources have already been matched to each other before the merging process conducts.

[1] In this paper, we shall use ontologies to refer to knowledge graphs as well.

2375-0197/20/$31.00 ©2020 IEEE
DOI 10.1109/ICTAI50040.2020.00015

Syntax	Semantics
$C \sqsubseteq D$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
r	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
a	$a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$
$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
$\top$	$\Delta^{\mathcal{I}}$
$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y \in \Delta^{\mathcal{I}} s.t. (x, y) \in r^{\mathcal{I}}, y \in C^{\mathcal{I}}\}$

TABLE I: Syntax and semantics of description logic $\mathcal{EL}$

II. Background

To introduce our method, we choose description logics, as they provide the formal foundations of ontologies. For simplicity, we will consider $\mathcal{EL}$ [22], which is one of the most basic description logics.

Let N_C, N_R, N_I be three pairwise disjoint sets where N_C denotes a set of atomic concepts, N_R denotes a set of atomic relations (roles), and N_I denotes a set of individuals. The $\mathcal{EL}$ concept expressions are built according to the following grammar: $C ::= \top \mid N_C \mid C \sqcap C \mid \exists r.C$ where $r \in N_R$. An $\mathcal{EL}$ ontology (a.k.a. knowledge base) consists of a set of general concept inclusion (GCI) axioms of the form $C \sqsubseteq D$. Furthermore, a set of equivalence axioms of the form $C \equiv D$ means that the two general concept inclusions $C \sqsubseteq D$ and $D \sqsubseteq C$ hold. There is also a set of concept assertions of the form $C(a)$, and a set of role assertions of the form $r(a, b)$. In this paper, we consider assertion free ontologies. Moreover, given $A, B, A_1, A_2 \in N_C$, an $\mathcal{EL}$ TBox $\mathcal{T}$ is in the **normal form** [23], [24] if it consists of inclusions of the form: $A \sqsubseteq B, A_1 \sqcap A_2 \sqsubseteq B, A \sqsubseteq \exists r.B, \exists r.A \sqsubseteq B$. We assume that all ontologies in this paper are in normal form.

The semantics is in terms of interpretations $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ which consist of a non-empty interpretation domain $\Delta^{\mathcal{I}}$ and an interpretation function $\cdot^{\mathcal{I}}$ that maps each individual $a^{\mathcal{I}} \in N_I$ into an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, each concept $A \in N_C$ into a subset $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and each role $r \in N_R$ into a subset $r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. Table 1 summarizes the syntax and semantics of $\mathcal{EL}$. An interpretation $\mathcal{I}$ is said to be a model of (or satisfies) a GCI axiom, denoted by $\mathcal{I} \models C \sqsubseteq D$, if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. Similarly, $\mathcal{I}$ satisfies a concept (resp. role) assertion, denoted $\mathcal{I} \models C(a)$ (resp. $\mathcal{I} \models r(a, b)$), if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ (resp. $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$). An interpretation $\mathcal{I}$ is a model of an ontology $\mathcal{O}$ if it satisfies all the axioms in $\mathcal{O}$. An ontology is said to be consistent if it has a model. Otherwise, it is inconsistent. In this work, for simplicity, we assume that every ontology is consistent. An axiom Φ is entailed by an ontology, denoted by $\mathcal{O} \models \Phi$, if Φ is satisfied by every model of $\mathcal{O}$.

III. Semantic-Based Ontology Merging

Given a profile of ontologies describing a particular domain knowledge using the same signature (same concept, role, and individual names), one can distinguish two possible cases: (1) the case where the ontologies **agree** on the same claims (e.g., all ontologies claim that a concept A is subsumed by B) and (2) the other case where there is **a disagreement** (or conflict) on how to express knowledge.

In the first case, there is no conflict, and the result of merging should contain the common statements. In the second case, a solution that appropriately solves the conflicts needs to find all possible statements and select the most plausible one, in which the different sources agree. This case could be in the form of semantic disagreement (e.g., "*Technique*" is subsumed by "*Procedure*" in SUMO ontology whereas Wikidata state that "*Procedure*" is subsumed by "*Technique*").

