Undefined 1 (2015) 1–5
IOS Press

Pushing the Limits of OWL 2 Reasoners in Ontology Alignment Repair Problems

Alessandro Solimando [a,*], Ernesto Jiménez-Ruiz [b] and Giovanna Guerrini [a]
[a] *Dipartimento di Informatica, Bioingegneria, Robotica e Ingegneria dei Sistemi, Università di Genova, Via Dodecaneso 35, 16146, Italy*
E-mail: alessandro.solimando@unige.it, giovanna.guerrini@unige.it
[b] *Department of Computer Science, University of Oxford, Wolfson Building, Parks Road, Oxford OX1 3QD, UK*
E-mail: ernesto.jimenez-ruiz@cs.ox.ac.uk

Abstract. Ontologies play a key role in the development of the Semantic Web and are being used in many diverse application domains such as biomedicine and e-commerce. An application domain may have been modeled according to different points of view and purposes. This situation usually leads to the development of different ontologies that intuitively overlap, but that use different naming and modeling conventions. The problem of (semi-)automatically integrating independently developed ontologies through mappings, is usually referred to as the ontology matching problem. Ontology matching systems, however, rely on lexical and structural heuristics, and the integration of the input ontologies and the mappings may lead to many undesired logical consequences, which could sensibly diminish their usefulness. The present paper, on the one hand aims at veryfing the hypothesis that classification of large ontologies via mappings still poses a challenge to OWL 2 reasoners. On the other it also explores the applicability of OWL 2 reasoning for the repair of unintended entailments (namely, unsatisfiable concepts or violations of the conservativity principle). In this paper we provide an update on the feasibility of using OWL 2 reasoners to repair the integration of ontologies via mappings, providing a more accurate evaluation of the feasibility of extracting all the justifications. Additionally, the current evaluation also encompasses the analysis of the use of OWL 2 reasoners for solving the violations of the so-called conservativity principle.

Keywords: Reasoning, Ontology Matching, Ontology Alignment Debugging, Ontology-based Data Integration

1. Introduction

The problem of (semi-)automatically computing mappings between independently developed ontologies is usually referred to as the ontology matching problem. A number of sophisticated ontology matching systems have been developed in the last years [8,29]. Ontology matching systems, however, rely on lexical and structural heuristics and the integration of the input ontologies and the mappings may lead to many undesired logical consequences (*e.g.*, unsatisfiable classes or violations of the conservativity principle).

The fix of undesired logical consequences caused by ontology mappings is known as the mapping repair problem [14]. Mapping repair can be addressed using state-of-the-art approaches for debugging OWL 2 ontologies, which rely on the extraction of justifications for the unwanted axiom (*e.g.*, [13,15,27,34]). However, in [12] it was pointed out that justification-based technologies do not scale when the number of such axioms is large (a typical scenario in mapping repair problems).

This paper extends our previous evaluations presented in [12,32], in different respects. [12] provided a first evaluation on the use of OWL 2 reasoning for

*Corresponding author. E-mail: alessandro.solimando@unige.it Tel: +39-010-353-6616. Fax: +39-010-353-6699.

the classification and repair of integrated ontologies. [32] also considered reasoners for OWL 2 profiles (*i.e.*, *ELK*), and tested the extraction of more than a single justification for computing a repair for logical violations represented by incoherent classes.

Our extended evaluation is based on the datasets and ontology matching systems from the Ontology Alignment Evaluation Initiative (OAEI) [8]. In addition to the previous versions of the evaluation, in this paper we also: *(i)* consider the so-called violations of the conservativity principle, that is, novel axioms entailed by the aligned ontology, involving elements of one of the two input ontologies, that are not entailed by the input ontologies in isolation [1,31]. *(ii)* provide extended experimental results concerning the extraction of (a subset of) all the justifications for a given entailment, providing additional insights on the problem. *(iii)* compare the black-box justification extraction techniques with one of the latest glass-box approaches based on tracing, for the optimization of the justifications extraction.

Our results suggest that the classification of the integration of large ontologies via mappings still poses a challenge to OWL 2 reasoners. Furthermore, the repair of unintended entailments (*e.g.*, unsatisfiable concepts or conservativity violations) using OWL 2 reasoners critically compromises the performance of mapping repair systems in the best case, or it is simply not tractable when all the justifications need to be extracted in order to compute an optimal repair.

The remainder of the paper is organised as follows: Section 2 introduces the needed preliminaries, Section 3 describes the dataset, the environment used for the evaluation, and also provides and discusses in detail its results. Finally, Section 4 concludes the paper.

2. Preliminaries

In this section, we present the formal representation of ontology mappings (Section 2.1), the notions of semantic difference (Section 2.2), mapping coherence and conservativity principle violations (Section 2.3).

2.1. Representation of Ontology Mappings

Mappings are conceptualised as 4-tuples of the form $\langle e_1, e_2, n, \rho \rangle$, where e_1, e_2 are entities in the vocabulary or signature of the relevant input ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ (*i.e.*, $e_1 \in \mathsf{Sig}(\mathcal{O}_1)$ and $e_2 \in \mathsf{Sig}(\mathcal{O}_2)$), n is a confidence measure between 0 and 1, and ρ is a relationship between e_1 and e_2, typically subsumption (*i.e.*, e_1 is more specific than e_2), equivalence (*i.e.*, e_1 and e_2 are synonyms) or disjointness (*i.e.*, e_1 and e_2 cannot share individuals) [7].

RDF Alignment [5] is the main format used in the Ontology Alignment Evaluation Initiative (OAEI) to represent mappings containing the aforementioned elements. Additionally, mappings are also represented as OWL 2 subclass, equivalence, and disjointness axioms [4]; mapping confidence values (n) are then represented as axiom annotations. Such a representation enables the reuse of the extensive range of OWL 2 reasoning infrastructure that is currently available. Note that alternative formal semantics for ontology mappings have been proposed in the literature (*e.g.*, [2,7,23]), but they are out of scope of the present article because the reasoning in the aligned ontology cannot be achieved directly with OWL 2 reasoning infrastructure, rather it requires custom reasoning facilities.

2.2. Semantic Consequences of the Integration

The ontology resulting from the integration of the two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ via a set of mappings $\mathcal{M}$ typically entails axioms that do not follow from $\mathcal{O}_1$, $\mathcal{O}_2$, or $\mathcal{M}$ alone. These new semantic consequences can be captured by the notion of *deductive difference* [18,19].

Intuitively, the deductive difference between $\mathcal{O}$ and $\mathcal{O}'$ w.r.t. a signature Σ is the set of entailments constructed over Σ that do not hold in $\mathcal{O}$, but do hold in $\mathcal{O}'$. No algorithm is available for computing the deductive difference for DLs more expressive than $\mathcal{EL}$, for which the existence of tractable algorithms is still open [18].

Thus in this paper we rely on the *approximation* of the deductive difference given in Definition 1. This approximation only requires comparing the classification hierarchies of $\mathcal{O}$ and $\mathcal{O}'$ provided by an OWL 2 reasoner, and it has successfully been used in the past in the context of ontology integration [13].

Definition 1 (Approximation of the Deductive Difference)**.** Let A, B be atomic concepts from Σ, $\mathcal{O}$ and $\mathcal{O}'$ be two OWL 2 ontologies, with Σ a signature. We define the approximation of the Σ-deductive difference between $\mathcal{O}$ and $\mathcal{O}'$ (denoted $\mathsf{diff}^{\approx}_{\Sigma}(\mathcal{O}, \mathcal{O}')$ as the set of axioms of the form $A \sqsubseteq B$ satisfying: *(i)* $\mathcal{O} \not\models A \sqsubseteq B$, and *(ii)* $\mathcal{O}' \models A \sqsubseteq B$.

2.3. Violations and Mapping Repair

As already discussed in Section 2.2, the notion of deductive difference can capture the novel entailments of an aligned ontology w.r.t. the input ontologies. However, some of these entailments may be undesired, and are called *violations*, stemming from erroneous mappings in $\mathcal{M}$, or from an inherent incompabilities between the input ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$.