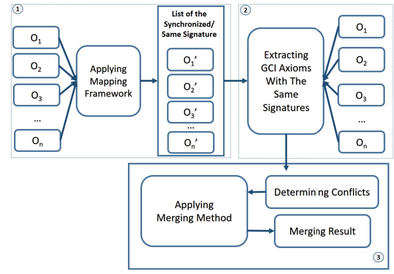

Fig. 1: A general framework of the merging process

Definition 1 (Semantic Conflict): *Let $\{\mathcal{O}_1, \ldots, \mathcal{O}_n\}$ be a profile of ontologies that share the same signature. Let $A \sqsubseteq B$ be an axiom of $\mathcal{O}_i$ ($\mathcal{O}_i \models A \sqsubseteq B$). We say that $A \sqsubseteq B$ is in a semantic conflict if $\mathcal{O}_i \not\models B \sqsubseteq A$ and there is an ontology $\mathcal{O}_j$, s.t. $\mathcal{O}_j \models B \sqsubseteq A$.*

In this paper, we follow a semantic (model-based) approach for merging, by first computing the set of statements which are close to the input ontologies. When this result is obtained, we compute from the set of selected statements an ontology that gives us the merging result. Intuitively, given the signatures of the different ontologies, we generate all possible statements, which will be encoded using a set of interpretations. Once these interpretations are found, we select the most plausible interpretations that agree with the sources. We propose an ontology merging framework to outline the merging process and deal with the existing conflicts. Our merging framework is summarized in Figure 1. The framework splits into three main parts: (1) applying an existing mapping process to generate a list of synchronized names. (2) extracting the axioms with synchronized names *(i.e., rewrite all ontologies with the same signature)* from the sources. (3) applying an ontology merging method. The steps of the third part are the following: (i) we enumerate all possible ways to express knowledge (Intuitively one possible way will be the result of merging). (ii) We express these possible ways semantically in terms of interpretations. (iii) We apply merging theory to measure the satisfaction of each solution w.r.t. the ontology profile. In the following, we will describe the merging steps in detail.

A. Generating Possible Merging Solutions

Given the set of axioms, we first rewrite them in the normal form [22]. Then, these rules are rewritten by a combination between the relations and the set of concept names, called *patterns*.[2] We define the patterns as follows:

[2]We can see these patterns as second-order is-a relations, whose instances are the concepts from the ontologies.

Definition 2 (Pattern): *Let the symbols $\star, \diamond$ be the representative placeholders of the concepts names. We will use π to denote a pattern. The patterns are as follows: subsumption ($\star \sqsubseteq \diamond$), intersection ($\star \sqcap \diamond$), existence ($\star \sqsubseteq \exists r.\diamond$, $\exists r.\star \sqsubseteq \diamond$). The set of patterns is denoted as ξ.*

In order to have an axiom, we collect concept pairs from the set of concepts defined as follows:

Definition 3 (Concept Pair): *Let N_C be the set of concept names. A pair of concept names is an element of $N_C \times N_C$. For brevity, we shall often write a pair (A, A') as AA'. The set of concept name pairs is defined as N_C^2.*

We use Definition 3 to generate the concept names for the combination per each pattern. In other words, let $(A, B) \in N_C^2$ and π be a pattern. The instantiation of π with AB, denoted π_{AB}, is the result of replacing $\star$ with A and $\diamond$ with B in π. For example, given the pattern π = ($\star \sqsubseteq \exists r.\diamond$) and the concept name pair AB, then π_{AB} = $A \sqsubseteq \exists r.B$. The replacement of the concept names with the placeholders in the patterns is called "a concept pattern". The set of concept patterns is defined as:

Definition 4 (Concept Pattern): *Let ξ be the set of patterns and let N_C^2 be the set of concept name pairs. Let $\times$ be a combination between the patterns and the concept name pairs. The set of concept patterns is $\alpha = N_C^2 \times \xi$.*

Now, we combine the concept patterns called a *combination*. These *"combinations"* will allow us to generate all possible ways to express knowledge.

Definition 5 (Combination): *Let α be the set of concept patterns. The set of combinations is $\mathcal{H} = \bigcup_{2 \leq n \leq |\alpha|} \{X \mid X \subseteq \alpha \text{ and } |X| = n\}$.*

We use Definitions 4 and 5 to generate all the possible combinations that can be expressed using the same terminology. Next, we need to generate the interpretations corresponding to the combination. We first introduce a closure definition which is also the foundation of collecting all possible interpretations. Let $\mathcal{I}$=$\langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ with $\mathcal{A}_{\mathcal{I}}$=$\{A(a) \mid a \in N_I \text{ and } \mathcal{I} \models A(a)\} \cup \{r(a,b) \mid a, b \in N_I \text{ and } \mathcal{I} \models r(a,b)\}$. We denote $\mathcal{A}_{\mathcal{I}}$ as the pseudo-ABox induced by $\mathcal{I}$. The closure definition is as:

Definition 6 (Closure): *Let Φ be an axiom, B be a concept, r be a role and $\{a, b\}$ be individuals. The closure of Φ is defined as follows: $\mathcal{C}l_{\Phi}(\mathcal{A}_{\mathcal{I}})$=$\{B(a) \mid \langle \Phi, \mathcal{A}_{\mathcal{I}} \rangle \models B(a)\} \cup \{r(a,b) \mid \langle \Phi, \mathcal{A}_{\mathcal{I}} \rangle \models r(a,b)\}$.*

An illustration to explain the closure is as follows: assuming we have an axioms Φ including $A \sqsubseteq B$, and $\mathcal{S}$=$\{A^{\mathcal{I}}= \{a\}, B^{\mathcal{I}}= \{b\}, C^{\mathcal{I}}= \{c\}\}$, then the closure of Φ is $\mathcal{C}l_{\Phi}(\mathcal{S})$= $\{A^{\mathcal{I}}$=$\{a\}$, $B^{\mathcal{I}}$= $\{a, b\}$, $C^{\mathcal{I}}$= $\{c\}\}$.

Definition 7: *Let $\mathcal{H}$ be a set of combinations and $\mathcal{A}_{\mathcal{I}}$ be the pseudo-ABox. The closure of $\Phi_i \in \mathcal{H}$ is denoted as $\mathcal{C}l_{\Phi_i}(\mathcal{A}_{\mathcal{I}})$. The universe of interpretations is $\mathcal{U} = \bigcup_{1 \leq i \leq |\mathcal{H}|} \mathcal{C}l_{\Phi_i}(\mathcal{A}_{\mathcal{I}})$.*

Consider Φ_1= $\{C \sqsubseteq B, C \sqsubseteq A\}$, then the closure of Φ_1 is $\mathcal{C}l_{\Phi_1}(\mathcal{A}_{\mathcal{I}})$=$\{A^{\mathcal{I}}$=$\{a, c\}$,$B^{\mathcal{I}}$=$\{b, c\}$,$C^{\mathcal{I}}$=$\{c\}\}$. Finally, we aggregate them (using Definitions 6 and 7) to collect the set of interpretations.

Regarding the restrictions, there are three constraint cases as follows: (1) the interpretation does not contain the empty set. (2) each concept has at least one representative (or typical) individual that belongs to it. (3) the interpretation is not duplicated. We focus crucially on generating the interpretations in which each concept will store completely individuals, e.g., $\mathcal{U}$=$\{A^{\mathcal{I}}$=$\{a\}$,$B^{\mathcal{I}}$=$\{a, b, c\}$,$C^{\mathcal{I}}$=$\{a, c\}\}$. More formally, we call the interpretations that are restricted by the constraints as "spurious" interpretations.

Definition 8: *Let $\mathcal{I}$ be an interpretation. We say $\mathcal{I}$ is spurious if one of the following three conditions holds:*

- *There is $A \in N_C$ s.t. $A^{\mathcal{I}} = \emptyset$; or*
- *There is $a \in N_I$ s.t. for every $A \in N_C, a^{\mathcal{I}} \notin A^{\mathcal{I}}$; or*
- *There is $\mathcal{I}'$ and $\mathcal{I}' \neq \mathcal{I}$ s.t. $\Delta^{\mathcal{I}}$=$\Delta^{\mathcal{I}'}$ and $\cdot^{\mathcal{I}} = \cdot^{\mathcal{I}'}$.*

With $\mathcal{SP}$ we denote the set of all spurious interpretations.

Finally, we generate "useful" interpretations by eliminating the "spurious" interpretations from the set of "universal" interpretations. The set of useful interpretations is defined as:

Definition 9: *Let $\mathcal{U}$ be the set of universal interpretations and $\mathcal{SP}$ be the set of spurious interpretations. The set of useful interpretations is defined as $\mathcal{W} = \mathcal{U} \setminus \mathcal{SP}$.*