A set of mappings that leads to unsatisfiable classes in $\mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$ is referred to as *incoherent* w.r.t. $\mathcal{O}_1$ and $\mathcal{O}_2$ [21], as formalized in Definition 2. Analogously, a set of mappings that leads to violations of the conservativity principle in the aligned ontology $\mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$ is referred to as *nonconservative* w.r.t. $\mathcal{O}_1$ and $\mathcal{O}_2$ [1,31], as formalized in Definition 3.

Definition 2 (Mapping Incoherence)**.** A set of mappings $\mathcal{M}$ is incoherent with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$, if a class A exists in the signature of $\mathcal{O}_1 \cup \mathcal{O}_2$ such that $\mathcal{O}_1 \cup \mathcal{O}_2 \not\models A \sqsubseteq \bot$ and $\mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M} \models A \sqsubseteq \bot$.

Definition 3 (Mapping Nonconservativity)**.** A set of mappings $\mathcal{M}$ is nonconservative with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$, if a pair of classes A, B exist in the signature of $\mathcal{O}_i$, with $B \neq \bot$, such that $\mathcal{O}_i \not\models A \sqsubseteq B$, and $\mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M} \models A \sqsubseteq B$.

More generally, an alignment being incoherent and/or nonconservative, is called *problematic*, as introduced in Definition 4.

Definition 4 (Problematic Mappings)**.** A set of mappings $\mathcal{M}$ between two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ is problematic, if $\mathcal{M}$ is incoherent and/or nonconservative with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$.

A problematic set of mappings $\mathcal{M}$ can be fixed by removing mappings from $\mathcal{M}$. This process is referred to as *mapping repair* (or repair for short).

Definition 5 (Mapping Repair)**.** Let $\mathcal{M}$ be a problematic set of mappings w.r.t. $\mathcal{O}_1$ and $\mathcal{O}_2$. A set of mappings $\mathcal{R} \subseteq \mathcal{M}$ is a mapping repair for $\mathcal{M}$ w.r.t. $\mathcal{O}_1$ and $\mathcal{O}_2$ if $\mathcal{M} \setminus \mathcal{R}$ is not problematic w.r.t. $\mathcal{O}_1$ and $\mathcal{O}_2$.

A trivial repair is $\mathcal{R} = \mathcal{M}$, since an empty set of mappings is obviously nonproblematic. Nevertheless, the objective is to minimize a loss function over the alignment (*e.g.*, to remove as few mappings as possible or to minimize the total confidence of the removed mappings). Minimal (mapping) repairs are typically referred to in the literature as *mapping diagnosis* [20] — a term coined by Reiter [25] and introduced to the field of ontology debugging in [28].

Definition 6 (Mapping diagnosis)**.** Let $\mathcal{R}$ be a repair for $\mathcal{M}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$. $\mathcal{R}$ is a diagnosis if each $\mathcal{R}' \subset \mathcal{R}$ is not a repair for $\mathcal{M}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$.

In the literature there are different approaches to compute a repair or diagnosis for an incoherent set of mappings. Early approaches were based on Distributed Description Logics (DDL) (*e.g.*, [22,23,24]). Alternatively, if mappings are represented as OWL 2 axioms, a repair or diagnosis can also be computed using the state-of-the-art approaches for debugging and repairing OWL 2 ontologies, which rely on the extraction of justifications for the undesired entailments (*e.g.*, [13,15,27,34]).

"A justification for an entailment in an ontology is a minimal subset of the ontology that is sufficient for the entailment to hold. The set of axioms corresponding to the justification is minimal in the sense that if an axiom is removed from the set, the remaining axioms no longer support the entailment." [10] Definition 7 formally introduces the notion of justification.

Definition 7 (Justification [10])**.** Given an ontology $\mathcal{O}$, and an entailment η such that $\mathcal{O} \models \eta$, $\mathcal{J}$ is a justification in $\mathcal{O}$ of η if $\mathcal{J} \subseteq \mathcal{O}$, $\mathcal{J} \models \eta$, and for all $\mathcal{J}' \subsetneq \mathcal{J}$ it is the case that $\mathcal{J}' \not\models \eta$.

In ontology matching scenarios the use of incomplete reasoning techniques to enhance scalability is very frequent (*e.g.*, [1,11,20,26,31]). Incomplete reasoning leads to an *approximate repair* $\mathcal{R}^{\approx}$, *i.e.*, there is no guaranteee that $\mathcal{M} \setminus \mathcal{R}^{\approx}$ is nonproblematic, but the number of violations caused by the original set of mappings $\mathcal{M}$ tends to be reduced while minimizing the loss function over the original alignment.

Given that the justifications for an entailment are usually exponential in the size of the ontology, [10] approximate mapping repair techniques, based on the extraction of a single justification, has been successfully used in the past to achieve scalability (*e.g.*, LogMap-Full [11]). For this reason, our empirical evaluation does not only consider the (limited) extraction of all the justifications, but also the computation of a single one, as described in details in [10].

3. Experimental Evaluation

This section describes the conducted experimental evaluation. In Section 3.1 we present the used datasets and mapping sets. Section 3.2 introduces the evaluation setting. The obtained results are discussed in Section 3.3.

3.1. Datasets

The datasets are based on the OAEI, an international campaign for the systematic evaluation of ontology matching systems. The matching problems in the OAEI are organised in several tracks, with each track involving different kinds of test ontologies [3,6,8]. In this paper, we have focused on the *anatomy*, *largebio*, *library* and *conference* tracks. For *largebio* we used both the *small* setting, in which reduced fragments of FMA, NCI and SNOMED CT are employed (where the fragments are relevant portions of one of the ontologies with respect to the other two), and the *big* setting, employing the whole ontologies (at the exception of SNOMED CT, for which a large fragment for both FMA and NCI is used). *Library* is composed by not very expressive medium-sized ontologies, while *conference* ontologies are very expressive but of limited size. *Anatomy* is composed by a fragment of NCI ontology (named HUMAN in this context to avoid confusion with the *largebio* dataset) involving human anatomy, that should be matched with an ontology describing the anatomy of mice (called MOUSE ontology). Table 1 summarizes the metrics of the selected ontology pairs for the evaluation, while Tables 2–4 provides the details about the selected subset of mapping sets computed by ontology matching systems participating in the OAEI 2013 and 2014 campaigns. Due to the excessive time required for running a so expensive evaluation, we were forced to select only a representative subset of the computed mappings sets (we have selected, for each track, the alignments with the highest or lowest precision and recall values).[1] Please refer to [3,6] for more information about the datasets and ontology matching systems.

Table 2
Metrics about the relevant mapping sets of the *largebio* dataset.

Ontology 1	Ontology 2	# Mappings	Matching System
FMA	NCI	5862	AML_{14} (BIG)
FMA	NCI	5686	$GOMMA_{13}$ (BIG)
FMA	NCI	3788	$IAMA_{13}$ (BIG)
FMA	NCI	6823	$LogMapBio_{14}$ (BIG)
FMA	NCI	2806	$OMReasoner_{14}$ (BIG)
FMA	NCI	6048	$Reference_{13}$ (BIG)
FMA	NCI	5518	$YAM{+}{+}_{13}$ (BIG)
FMA	SNOMED	12384	AML_{14} (BIG)
FMA	SNOMED	11294	$GOMMA_{13}$ (BIG)
FMA	SNOMED	3198	$IAMA_{13}$ (BIG)
FMA	SNOMED	13704	$LogMapBio_{14}$ (BIG)
FMA	SNOMED	18016	$Reference_{13}$ (BIG)
FMA	SNOMED	13684	$YAM{+}{+}_{13}$ (BIG)
SNOMED	NCI	25252	AML_{14} (BIG)
SNOMED	NCI	24880	$GOMMA_{13}$ (BIG)
SNOMED	NCI	17686	$IAMA_{13}$ (BIG)
SNOMED	NCI	24984	$LogMapBio_{14}$ (BIG)
SNOMED	NCI	37688	$Reference_{13}$ (BIG)
SNOMED	NCI	25200	$YAM{+}{+}_{13}$ (BIG)
FMA	NCI	5380	AML_{14}
FMA	NCI	5252	$GOMMA_{13}$
FMA	NCI	3502	$IAMA_{13}$
FMA	NCI	5960	$MaasMatch_{14}$
FMA	NCI	5781	$LogMapBio_{14}$
FMA	NCI	2724	$OMReasoner_{14}$
FMA	NCI	5122	$YAM{+}{+}_{13}$
FMA	SNOMED	13582	AML_{14}
FMA	SNOMED	7332	$GOMMA_{13}$
FMA	SNOMED	2500	$IAMA_{13}$
FMA	SNOMED	12884	$LogMapBio_{14}$
FMA	SNOMED	16232	$MaasMatch_{14}$
FMA	SNOMED	3040	$OMReasoner_{14}$
FMA	SNOMED	13270	$YAM{+}{+}_{13}$
SNOMED	NCI	28262	AML_{14}
SNOMED	NCI	21110	$GOMMA_{13}$
SNOMED	NCI	16812	$IAMA_{13}$
SNOMED	NCI	28711	$LogMapBio_{14}$
SNOMED	NCI	14240	$OMReasoner_{14}$
SNOMED	NCI	23344	$YAM{+}{+}_{13}$

3.2. Evaluation Settings

System Details. The test environment consists of a desktop computer equipped with 32*GB DDR*3 *RAM* at 1333*MHz* and an *AMD Fusion FX 4350* (quad-core, each running at 4.2*GHz*) as *CPU*. The dataset is stored on a 128*GB SSD*, where the operating system (*Ubuntu* 12.04, 64-bit version) is installed. The employed build of Java Runtime Environment (JRE) is 1.8.0_45-b14, while the one for the Oracle 64-Bit Java Virtual Machine (JVM) is the 25.45-b02 (mixed mode). The amount of memory allocated for the heap of the JVM is 12GB, the processes not involved in the evaluation require approximately 3GB of space, thus leaving 17GB of free RAM (plus 1.8GB of swap memory, that is not used unless totally necessary[2]).

Tested Reasoners. The versions of the employed reasoners are: *(i) Konclude* 0.6.0-408 64-bit [33] (linux

[1] Due to space reasons we can only present a subset of the computed evaluation, a technical report with the full analysis is available at `ftp://ftp.disi.unige.it/person/SolimandoA/aijournaltr.pdf`

[2] This behaviour is enforced by means of the swappiness Linux kernel parameter set to 0, see `http://en.wikipedia.org/wiki/Swappiness` for more information.

Table 1
Metrics about the ontologies employed in the evaluation.

Ontology	Track	#Concepts	#DatatypeP.	#ObjectP.	DL
MOUSE	Anatomy	2744	0	3	$\mathcal{ALE}(\boldsymbol{D})$
HUMAN	Anatomy	3304	0	2	$\mathcal{S}(\boldsymbol{D})$
CMT	Conference	36	10	49	$\mathcal{ALCIN}(\boldsymbol{D})$
CONFERENCE	Conference	60	18	46	$\mathcal{ALCHIF}(\boldsymbol{D})$
CONFOF	Conference	38	23	13	$\mathcal{SIN}(\boldsymbol{D})$
EKAW	Conference	74	0	33	$\mathcal{SHIN}$
IASTED	Conference	140	3	38	$\mathcal{ALCIN}(\boldsymbol{D})$
SIGKDD	Conference	49	11	17	$\mathcal{ALEI}(\boldsymbol{D})$
FMA (NCI)	LargebioSmall	3696	24	0	$\mathcal{ALCN}(\boldsymbol{D})$
FMA (SNOMED)	LargebioSmall	10157	24	0	$\mathcal{ALCN}(\boldsymbol{D})$
NCI (FMA)	LargebioSmall	6488	0	63	$\mathcal{ALC}$
NCI (SNOMED)	LargebioSmall	23958	0	82	$\mathcal{ALCH}$
SNOMED (FMA)	LargebioSmall	13412	0	18	$\mathcal{ALER}$
SNOMED (NCI)	LargebioSmall	51128	0	51	$\mathcal{ALER}$
FMA	LargebioBig	78988	54	0	$\mathcal{ALCN}(\boldsymbol{D})$
NCI	LargebioBig	66724	1	123	$\mathcal{ALCH}(\boldsymbol{D})$
SNOMED (NCI and FMA)	LargebioBig	122464	1	55	$\mathcal{ALER}$
STW	Library	6575	0	0	$\mathcal{AL}$
TheSoz	Library	8376	0	0	$\mathcal{AL}$

Table 3
Metrics about the relevant mapping sets of the *conference* dataset.

Ontology 1	Ontology 2	# Mappings	Matching System
CMT	IASTED	10	AML_{14}
CMT	IASTED	10	AMLbk_{13}
CMT	IASTED	32	MaasMatch_{14}
CMT	IASTED	8	Reference_{14}
CMT	IASTED	15	XMapGen_{13}
CMT	IASTED	11	XMapSig_{13}
CONFOF	IASTED	10	AML_{14}
CONFOF	IASTED	10	AMLbk_{13}
CONFOF	IASTED	18	Reference_{14}
CONFOF	IASTED	19	XMapGen_{13}
CONFOF	IASTED	9	XMapSig_{13}
CONFOF	IASTED	16	YAM++_{13}
CONFERENCE	EKAW	164	MaasMatch_{14}
CONFERENCE	IASTED	68	MaasMatch_{14}
CONFERENCE	IASTED	12	AML_{14}
CONFERENCE	IASTED	12	AMLbk_{13}
CONFERENCE	IASTED	28	Reference_{14}
CONFERENCE	IASTED	17	XMapGen_{13}
CONFERENCE	IASTED	13	XMapSig_{13}
CONFERENCE	IASTED	12	YAM++_{13}
IASTED	SIGKDD	70	AOTL_{14}
IASTED	SIGKDD	68	MaasMatch_{14}

binaries), *(ii) ELK* 0.4.2 [17],[3] *(iii) Pellet* 2.3.1 [30], *(iv) HermiT* 1.3.8 [9].

ELK, *Pellet*, and *HermiT* implement the *OWLReasoner* interface of the *OWL-API* and they all are called

[3]Version compiled on the 29th of May 2015 from the sources available at `https://github.com/klinovp/elk/tree/feature/tracing-complete`

Table 4
Metrics about the relevant mapping sets of the *anatomy* and *library* datasets.

Ontology 1	Ontology 2	# Mappings	Matching System
MOUSE	HUMAN	2956	AML_{14}
MOUSE	HUMAN	2954	AMLbk_{13}
MOUSE	HUMAN	5400	AOT_{14}
MOUSE	HUMAN	336	AOTL_{14}
MOUSE	HUMAN	3077	GOMMAbk_{13}
MOUSE	HUMAN	1962	IAMA_{13}
MOUSE	HUMAN	3107	LogMapBio_{14}
MOUSE	HUMAN	2235	LogMapC_{14}
MOUSE	HUMAN	4024	MaasMatch_{13}
MOUSE	HUMAN	2206	ODGOMS_{13}
MOUSE	HUMAN	3032	Reference_{14}
MOUSE	HUMAN	1891	RSDLWB_{14}
MOUSE	HUMAN	1872	WeSeE_{13}
MOUSE	HUMAN	2056	WMatch_{13}
MOUSE	HUMAN	2790	YAM++_{13}
STW	TheSoz	9582	AML_{13}
STW	TheSoz	7254	AML_{14}
STW	TheSoz	12032	Hertuda_{13}
STW	TheSoz	378	IAMA_{13}
STW	TheSoz	5684	LogMap_{13}
STW	TheSoz	2925	LogMapC_{14}
STW	TheSoz	8922	MaasMatch_{14}
STW	TheSoz	7794	ODGOMS_{13}
STW	TheSoz	6322	Reference_{14}
STW	TheSoz	1624	StringsAuto_{13}
STW	TheSoz	342	RSDLWB_{14}
STW	TheSoz	11948	Xmap_{14}
STW	TheSoz	80686	XMapGen_{13}
STW	TheSoz	2870	XMapSig_{13}
STW	TheSoz	7940	YAM++_{13}

on a fresh thread. A timeout on the classification task is enforced by killing the thread after reaching the timeout value, times are measured using the *getNanoSec* function, because it measures the elapsed time without skew corrections.[4]

ELK is a (very fast) reasoner for the OWL 2 EL profile, thus it cannot guarantee complete results for ontologies outside this profile.