Example 1: *We consider an example of a profile $\odot_1$ of three ontologies $\{\mathcal{O}_1, \mathcal{O}_2, \mathcal{O}_3\}$ in which three concepts A, B, C have the same names ($A, B, C \in \mathcal{O}_{1,2,3}$): $\odot_1 := \{ \mathcal{O}_1 : C \sqsubseteq B, A \sqsubseteq B, \mathcal{O}_2 : A \sqsubseteq B, B \equiv C, \mathcal{O}_3 : A \sqsubseteq C \sqcap B \}$. First of all, the combination between the concept names $N_C^2 = \{A, B, C\}$ and six forms (corresponding to the concept patterns) will generate $6 \times |N_C^2| = 6 \times 3 = 18$ patterns. Next step, we apply Definition 4 to collect the set of combinations. (e.g., $\{A \sqsubseteq B, A \sqcap C\}$, $\{A \sqsubseteq C, \exists r.C \sqsubseteq B\}$)). Next, all interpretations are also generated by the initial interpretation ($A^{\mathcal{I}}$=$\{a\}$, $B^{\mathcal{I}}$=$\{b\}$, $C^{\mathcal{I}}$=$\{c\}$, $r^{\mathcal{I}}$=(a, b)) and the set of combinations $\mathcal{H}$. Applying Definition 7, the total number of collected interpretations are ($\Sigma_{k \in \mathcal{H}} |(N_{I)_k}|$)=104 interpretations.*

B. Selection of the Best Interpretation

In this section, we provide ontology merging operators. The aim is to seek the interpretations that are the best representatives of the sources ontologies. First of all, let $\odot$ be a profile of ontologies sources ($\odot = \{\mathcal{O}_1, ..., \mathcal{O}_n\}$). Let $\oplus$ be an ontology merging operator that assigns each ontology set $\odot$ to a set of selected interpretations, denoted by $\oplus(\odot)$. A model-based operator is defined by selecting the closest interpretations to the ontology sources. First, we compute the distance between the interpretation $\mathcal{I}$ and the ontology $\mathcal{O}$ through axioms $\Phi_i \in \mathcal{O}$ as follows:

Definition 10: *Let $\mathcal{I}$ be an $\mathcal{EL}$ interpretation and let $\mathcal{O}$ be an ontology. The distance between $\mathcal{I}$ and $\mathcal{O}$ is defined as: $dist(\mathcal{I}, \mathcal{O}) = |\{\Phi_i \in \mathcal{O} \mid \mathcal{I} \not\models \Phi_i\}|$*

The distance from $\mathcal{I}$ to $\mathcal{O}$ is the number of axioms in $\mathcal{O}$ that are not satisfied by $\mathcal{I}$. We call *dissatisfaction* as the disagreement of $\mathcal{I}$ with $\mathcal{O}$. In this work, the disagreement will be computed particularly at axiom level. Notice that, when calculating the distance of an interpretation to the ontology $\mathcal{O}$, we consider both explicit and implicit knowledge. Namely, if only explicit knowledge is considered, then the distance of an interpretation to two equivalent ontologies could be different.

Algorithm 1 SIF algorithm

```
Input in IntpList=[]: List of Interpretations (the closest distance)
Output in SIF={}
 1: Let OneI ← ∅
 2: for each concept, indivList : IntpList do
 3:     N ← getConceptName(concept)
 4:     OneInpt[N] ← OneInpt[N] ∪ getIndividuals(indivList)
 5: end for
 6: for each nameConcept, valueList : OneInpt do
 7:     F ← Frequency(valueList)
 8:     T ← size(IntpList)/2
 9:     for each i, v : F do
10:         if v > T then
11:             SIF[nameConcept] ← SIF[nameConcept] ∪ i
12:         end if
13:     end for
14: end for
15: return SIF
```

Consider the following example where $\mathcal{O}_1$=$\{A \sqsubseteq B, B \sqsubseteq C\}$ and $\mathcal{O}_2 = \{A \sqsubseteq B, B \sqsubseteq C, A \sqsubseteq C\}$. It is clear that the two ontologies are equivalent, but $\mathcal{O}_1$ contains fewer axioms than $\mathcal{O}_2$. Given an interpretation $\mathcal{I} = \{A^{\mathcal{I}} = \{a, b, c\}, \{B^{\mathcal{I}} = \{b\}, \{C^{\mathcal{I}} = \{c\}\}$, then $dist(\mathcal{I}, \mathcal{O}_1) = dist(\mathcal{I}, \mathcal{O}_2) = 2$ when considering the closure, but it is not the case if we only consider explicit knowledge (i.e., $dist(\mathcal{I}, \mathcal{O}_1) = 2$ is different from $dist(\mathcal{I}, \mathcal{O}_2) = 3$).