Konclude does not implement the *OWL-API*'s *OWLReasoner* interface and its invocation through *OWLlink* 1.2.1 is raising an *OWLlinkReasonerRuntimeException* exception caused by an *IndexOutOfBoundsException* exception during the parsing of most of the ontologies in our dataset (expressed in OWL/XML format). Thus, *Konclude* is instead called using an external process,[5] using the *ProcessBuilder* class,[6] and it is allowed to use all the available cores. For *Konclude*, timeout on classification is enforced using *timeout* program for Linux,[7] and wall-clock time is measured using the *time* program.[8]

Note that due to its invocation, the measured time for *Konclude* also includes the loading time of the aligned ontology. However, given that we aim at verifying the feasibility of using full OWL 2 reasoners in a mapping diagnosis context, and not at comparing reasoners, all these biases are not influencing our analysis.

It was not possible to extend our analysis to *FaCT++* because its invocation using *JNI* is permanently failing with a *StackOverflowError*.

Justification Extractor. Our evaluation is based on the *black-box* justification extractor described in [10].[9] Black-box extractors typically allow to use any reasoner implementing the axiom pinpointing service. In addition to our previous evaluation, we have also compared the performance of the aforementioned black-box approach using *ELK* 0.4.2, and the *glass-box* trace extraction technique offered by *ELK* reasoner [16]. The tracing functionality offerred by *ELK* keeps track of the subset of the ontology used for the entailment check of the axiom of interest, then the black-box justification module is run only against this relevant subset of the ontology, allowing, in principle, a significant reduction of the required runtime. [10] When we refer to this additional functionality offered by *ELK*, the reasoner will be referred to as ELK_{trace}.

Algorithm 1 Conducted evaluation over (part of) the OAEI 2013-2014 dataset

Input: $\mathcal{O}_1$, $\mathcal{O}_2$: input ontologies $\mathcal{M}$: mappings for $\mathcal{O}_1$ and $\mathcal{O}_2$

$\mathcal{O}_\mathcal{U} := \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$
for each reasoner **do**
  Compute classification of $\mathcal{O}_\mathcal{U}$ (store time in (**I**))
  Compute all the unsatisfiable concepts $unsats$ in $\mathcal{O}_\mathcal{U}$ (store their number in (**II**))
  if $\#unsats > 50$ **then**
    $unsats \leftarrow$ randomly select 50 concepts
  end if
  Compute a single justification for each unsat c (store total time in (**III**))
  Compute 50 justifications for each c (store total time in (**IV**))
  if $\#conservViols > 50$ **then**
    $conservViols \leftarrow$ randomly select 50 violations
  end if
  Compute a single justification for each violations v (store total time in (**V**))
  Compute 50 justifications for each v (store total time in (**VI**))
end for

3.3. Conducted Evaluation

The evaluation algorithm is presented in Algorithm 1, which takes as input a pair of ontologies ($\mathcal{O}_1$ and $\mathcal{O}_2$) and an alignment $\mathcal{M}$ between them from the datasets described in Section 3.1. For each of the available reasoners we compute the classification[10] and record the classification times in seconds (see Tables 5-9 and *Class.(s)* in Tables 10–31). Then, if the classification succeeds, we record the number of unsatisfiable concepts (*#Unsat* in Tables 10–31). For at most 50 of them, we compute justifications[11] (a single one and up to a maximum of 50 justifications, recording the total time in seconds required for completing the respective operations (*1Just.(s)* and *50Just.(s)* in Tables 10–31, respectively). In addition, we also keep track of the percentage of violations for which neither error nor timeout occurred (*%1JustOK* and *%50JustOK*). When

[4] https://docs.oracle.com/javase/8/docs/api/java/lang/System.html#nanoTime--

[5] *Konclude* is runned with "Konclude classification -w AUTO -i aligneOntology.owl"

[6] https://docs.oracle.com/javase/8/docs/api/java/lang/ProcessBuilder.html

[7] With the command "timeout –preserve-status –s TERM timeoutVal cmd"

[8] Using "/usr/bin/time -f %E cmd" command.

[9] Current version available at https://github.com/matthewhorridge/owlexplanation For the experiments we used the version available here: https://github.com/protegeproject/mvn-repo/tree/master/releases/org/semanticweb/owl/explanation/3.3.0

[10] With a timeout of 10, 60, 20 and 10 minutes for *anatomy*, *large-bio*, *library* and *conference*, respectively.

[11] With a timeout of 60 seconds to find each new justification. ELK_{trace} has a timeout of 60 seconds for computing the trace, and than another timeout of 60 seconds for computing each justification.

a timeout or an error occurs, the measured time is only a lower-bound of the real required time.

Analogously, the number of detected conservativity violations are recorded (*#Viols* in Tables 10–31), and, for at most 50 of them, we compute justifications.

Notice that when computing more than one justification, the timeout is enforced for each single justification computation, and reaching a timeout on one of them stops the whole computation. This is motivated by the fact that the search space for justifications is explored by increasing the number of axioms composing them, and it is therefore reasonable that at the first failure the justification module will not be able to find other ones within the given time limit.

Classification. In Tables 5-9 the classification time for a selection of the testcases is shown.

For the *anatomy* dataset (Table 5), all the reasoners succesfully managed to compute the classification of the aligned ontology. The only reasoner that experienced problems was *Pellet*, that permanently raised a *ConcurrentModificationException* exception on several aligned ontologies of this dataset.

In the *largebio-small* dataset (Table 6), for the mapping sets involving FMA and NCI, none of the reasoners failed at classifying the aligned ontology. For the FMA and SNOMED testcase, *Pellet* failed to classify, due to timeouts (T/OUT), half of the cases. For the integration of SNOMED and NCI, *Pellet* failed due to timeouts in all the cases, and exclusively *ELK* could classify the aligned ontology in 3 out of 7 testcases.[12] *Konclude*, for instance, failed with an out of memory error (OOM), and *HermiT* reached the timeout.

For the *largebio-big* dataset (Table 7), not all the mapping sets are available because some of the ontology matchers failed at computing the alignment for this extended case. Regarding classification, the results are very similar to the small version of the dataset, with a little increase of the number of timeouts for *Pellet* due to the increase of size of the aligned ontologies.

For *library* (Table 8), instead, the reasoners succeeded in most of the cases (with the partial exception of *Pellet*), but only *Konclude* managed to classify, within the timeout, the integrated ontology via the mappings computed by *XMapGen*. These mappings include an extremely high number of many to many correspondences, that caused problems to all the reasoners but *Konclude*.

Concerning *conference*, the classification could be performed in the vast majority of the cases, with only a single failure for both *HermiT* and *Pellet*. In Table 9 we only report the cases in which the classification either required a time greater than 1 second, or during which a timeout or error occurred.

Computation of Justifications for Unsatisfiabilities. Tables 10–30, instead, show the details for justification computation for the existing unsatisfiabilites, for relevant cases. For the *library* dataset, the results are omitted due to the lack of unsatisfiable classes in the aligned ontologies (the input ontologies are simple and they do not contain disjointness axioms). Note that *Konclude*, since it was invoked from the command line, could not be evaluated on the justification extraction tasks, due to the limitation of using it from the command line. Missing rows mean that the corresponding reasoner failed at classifying the integrated ontology, due to timeout or an error, and has been excluded.

Note that, the computed times in the Tables 10–30 are only for 50 unsatisfiable classes. Thus, the total times given below for all unsatisfiable classes have been extrapolated from these results.

A subset of the results for the *anatomy* dataset is reported in Tables 10–11. Consider for instance Table 10a, which presents the justification extraction for the mapping set computed by *MaasMatch*. Computing a single justification for each of the $5,972$ unsatisfiable classes, would require for *ELK* 2h (68s for 50 unsatisfiable classes), while >11 days for computing fifty of them (>2h for 50 unsatisfiable classes).

The available version of ELK_{trace} is still in beta quality, and therefore we experienced some problems in the justification extraction process. For this reason, despite the great improvement in the runtime for computing justifications, it exhibited a high number of errors during the evaluation. For instance, for the present task, ELK_{trace} did not manage to correctly compute any justification. When *HermiT* is used, >37m and >90h would be required, respectively. These times are surprisingly lower than the corresponding for *ELK*, despite the latter is an approximated reasoner.

While already the reported times could be considered to be incompatible with "online" repair scenarios, we also need to consider that the runtime is a lower-bound, limited to at most 50 justifications, of the one

[12] Note that *ELK* is an OWL 2 EL reasoner and since NCI falls outside the OWL 2 EL profile, the classification computed by *ELK* for the integration of SNOMED and NCI is incomplete.

extracting all the justifications, whose number is usually exponential in the size of the ontology.

Considering the *conference* dataset, composed by small sized ontologies having high expressivity, we also find cases that could not be compatible with an "online" mapping repair (*e.g.*, almost two hours for *HermiT* in Table 14a, >53m for *Pellet* in Table 13a and >47m for *ELK* in Table 15a). Notice also that in this last case, only 16% of the computations involving *Pellet* finished before the timeout, and therefore the total runtime is expected to be much higher. The worst result for *ELK*$_{trace}$, instead, is in Table 15a, with a runtime >22m.

For the *largebio* datasets, shown in Tables 16-30, both big and small variants, the values are definitely higher and not affordable for an online repair process. Computing a single justification for each unsatisfiable concept in the *largebio* testcase of Table 20a would require >63h for *ELK*, >6h for *ELK*$_{trace}$, >68h for *HermiT*, while >613 days, >485 days and >51 days for computing 50 of them, respectively.

Even considering a testcase of *largebio-small*, with a fairly limited number of unsatisfiable classes (2058 for the mapping set involving FMA and SNOMED and computed by GOMMA, Table 25a), the runtime is still prohibitive if more than one justification is computed. Indeed, computing a single justification would require >10m for *ELK*, >44s for *ELK*$_{trace}$, >6m for *HermiT* and >4m for *Pellet*, while >24 days, >32h, >10 days and >7 days for computing 50 of them, respectively.

It is evident that the proposed runtimes for *largebio* might be only acceptable in an off-line mapping repair process for the small testcases, while they are not affordable for the largest ones.

Computation of Justifications for Conservativity Violations. In Tables 10–31 we show the details for justification computation for the conservativity violations, restricted to the relevant cases.

What is evident from the analyzed testcases is that, the average runtime required for computing the justifications for a conservativity violation (that is, a subsumption axiom between named classes) is in general lower than the time required for computing the justifications for an unsatisfiable class.

On the other hand, it is also true that the number of violations is usually higher than the number of unsatisfiabilities, with millions of violations like in the *library* testcase, and this is balancing the required time for computing justifications.

For instance, for the *anatomy* dataset, in Table 10b, computing a single justification requires >25m for *ELK*, >5h for *ELK*$_{trace}$, and >4m for *HermiT*, while >50h, >43h, and >16h for computing 50 justifications, respectively. It is again notable that the required runtimes for *HermiT* are lower than the corresponding for *ELK* (both variants).

The *conference* dataset exhibits a very limited number of conservativity violations, in general, favored by the extremely reduced size of the signature of the involved ontologies. An exception is represented by the testcase of Table 13b, where computing a single justification requires less than 2 seconds for all the tested reasoners, but computing 50 justifications caused 32% of timeouts for *Pellet*. All the other reasoners were still able to accomplish the task within one minute.

For *largebio*, Table 26b presents a very limited number of conservativity violations (2270), that paired with the reduced runtime for the computation of their justifications, as already discussed, requires at most 10 seconds for computing a single justification, and at most 39 minutes for computing 50 justifications. Even if the required time can appear to be limited, we need to remind again that in order to provide a complete and minimal repair, all the (possibly exponentially many) justifications must be computed, while the proposed runtime is for computing at most 50 justifications.

In other *largebio* testcases, such as that described in Table 28b, *ELK* would need 12 hours to compute a single justification (*ELK*$_{trace}$ raised an error on 90% of the computations and its runtime cannot be compared), while >127 days would be required by *ELK*, and >5 days by *ELK*$_{trace}$ for computing 50 justifications.

The highest number of conservativity violations occurs in the *library* dataset, favored by the extremely high number of many to many correspondences that its mapping sets exhibit. For instance, in the testcase of Table 31b, the required time for the computation of a single justification is >229 days for *ELK*, and >18 days for *HermiT*, while for computing 50 justifications the runtime is >48 and >7 years, respectively. The results for *ELK*$_{trace}$ are omitted because, for the selected testcases, all the single justification computation failed, and all that for multiple justifications reached the timeout. Similarly to the *anatomy* testcase, also for *library* the runtime for justification computation with *HermiT* has been lower than that of *ELK*.

It is again evident that despite the smaller runtime required for computing justifications for conservativity violations, the total runtime would be prohibitive in the majority of the cases, also for off-line repair scenarios.

4. Conclusions

In this paper, we have extended previous evaluations of the feasibility of using OWL 2 reasoning capabilities in mapping repair related tasks, under different aspects. Firstly, the evaluation has been extended to extract more justifications than in the previous evaluation, in order to provide more accurate results. Additionally, the black-box justification extraction method has also been compared against glass-box techniques offered by *ELK* reasoner, which provides the capability of extracting a trace of the relevant subset of an ontology, for the entailment check of a given axiom. Finally, we also analyzed the performances of justification extraction for repairing the so-called conservativity principle.

Our empirical results on the performances of several top-level reasoners suggest that the classification of the integration of medium/large size ontologies via mappings is hard to compute for current OWL 2 reasoners. Furthermore, when OWL 2 reasoners are to be used in mapping repair tasks, the computation time increases considerably, and in the majority of the cases it becomes impractical, even when restricting to reasoners for one of the OWL 2 profiles. From the extended empirical evaluation presented in this paper, the problem is even exacerbated when considering conservativity violations, in addition to consistency ones.

Hence, the integration of ontologies via mappings seems an ideal reasoning benchmarks, and its hardness motivates the interest in approximated repair techniques in the context of ontology matching.

Acknowledgements

This work was supported by the EU FP7 IP project Optique (no. 318338) and the EPSRC project Score!. Alessandro Solimando has been fully supported by the AI*IA 2014 Mobility Grant for visiting Ernesto Jiménez-Ruiz at the Knowledge Representation and Reasoning group, University of Oxford, UK. We also thank the unvaluable help provided by Bernardo Cuenca Grau and Ian Horrocks.

References

[1] Alessandro Solimando, Ernesto Jiménez-Ruiz and Giovanna Guerrini. A Multi-strategy Approach for Detecting and Correcting Conservativity Principle Violations in Ontology Alignments. In *OWLED Workshop*, volume CEUR-WS 1265, pages 13–24, 2014.

[2] Alexander Borgida and Luciano Serafini. Distributed Description Logics: Assimilating Information from Peer Sources. *J. Data Sem.*, 1:153–184, 2003.

[3] Bernardo Cuenca Grau, Zlatan Dragisic, Kai Eckert, et al. Results of the Ontology Alignment Evaluation Initiative 2013. In *Ontology Matching (OM)*, 2013.

[4] Bernardo Cuenca Grau, Ian Horrocks, Boris Motik, Bijan Parsia, Peter F. Patel-Schneider, and Ulrike Sattler. OWL 2: The next step for OWL. *J. Web Sem.*, 6(4):309–322, 2008.

[5] Jérôme David, Jérôme Euzenat, François Scharffe, and Cássia Trojahn. The Alignment API 4.0. *J. Sem. Web*, 2(1):3–10, 2011.

[6] Zlatan Dragisic, Kai Eckert, Jérôme Euzenat, Daniel Faria, et al. Results of the ontology alignment evaluation initiative 2014. In *Int'l Workshop on Ontology Matching (OM)*, pages 61–104, 2014.

[7] Jérôme Euzenat. Semantic Precision and Recall for Ontology Alignment Evaluation. In *Int'l Joint Conf. on Artif. Intell. (IJCAI)*, pages 348–353, 2007.

[8] Jérôme Euzenat, Christian Meilicke, Heiner Stuckenschmidt, Pavel Shvaiko, and Cássia Trojahn. Ontology Alignment Evaluation Initiative: Six Years of Experience. *J. Data Sem.*, 15:158–192, 2011.

[9] Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. Hermit: An OWL 2 reasoner. *J. Autom. Reasoning*, 53(3):245–269, 2014.