Next, ontology merging operators are based on the aggregation of these distances. The idea is to find the closest information to the overall profile. Suppose that we have an aggregation function g [25], then we define:

Definition 11 *($\oplus_g$): Let $\mathcal{I}$ be an interpretation, let $\odot$ be a set of ontologies, and g be an aggregation function. We define the distance $dist_g(\mathcal{I}, \odot) = g_{\mathcal{O} \in \odot}(dist(\mathcal{I}, \mathcal{O}))$.*

In this paper, we will focus on the aggregation functions $g \in \{\max, \Sigma, leximax\}$[3], that echoes the most usual propositional merging operators. We have an assignment that maps each knowledge base set $\odot$ to a pre-order $\leq_\odot$ over interpretations (on $\mathcal{W}$), referring to [14]. And finally, we define our ontology merging operators $\oplus_g$ as follows:

Definition 12: *Let $\odot$ be a profile of ontologies, g be an aggregation function, and $\mathcal{I}, \mathcal{J} \in \mathcal{W}$. We define the ontology merging operator $\oplus_g$ as*

$$\mathcal{I} \leq^g_\odot \mathcal{J} \text{ iff } dist_g(\mathcal{I}, \odot) \leq dist_g(\mathcal{J}, \odot)$$

$$Mod(\oplus_g(\odot)) = \min(\mathcal{W}, \leq^g_\odot)$$

Example 2: *Let us continue with example 1 for illustration. Assuming we have the interpretation $\mathcal{I}_1 = \{A^{\mathcal{I}_1} = \{a, b, c\}, B^{\mathcal{I}_1} = \{b\}, C^{\mathcal{I}_1} = \{b, c\}, r^{\mathcal{I}_1} = (a, b)\}$. Computing the distance from $\mathcal{I}$ to $\odot_1$ as follows: $dist(\mathcal{I}_1, \mathcal{O}_1) = d^{\mathcal{I}_1}_{\mathcal{O}_1}$ = 2 because an axiom of $\mathcal{O}_1$ is unsatisfied by $\mathcal{I}_1$ (namely $\mathcal{I}_1 \not\models A \sqsubseteq B$ and $\mathcal{I}_1 \not\models C \sqsubseteq B$); similarly, $d^{\mathcal{I}_1}_{\mathcal{O}_2}$ = 2, since for $\mathcal{I}_1 \not\models A \sqsubseteq B$ and $\mathcal{I}_1 \not\models B \equiv C$; $d^{\mathcal{I}_1}_{\mathcal{O}_3}$ = 1 due to $\mathcal{I}_1 \not\models A \sqsubseteq C \sqcap B$. Accordingly, we have the distance between $\mathcal{I}$ and $\odot_1$ by applying Definitions 11 and 12 as follows: $dist_{\max}(\mathcal{I}_1, \odot_1)$= $max(d^{\mathcal{I}_1}_{\mathcal{O}_1}, d^{\mathcal{I}_1}_{\mathcal{O}_2}, d^{\mathcal{I}_1}_{\mathcal{O}_3})$= $max(2, 2, 1)$= 2, $dist_\Sigma(\mathcal{I}_1, \odot_1) = \sum_{i=1}^{3}(d^{\mathcal{I}_1}_{\mathcal{O}_i})$ = (2+2+1) = 5, $dist_{Gmax}(\mathcal{I}_1, \odot_1)$=$(2, 2, 1)$. For this example, there are three selected closest interpretations: $Mod(\oplus_g(\odot_1))$= $\{\{A^{\mathcal{I}}$=$\{a\}$, $B^{\mathcal{I}}$=$\{a, b, c\}$, $C^{\mathcal{I}}$= $\{a, b, c\}\}; \{A^{\mathcal{I}}$= $\{a, b\}$, $B^{\mathcal{I}}$= $\{a, b, c\}$, $C^{\mathcal{I}}$= $\{a, b, c\}\}; \{A^{\mathcal{I}}$= $\{a, c\}$, $B^{\mathcal{I}}$= $\{a, b, c\}$, $C^{\mathcal{I}}$= $\{a, b, c\}\}$.*

Dataset	Type	Number of Concept	Number of Axiom	Number of is-a relation
SUMO	(1)	4558	587842	5330
	(2)	3432	2138	2138
Wikidata	(1)	69188843	2941036	2941036
	(2)	119152	16876	16876
Babelnet	(1)	6113467	277036611	15831054
	(2)	119957	165121	165121

TABLE II: The number of Concepts, Axioms, i-a Relations of three ontologies sources. (1) is the number of the original ontology, (2) is the number after the mapping process.