[10] Matthew Horridge. Justification Based Explanation in Ontologies. Ph.D. thesis, School. *Ph.D thesis, School of Computer Science, The University of Manchester*, 2011.

[11] Ernesto Jiménez-Ruiz and Bernardo Cuenca Grau. LogMap: Logic-based and Scalable Ontology Matching. In *Int'l Sem. Web Conf. (ISWC)*, pages 273–288, 2011.

[12] Ernesto Jiménez-Ruiz, Bernardo Cuenca Grau, and Ian Horrocks. On the feasibility of using OWL 2 DL reasoners for ontology matching problems. In *OWL Reasoner Evaluation Workshop (ORE)*, 2012.

[13] Ernesto Jiménez-Ruiz, Bernardo Cuenca Grau, Ian Horrocks, and Rafael Berlanga. Ontology integration using mappings: Towards getting the right logical consequences. In *Eur. Sem. Web Conf.*, 2009.

[14] Ernesto Jiménez-Ruiz, Christian Meilicke, Bernardo Cuenca Grau, and Ian Horrocks. Evaluating Mapping Repair Systems with Large Biomedical Ontologies. In *Description Logics*, pages 246–257, 2013.

[15] Aditya Kalyanpur, Bijan Parsia, Matthew Horridge, and Evren Sirin. Finding all justifications of OWL DL entailments. *Int'l Sem. Web Conf. (ISWC)*, pages 267–280, 2007.

[16] Yevgeny Kazakov and Pavel Klinov. Goal-directed tracing of inferences in EL ontologies. In *13th International Semantic Web Conference*, pages 196–211, 2014.

[17] Yevgeny Kazakov, Markus Krötzsch, and Frantisek Simancik. Concurrent Classification of EL Ontologies. In *Int'l Sem. Web Conf. (ISWC)*, pages 305–320, 2011.

[18] Boris Konev, Dirk Walther, and Frank Wolter. The Logical Difference Problem for Description Logic Terminologies. In *Int'l Joint Conf. on Automated Reasoning (IJCAR)*, pages 259–274, 2008.

[19] Roman Kontchakov, Frank Wolter, and Michael Zakharyaschev. Can you Tell the Difference Between DL-Lite Ontologies? In *Int'l Conf. on Knowl. Representation and Reasoning (KR)*, 2008.

[20] Christian Meilicke. *Alignments Incoherency in Ontology Matching*. PhD thesis, University of Mannheim, 2011.

[21] Christian Meilicke and Heiner Stuckenschmidt. Incoherence as a basis for measuring the quality of ontology mappings. In *Ontology Matching (OM)*, 2008.

[22] Christian Meilicke, Heiner Stuckenschmidt, and Andrei Tamilin. Repairing ontology mappings. In *Proc. of AAAI Conf. on Artif. Intell.*, pages 1408–1413, 2007.

[23] Christian Meilicke, Heiner Stuckenschmidt, and Andrei Tamilin. Reasoning Support for Mapping Revision. *J. Log. Comput.*, 19(5), 2009.

[24] Jyotishman Pathak and Christopher G. Chute. Debugging Mappings between Biomedical Ontologies: Preliminary Results from the NCBO BioPortal Mapping Repository. In *Int'l Conf. on Biomedical Ontology (ICBO)*, 2009.

[25] Raymond Reiter. A Theory of Diagnosis from First Principles. *Artif. Intell.*, 32(1), 1987.

[26] Emanuel Santos, Daniel Faria, Cátia Pesquita, and Francisco Couto. Ontology Alignment Repair Through Modularization and Confidence-based Heuristics. *arXiv:1307.5322 preprint*, 2013.

[27] Stefan Schlobach. Debugging and Semantic Clarification by Pinpointing. In *Eur. Sem. Web Conf. (ESWC)*, pages 226–240. Springer, 2005.

[28] Stefan Schlobach and Ronald Cornet. Non-standard Reasoning Services for the Debugging of Description Logic Terminologies. In *Int'l Joint Conf. on Artif. Intell. (IJCAI)*, pages 355–362, 2003.

[29] Pavel Shvaiko and Jérôme Euzenat. Ontology Matching: State of the Art and Future Challenges. *IEEE Transactions on Knowl. and Data Eng. (TKDE)*, 2012.

[30] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical OWL-DL reasoner. *J. Web Sem.*, 5(2):51–53, 2007.

[31] Alessandro Solimando, Ernesto Jiménez-Ruiz, and Giovanna Guerrini. Detecting and correcting conservativity principle violations in ontology-to-ontology mappings. In *Int'l Semantic Web Conference (ISWC)*, volume LNCS 8797, pages 1–16, 2014.

[32] Alessandro Solimando, Ernesto Jiménez-Ruiz, and Giovanna Guerrini. On the Feasibility of Using OWL 2 Reasoners in Ontology Alignment Repair Problems. In *OWL Reasoner Evaluation Workshop (ORE)*, pages 60–67, 2015.

[33] Andreas Steigmiller, Thorsten Liebig, and Birte Glimm. Konclude: System description. *J. Web Sem.*, 27:78–85, 2014.

[34] Boontawee Suntisrivaraporn, Guilin Qi, Qiu Ji, and Peter Haase. A Modularization-Based Approach to Finding All Justifications for OWL DL Entailments. In *Asian Sem. Web Conf. (ASWC)*, 2008.

Table 5

Classification times (s), *anatomy* dataset with selected mapping sets.

	AML$_{14}$	**AMLBK**$_{13}$	**AOT**$_{14}$	**AOTL**$_{14}$	**GOMMABK**$_{13}$	**IAMA**$_{13}$	**LogMapBio**$_{14}$	**LogMapC**$_{14}$
ELK	0.07	0.05	17	0.11	0.12	0.45	0.13	0.08
HERMIT	0.6	0.5	5.93	0.6	1.03	1.12	0.7	0.48
KONCLUDE	0.54	0.7	0.61	0.51	0.57	0.79	0.57	0.6
PELLET	2.45	2.22	ERR	ERR	ERR	ERR	ERR	ERR

	MaasMatch$_{13}$	**ODGOMS**$_{13}$	**Reference**$_{14}$	**RSDLWB**$_{14}$	**WeSeE**$_{13}$	**WMatch**$_{13}$	**YAM++**$_{13}$
ELK	6.67	0.12	0.05	0.08	0.28	0.11	0.06
HERMIT	5.6	0.91	0.58	0.88	4	0.84	0.8
KONCLUDE	0.6	0.55	0.53	0.55	0.62	0.67	0.56
PELLET	ERR	ERR	2.11	ERR	ERR	ERR	2.32

Table 6

Classification times (s), *largebio-small* dataset with selected mapping sets.

FMA-NCI	**AML**$_{14}$	**GOMMA**$_{13}$	**IAMA**$_{13}$	**LogMapBio**$_{14}$	**MaasMatch**$_{14}$	**OMReasoner**$_{14}$	**Reference**$_{13}$	**YAM++**$_{13}$
ELK	0.12	1.17	0.1	0.19	0.21	0.1	1.07	0.11
HERMIT	16	21	15	18	3.58	18	53	18
KONCLUDE	7.6	7.45	7.44	8.25	1.3	9.52	6.01	9.81
PELLET	19	17	17	29	T/OUT	16	11	22

FMA-SNOMED	**AML**$_{14}$	**GOMMA**$_{13}$	**IAMA**$_{13}$	**LogMapBio**$_{14}$	**MaasMatch**$_{14}$	**OMReasoner**$_{14}$	**Reference**$_{13}$	**YAM++**$_{13}$
ELK	0.78	0.75	0.49	0.78	4.22	0.57	0.89	0.58
HERMIT	11	9.33	0.54	11	2.86	11	3.75	5.23
KONCLUDE	4.82	4.33	1.52	4.63	5.77	4.99	3.37	3.83
PELLET	2,119	273	1.73	T/OUT	T/OUT	192	T/OUT	T/OUT

SNOMED-NCI	**AML**$_{14}$	**GOMMA**$_{13}$	**IAMA**$_{13}$	**LogMapBio**$_{14}$	**OMReasoner**$_{14}$	**Reference**$_{13}$	**YAM++**$_{13}$
ELK	4.26	3.58	3.09	4.8	2.95	4.36	3.41
HERMIT	T/OUT	53	57	T/OUT	74	4.94	T/OUT
KONCLUDE	OOM	17	16	OOM	18	19	OOM
PELLET	T/OUT	T/OUT	T/OUT	T/OUT	T/OUT	T/OUT	T/OUT

Table 7

Classification times (s), *largebio-big* dataset with selected mapping sets.