C. Expressing the Result of Merging

Once the set of closest interpretations from the profile is computed, we build an ontology that corresponds to these interpretations. In fact, we need only one interpretation to represent the outcome of merging since we can not model disjunction in this framework. Hence, we built a *Selected Interpretation Frequency* (SIF) algorithm based on the frequency of individuals of each concept to decide the selection. The algorithm's idea is to collect/put all individuals *(of the selected interpretations)* into one interpretation. Next, we compute the frequency of individuals using the majority technique to select the final result. An individual is selected if it is in the majority of the interpretations. Notice that we only consider assertions free open-domain ontologies, and we only focus on merging terminological knowledge. A pseudo-ABox is then assumed in this paper. Notably, when there is a real ABox, it could certainly help us to solve the semantic conflict and to choose between the candidate interpretations, which is the good one. This improvement of the framework is left for future work. However, in the next sections, we will show with the experiments that even without this real ABox, we obtain very sensible results.

Definition 13: *$f_\oplus$ is called a full ontology merging operator if $\oplus$ is an ontology merging operator and $f_\oplus(\odot) = SIF(Mod(\oplus_g(\odot)))$.*

The input of the SIF algorithm is the list of closest interpretations, and the output is the most plausible interpretation. Explaining as follows: assume that there are the individuals in each concept including $a \in A$, $b \in B$, $c \in C$.[4] Firstly, *(continue with Example 2)* we collect all the individuals of interpretations *(line 2 to 5)*, the outcome is $\{A$:$\{a, a, a, b, c\}$, B:$\{a, a, a, b, b, b, c, c, c\}$, C:$\{a, a, a, b, b, b, c, c, c\}\}$. Secondly, we computed the frequency of the individuals in each concept, the result includes: $\{A^{\mathcal{I}}$=$\{a$:3$, b$:1$, c$:1$\}$, $B^{\mathcal{I}}$=$\{a$:3$, b$:3$, c$:3$\}$, $C^{\mathcal{I}}$=$\{a$:3$, b$:3$, c$:3$\}\}$ *(line 7)*. The purpose of the next step is to select a unique interpretation. Hence we use simple majority voting (line 8) to filter out the individuals in each concept and select the most plausible interpretation (lines 9 to 13).

[3]We will write $Gmax$ instead of $leximax$ like in [11].

[4]All interpretations contain these individuals.

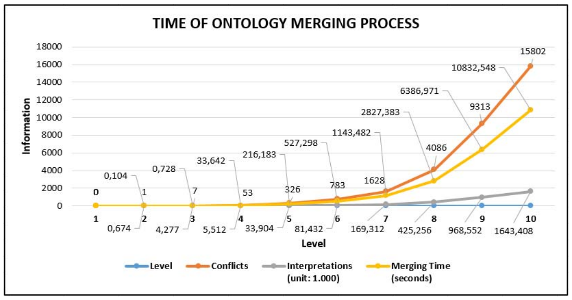

Fig. 2: The computation time of the merging process (seconds).

In the case of the example $\odot_1$, the selected result using the SIF algorithm is as follows: $f_{\oplus}(\odot_1)$= $\{A^{\mathcal{I}}$=$\{a\}$, $B^{\mathcal{I}}$=$\{a, b, c\}$, $C^{\mathcal{I}}$=$\{a, b, c\}\}$. In the following, we show how to translate the chosen interpretation into an ontology:

Definition 14: *Let $\mathcal{H}$ be the set of combinations (axioms) and $f_{\oplus}$ be the full ontology merging operator. The syntactic ontology merging result is defined as $\mathcal{Y} = \{M \mid M \in \mathcal{H}$ and $f_{\oplus}(\odot) \models M\}$*

Applying Definition 14, we seek an ontology M in the combinations $\mathcal{H}$, such that M is satisfied by $f_{\oplus}(\odot)$. In other words, one interpretation $f_{\oplus}(\odot)$ corresponds to one ontology M. Therefore, the syntactic ontology merging outcome of the example $\odot_1$ is: $\mathcal{Y} = \{A \sqsubseteq C, B \sqsubseteq C\}$.

IV. Experimental Evaluation

This section experimentally evaluates our approach. First of all, we collected the concepts from the three ontologies *(SUMO*[5]*, Wikidata*[6]*, and Babelnet*[7]*)*. We provide statistics of the ontologies before and after the mapping process in Table II. We use the mapping process between the three sources to seek correspondences between them. We create a synchronization of the new names of concepts based on those mappings.