FMA-NCI	**AML**$_{14}$	**GOMMA**$_{13}$	**IAMA**$_{13}$	**LogMapBio**$_{14}$	**OMReasoner**$_{14}$	**Reference**$_{13}$	**YAM++**$_{13}$
ELK	2.01	1.99	1.95	2.05	2.04	1.94	1.99
HERMIT	330	446	315	327	486	1,003	406
KONCLUDE	60	61	63	70	61	39	60
PELLET	1,182	722	827	1,997	1,108	165	2,292

FMA-SNOMED	**AML**$_{14}$	**GOMMA**$_{13}$	**IAMA**$_{13}$	**LogMapBio**$_{14}$	**Reference**$_{13}$	**YAM++**$_{13}$
ELK	6.16	6	6.3	6.72	6.65	6.47
HERMIT	840	840	89	883	302	730
KONCLUDE	63	71	26	64	41	55
PELLET	T/OUT	T/OUT	1,762	T/OUT	T/OUT	T/OUT

SNOMED-NCI	**AML**$_{14}$	**GOMMA**$_{13}$	**IAMA**$_{13}$	**LogMapBio**$_{14}$	**Reference**$_{13}$	**YAM++**$_{13}$
ELK	7.34	8.01	7	8.1	9.4	7.45
HERMIT	T/OUT	51	144	T/OUT	18	T/OUT
KONCLUDE	OOM	33	37	OOM	33	OOM
PELLET	T/OUT	T/OUT	T/OUT	T/OUT	T/OUT	T/OUT

Table 8

Classification times (s), library dataset with selected mapping sets.

	AML$_{13}$	**AML**$_{14}$	**Hertuda**$_{13}$	**IAMA**$_{13}$	**LogMap**$_{13}$	**LogMapC**$_{14}$	**MaasMatch**$_{14}$	**ODGOMS**$_{13}$
ELK	13	0.47	35	0.06	1.06	0.07	9.76	0.32
HERMIT	1,059	16	68	1.03	1.68	0.9	37	1.91
KONCLUDE	5.81	2.28	17	1.13	1.72	0.78	2.3	1.14
PELLET	T/OUT	7.79	T/OUT	0.22	1.08	0.17	T/OUT	1.69

	Reference$_{14}$	**RSDLWB**$_{14}$	**StringsAuto**$_{13}$	**Xmap**$_{14}$	**XmapGen**$_{13}$	**XmapSig**$_{13}$	**YAM++**$_{13}$
ELK	0.1	0.06	0.07	29	T/OUT	0.16	2.09
HERMIT	0.86	0.94	0.86	56	T/OUT	1.57	7.5
KONCLUDE	0.86	1.2	0.81	8.06	59	1.77	2.22
PELLET	0.29	0.22	0.2	T/OUT	T/OUT	1	87

Table 9

Classification times (s), *conference* dataset with selected mapping sets.

CMT-IASTED	**AML**$_{14}$	**AMLbk**$_{13}$	**MaasMatch**$_{14}$	**Reference**$_{14}$	**XMapGen**$_{13}$	**XMapSig**$_{13}$	**YAM++**$_{13}$
ELK	0.02	0.02	0.02	0.02	0	0.07	0.02
HERMIT	0.11	0.18	0.14	0.13	0.12	0.12	0.12
KONCLUDE	0.1	0.18	0.09	0.08	0.17	0.08	0.08
PELLET	2.58	4.11	ERR	2.64	2.6	2.58	2.57

CONFOF-IASTED	**AML**$_{14}$	**AMLbk**$_{13}$	**Reference**$_{14}$	**XMapGen**$_{13}$	**XMapSig**$_{13}$	**YAM++**$_{13}$
ELK	0.01	0	0.12	0	0.01	0.04
HERMIT	0.22	0.26	0.24	0.22	0.21	0.22
KONCLUDE	0.12	0.2	0.07	0.18	0.13	0.11
PELLET	3.28	3.48	3.36	4.26	3.28	3.3

EDAS-IASTED	**AML**$_{14}$	**AMLbk**$_{13}$	**Reference**$_{14}$	**XMapGen**$_{13}$	**XMapSig**$_{13}$	**YAM++**$_{13}$
ELK	0.03	0.01	0.02	0.01	0	0.01
HERMIT	0.14	0.14	0.16	0.26	0.14	0.15
KONCLUDE	0.18	0.16	0.08	0.22	0.12	0.11
PELLET	2.11	2.3	2.08	6.76	2.44	2.1

	CONFERENCE-IASTED:**MaasMatch**$_{14}$	IASTED-SIGKDD:**AOTL**$_{14}$	IASTED-SIGKDD:**MaasMatch**$_{14}$
ELK	0.06	0.02	0.04
HERMIT	T/OUT	13	0.58
KONCLUDE	0.36	0.25	0.09
PELLET	7.05	16	2.86

Table 10

Justification extraction in the *anatomy* dataset with MaasMatch$_{13}$

(a) Unsatisfiabilites

Reasoner	**Class.(s)**	**#Unsat**	**1Just.(s)**	**%1JustOK**	**50Just.(s)**	**%50JustOK**
ELK	6.67	5,972	68	100	8,542	100
ELK$_{trace}$	6.67	5,972	769	0	6,001	0
HermiT	5.6	5,972	19	100	2,726	100

(b) Conservativity violations

Reasoner	**Class.(s)**	**#Viol**	**1Just.(s)**	**%1JustOK**	**50Just.(s)**	**%50JustOK**
ELK	6.67	64,580	1.17	100	140	100
ELK$_{trace}$	6.67	64,580	15	0	120	0
HermiT	5.6	64,580	0.2	100	47	100

Table 11

Justification extraction in the *anatomy* dataset with WeSeE$_{13}$

(a) Unsatisfiabilites

Reasoner	**Class.(s)**	**#Unsat**	**1Just.(s)**	**%1JustOK**	**50Just.(s)**	**%50JustOK**
ELK	0.28	2,015	7.65	100	51	100
ELK$_{trace}$	0.28	2,015	1.19	0	160	98
HermiT	4	2,015	5.98	100	497	100

(b) Conservativity violations

Reasoner	**Class.(s)**	**#Viol**	**1Just.(s)**	**%1JustOK**	**50Just.(s)**	**%50JustOK**
ELK	0.28	7,298	0.12	100	43	100
ELK$_{trace}$	0.28	7,298	0.01	16	4.66	98
HermiT	4	7,298	0.08	100	71	98

Table 12
Justification extraction in the *conference* dataset (CONFERENCE-EDAS) with MaasMatch$_{13}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0	78	5.67	100	583	100
ELK$_{trace}$	0	78	0.44	0	271	100
HermiT	0.02	93	2.77	100	455	100
Pellet	0.03	93	1.92	100	277	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0	7	0.11	100	10	100
ELK$_{trace}$	0	7	0.01	0	4.79	100
HermiT	0.02	8	0.04	100	11	100
Pellet	0.03	8	0.03	100	6.99	100

Table 13
Justification extraction in the *conference* dataset (CONFERENCE-EKAW) with MaasMatch$_{14}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.02	54	37	100	1,037	100
ELK$_{trace}$	0.02	54	1.52	0	694	98
HermiT	0.01	63	3.3	100	296	100
Pellet	0.01	63	62	98	2,569	16

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.02	114	0.46	100	29	100
ELK$_{trace}$	0.02	114	0.05	12	7.48	98
HermiT	0.01	115	0.04	100	4.1	100
Pellet	0.01	115	0.02	100	21	68

Table 14
Justification extraction in the *conference* dataset (CONFOF-EDAS) with XMapGen$_{13}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0	23	2.7	100	23	100
ELK$_{trace}$	0	23	0.29	0	8	100
HermiT	0.02	31	2.26	100	6,702	97
Pellet	0.01	31	1.22	100	6,169	97