The mapping process includes the following steps: At step 1, for each concept name in SUMO, we extract its WordNet given in the annotation of the concept. For step 2, we implement two methods to collect Wikidata ID: (1) First, we use WordNet ID *(extracted at step 1)* to determine the synset of Babelnet. After that, we use BabelNet API to retrieve the Wikidata ID. *(The correspondences between Wikidata and Babelnet are available in the BabelNet dataset)*. (2) The second one uses the SUMO concept name to seek Wikidata ID through BabelSynset. In step 3, we check and collect the label/name of Wikidata ID through the Qwikidata library. This work is to ensure that the concept names are the same.[8] Notably, a synchronization process of the concept names is conducted before merging. Namely, we assign a new name at the end position of each mapping. The synchronization is essential because the concept names are not the same in practice.

[5]http://www.adampease.org/OP.

[6]https://www.wikidata.org.

[7]https://babelnet.org.

[8]The datasets and the implementation of the merging operator are available at https://github.com/ontologymerging/beliefmerging.

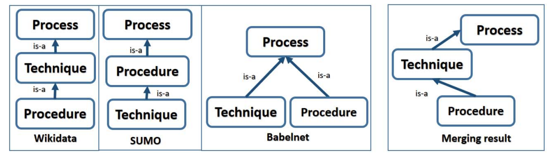

Fig. 3: Structure of concepts in Example 3

For the evaluation of our merging method, we consider the top-10-levels of the three ontologies. The purpose of splitting the considered levels is to evaluate the computation time needed to generate the combinations/interpretations as well as to determine the number of conflicts at each level. For each level, we output the number of axioms, conflicts and computed interpretations. The experimental results are provided in Figure 2, where we show the time needed to compute the result of merging. Overall we can see that our model of merging is fully efficient in solving conflicts. We focus on the axioms that express knowledge in different ways. Next, we compute the interpretations needed to determine the result of merging. Note that all concepts are handled based on the synchronized names between the three sources. We have run an experiment in order to test our proposal based on the number of conflicts and the processing time. The conflicts start to appear in level 2 with 783859 concepts. Subsequently, the number of contradictions increases when increasing the number of levels. The time of processing is fast and effective. A piece of evidence is at level 10 with 15802 conflicts, and the processing time is around 180,54 minutes (10832,55 seconds). Also, the result of merging is quite expected because it solves the problem of heterogeneous and ambiguous information well. The most plausible interpretation satisfied all the axioms and obtained the information entirely from the three ontologies.

As a matter of illustration, we show in the following a real example, and how our method performs merging.

Example 3: *Consider the three concepts ($Procedure$, $Technique$, $Process$) represented as in Figure 3. This example presented in the introduction section.*

The merging upshot is: $Mod(\oplus(\odot_{Ex3}))$= $\{Procedure^{\mathcal{I}}$= $\{proced, tech\}$, $Techniques^{\mathcal{I}}$= $\{tech, proced\}$, $Process^{\mathcal{I}}$= $\{tech, proced, procce\}$ $\}$. The syntactic result is the same as Wikidata because the expressing knowledge of Wikidata contained and satisfied the semantic merging result well.

V. Concluding Remarks

We deal with merging problems existing when conflicts arise between knowledge coming from multiple open-domain ontology sources. To this end, we provided a formal definition of a semantic-based merging operator. We then demonstrated our approach using real examples of open-domain ontologies. A framework of ontology merging and how the mapping between different sources is developed and made available online. Finally, we also provide experimental results as well as the practical example to demonstrate our method.

ACKNOWLEDGMENTS

This work was supported by the ANR CHAIRE IA BE4musIA: BElief change FOR better MUlti-Source Information Analysis.

REFERENCES

[1] R. Navigli, "Natural language understanding: Instructions for (present and future) use," in *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI'18)*, 07 2018, pp. 5697–5702.

[2] R. Gonçalves, M. Horridge, R. Li, Y. Liu, M. Musen, C. Nyulas, E. Obamos, D. Shrouty, and D. Temple, *Use of OWL and Semantic Web Technologies at Pinterest*. Proceedings of the International Semantic Web Conference (ISWC'19), 10 2019, pp. 418–435.

[3] J. Euzenat, "Interaction-based ontology alignment repair with expansion and relaxation," in *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI'17)*. Melbourne, Australia: AAAI Press, Aug. 2017, pp. 185–191, euzenat2017a.

[4] J. da Silva, K. Revoredo, F. Araujo Baião, and J. Euzenat, "Interactive ontology matching: using expert feedback to select attribute mappings," in *Proceedings of the International Semantic Web Conference (ISWC'18)*, Oct. 2018, pp. 25–36.