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0	6	0.03	100	1.36	100
ELK$_{trace}$	0	6	0.01	0	0.49	100
HermiT	0.02	6	0.02	100	27	100
Pellet	0.01	6	0.01	100	13	100

Table 15
Justification extraction in the *conference* dataset (CONFOF-SIGKDD) with AOTL_{14}

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.01	61	25	100	2,355	100
ELK$_{trace}$	0.01	61	1.53	0	1,104	90
HermiT	0	61	1.9	100	1,054	100
Pellet	0.01	61	1.01	100	1,015	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.01	2	0.3	100	1,500	50
ELK$_{trace}$	0.01	2	0.03	0	60	50
HermiT	0	2	0.04	100	1,500	50
Pellet	0.01	2	0.01	100	1,500	50

Table 16
Justification extraction in FMA-NCI (*largebio-big* dataset) with Reference_{13}

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	1.94	30,494	41	100	20,294	92
ELK$_{trace}$	1.94	30,494	2.58	0	1,587	82
HermiT	1,003	30,590	49	100	13,836	92
Pellet	165	30,590	42	100	30,878	80

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	1.94	20,710	0.88	100	429	86
ELK$_{trace}$	1.94	20,710	0.01	36	22	86
HermiT	1,003	20,814	1.06	100	422	88

Table 17
Justification extraction in FMA-NCI (*largebio-big* dataset) with GOMMA_{13}

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	1.99	5,561	45	100	153	100
ELK$_{trace}$	1.99	5,561	2.28	0	55	100
HermiT	446	5,574	76	100	700	100
Pellet	722	5,574	62	100	1,333	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	1.99	14,187	0.85	100	3.37	100
ELK$_{trace}$	1.99	14,187	0.03	68	1.61	100
HermiT	446	14,635	1.17	100	13	100

Table 18
Justification extraction in FMA-SNOMED (*largebio-big* dataset) with Reference$_{13}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	6.65	78,482	75	100	20,709	88
ELK$_{trace}$	6.65	78,482	7.3	0	2,666	78
HermiT	302	78,482	102	100	4,169	98

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	6.65	187,121	1.7	100	76	100
ELK$_{trace}$	6.65	187,121	0.02	30	29	94
HermiT	302	187,121	3	100	55	100

Table 19
Justification extraction in FMA-SNOMED (*largebio-big* dataset) with YAM++$_{13}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	6.47	57,074	67	100	12,052	96
ELK$_{trace}$	6.47	57,074	6.83	0	2,098	82
HermiT	730	57,074	179	100	5,001	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	6.47	117,316	1.74	100	440	86
ELK$_{trace}$	6.47	117,316	0.03	40	31	86
HermiT	730	119,236	4.44	100	269	94

Table 20
Justification extraction in SNOMED-NCI (*largebio-big* dataset) with GOMMA$_{13}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	8.01	127,850	90	100	20,741	90
ELK$_{trace}$	8.01	127,850	9.37	0	1,746	86
HermiT	51	131,222	94	100	15,983	84

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	8.01	151,488	2.43	100	250	96
ELK$_{trace}$	8.01	151,488	0.01	4	22	92

Table 21
Justification extraction in SNOMED-NCI (*largebio-big* dataset) with LogMapBio_{14}

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	8.1	37	123	100	4,387	100
ELK$_{trace}$	8.1	37	8.4	0	349	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	8.1	439,887	3.58	100	706	86
ELK$_{trace}$	8.1	439,887	0.02	14	31	100

Table 22
Justification extraction in SNOMED-NCI (*largebio-big* dataset) with Reference_{13}

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	9.4	158,645	99	100	17,293	92
ELK$_{trace}$	9.4	158,645	13	0	1,674	90
HermiT	18	161,202	83	100	15,126	90

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	9.4	576,736	3.03	100	332	92
ELK$_{trace}$	9.4	576,736	0.02	0	44	88

Table 23
Justification extraction in FMA-NCI (*largebio-small* dataset) with Reference$_{13}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	1.07	4,401	19	100	25,929	84
ELK$_{trace}$	1.07	4,401	1.53	0	1,864	78
HermiT	53	4,402	5.84	100	7,338	96
Pellet	11	4,402	7.3	100	12,609	94

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	1.07	21,296	0.15	100	253	94
ELK$_{trace}$	1.07	21,296	0.02	36	21	88
HermiT	53	21,400	0.07	100	309	90

Table 24
Justification extraction in FMA-NCI (*largebio-small* dataset) with LogMapBio$_{14}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.19	0	0	0	0	0
ELK$_{trace}$	0.19	0	0	0	0	0
HermiT	18	467	16	100	3,777	100
Pellet	29	467	11	100	2,288	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.19	26,657	0.14	100	61	98
ELK$_{trace}$	0.19	26,657	0.04	74	3.14	98
HermiT	18	26,603	0.08	100	1.44	100

Table 25
Justification extraction in FMA-SNOMED (*largebio-small* dataset) with GOMMA$_{13}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.75	2,058	16	100	50,919	70
ELK$_{trace}$	0.75	2,058	1.09	0	2,855	62
HermiT	9.33	2,058	9.42	100	22,909	86
Pellet	273	2,058	6.47	100	15,963	90

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.75	18,425	0.29	100	73	98
ELK$_{trace}$	0.75	18,425	0.06	88	8.68	94
HermiT	9.33	18,425	0.14	100	67	98

Table 26
Justification extraction in FMA-SNOMED (*largebio-small* dataset) with IAMA_{13}

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.49	22,925	8.7	100	10,839	94
ELK$_{trace}$	0.49	22,925	0.77	0	1,040	88
HermiT	0.54	22,925	5.78	100	151	100
Pellet	1.73	22,925	4.96	100	66	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	0.49	2,270	0.23	100	51	100
ELK$_{trace}$	0.49	2,270	0.01	2	26	90
HermiT	0.54	2,270	0.13	100	12	100

Table 27
Justification extraction in FMA-SNOMED (*largebio-small* dataset) with MaasMatch_{14}

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	4.22	21,946	40	100	10,267	98
ELK$_{trace}$	4.22	21,946	5.82	0	3,626	78
HermiT	2.86	21,946	25	100	3,219	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	4.22	717,697	0.89	100	289	100
ELK$_{trace}$	4.22	717,697	0.04	2	91	58
HermiT	2.86	697,459	0.51	100	188	98

Table 28
Justification extraction in SNOMED-NCI (largebiosmall dataset) with LogMapBio$_{14}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	4.8	23	210	87	10,030	87
ELK$_{trace}$	4.8	23	5.3	0	585	100

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	4.8	750,226	2.87	100	733	86
ELK$_{trace}$	4.8	750,226	0.03	10	34	98

Table 29
Justification extraction in SNOMED-NCI (largebiosmall dataset) with Reference$_{13}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	4.36	66,832	74	100	6,983	98
ELK$_{trace}$	4.36	66,832	6.23	0	1,417	94
HermiT	4.94	69,218	58	100	6,913	88

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	4.36	576,736	1.53	100	266	96
ELK$_{trace}$	4.36	576,736	0.02	0	42	86

Table 30
Justification extraction in SNOMED-NCI (largebiosmall dataset) with OMReasoner$_{14}$

(a) Unsatisfiabilites

Reasoner	Class.(s)	#Unsat	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	2.95	35,568	38	100	9,501	96
ELK$_{trace}$	2.95	35,568	3.74	0	1,517	86
HermiT	74	39,942	38	100	24,054	88

(b) Conservativity violations

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	2.95	62,783	1.03	100	201	96
ELK$_{trace}$	2.95	62,783	0.01	12	25	90

Table 31
Justification extraction in the library dataset

(a) Conservativity violations with AML_{13}

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	13	15,708,616	35	66	2,794	14
HermiT	1,059	15,708,616	3.58	100	315	100

(b) Conservativity violations with Hertuda_{13}

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	35	26,782,359	37	62	2,878	12
HermiT	68	26,782,359	2.91	100	414	98

(c) Conservativity violations with XMap_{14}

Reasoner	Class.(s)	#Viol	1Just.(s)	%1JustOK	50Just.(s)	%50JustOK
ELK	29	24,507,254	33	62	2,641	22
HermiT	56	24,507,254	2.07	100	303	98