[5] M. Zhao and S. Zhang, "FCA-Map results for OAEI 2016," in *Proceedings of the International Workshop on Ontology Matching collocated with the International Semantic Web Conference (OM@ISWC'16)*, 2016, pp. 172–177.

[6] A. Laadhar, F. Ghozzi, I. Megdiche, F. Ravat, O. Teste, and F. Gargouri, "POMap results for OAEI 2017," in *Proceedings of the International Workshop on Ontology Matching collocated with the International Semantic Web Conference (OM@ISWC'17)*, 2017, pp. 1–7.

[7] É. Thiéblin, O. Haemmerlé, and C. Trojahn, "CANARD complex matching system," in *Proceedings of the International Workshop on Ontology Matching co-located with the International Semantic Web Conference (OM@ISWC 2018)*, 10 2018, pp. 138–143.

[8] P. Chocron and M. Schorlemmer, "Attuning ontology alignments to semantically heterogeneous multi-agent interactions," in *Proceedings of the European Conference on Artificial Intelligence (ECAI'16)*, NLD, 2016, p. 871–879.

[9] M. Pavão and C. Pesquita, "Complex matching for multiple ontologies: an exploratory study," in *the Proceedings of International Workshop on Ontology Matching co-located with the International Semantic Web Conference (OM@ISWC'18)*, 2018, pp. 222–223.

[10] P. Revesz, "On the semantics of arbitration," *International Journal of Algebra and Computation*, vol. 07, 05 1997.

[11] S. Konieczny and R. P. Pérez, "Merging information under constraints: a logical framework," *Journal of Logic and computation*, vol. 12, no. 5, pp. 773–808, 2002.

[12] S. Konieczny, "On the difference between merging knowledge bases and combining them," in *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR'00)*, 01 2000, pp. 135–144.

[13] T. A. Meyer, A. K. Ghose, and S. Chopra, "Syntactic representations of semantic merging operations," in *the Pacific Rim International Conferences on Artificial Intelligence (PRICAI'02)*, 2002, pp. 620–620.

[14] S. Konieczny and R. P. Pérez, "Logic based merging," *Journal of Philosophical Logic*, vol. 40, no. 2, pp. 239–270, 2011.

[15] A. Haret and S. Woltran, "Belief revision operators with varying attitudes towards initial beliefs," in *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI'09)*, 2019, pp. 1726–1733.

[16] S. Gómez and G. Simari, "Merging of ontologies using belief revision and defeasible logic programming," *the journal of Revista Iberoamericana de Inteligencia Artificial*, vol. 16, pp. 16–28, 01 2013.

[17] D. Porello and U. Endriss, "Ontology merging as social choice," in *Computational Logic in Multi-Agent Systems*. Springer Berlin Heidelberg, 2011, pp. 157–170.

[18] C. R. Rangel, J. Altamiranda, M. Cerrada, and J. Aguilar, "Procedure based on semantic similarity for merging ontologies by non-redundant knowledge enrichment," *the International Journal of Knowledge Management*, vol. 14, no. 2, p. 16–36, Apr. 2018.

[19] S. Raunich and E. Rahm, "Atom: Automatic target-driven ontology merging," in *Proceedings of the 27th International Conference on Data Engineering (ICDE'11)*, 05 2011, pp. 1276 – 1279.

[20] D. Dou, "Ontology translation by ontology merging and automated reasoning," Ph.D. dissertation, Yale University, USA, 2004.

[21] O. Verhodubs and J. Grundspenkis, "Ontology merging in the context of a semantic web expert system," in *the International Conference on Knowledge Engineering and Semantic Web (KESW'13)*, 10 2013, pp. 191–201.

[22] F. Baader, S. Brandt, and C. Lutz, "Pushing the el envelope," in *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI'05)*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005, p. 364–369.

[23] F. Baader, R. Peñaloza, and B. Suntisrivaraporn, "Pinpointing in the description logic $\mathcal{EL}$," in *Proceedings of the German Conference on Artificial Intelligence (KI'2007)*. Springer Berlin Heidelberg, 2007, pp. 52–67.

[24] B. Suntisrivaraporn, *Polynomial-Time Reasoning Support for Design and Maintenance of Large-Scale Biomedical Ontologies*. Ph.D. thesis. Technical University of Dresden, 06 2009.

[25] P. Everaere, S. Konieczny, and P. Marquis, "On egalitarian belief merging," in *Proceedings of the International Conference on Principles of Knowledge Representation and Reasoning (KR'14)*, 07 2014, pp. 121–130.