Author manuscript, published in "Journal on Data Semantics 3, 1 (2014) 25-46"
DOI : 10.1007/s13740-013-0019-z

hal-00975284, version 1 - 8 Apr 2014

***Journal on Data Semantics* 3(1):25–46, 2014**
Authors copy

Context-based matching: design of a flexible framework and experiment

Angela Locoro · Jérôme David · Jérôme Euzenat

Submitted: 25/7/2012 / Accepted: 2013 / Published 2014

Abstract Context-based matching finds correspondences between entities from two ontologies by relating them to other resources. A general view of context-based matching is designed by analysing existing such matchers. This view is instantiated in a path-driven approach that (a) anchors the ontologies to external ontologies, (b) finds sequences of entities (path) that relate entities to match within and across these resources, and (c) uses algebras of relations for combining the relations obtained along these paths. Parameters governing such a system are identified and made explicit. They are used to conduct experiments with different parameter configurations in order to assess their influence. In particular, experiments confirm that restricting the set of ontologies reduces the time taken at the expense of recall and F-measure. Increasing path length within ontologies increases recall and F-measure as well. In addition, algebras of relations allows for a finer analysis, which shows that increasing path length provides more correct or non precise correspondences, but marginally increases incorrect correspondences.

Keywords Context-based Ontology Matching · Knowledge Representation and Interoperability · Algebras of Relations · Semantic Web

Angela Locoro
DIBRIS, University of Genova
Via Dodecaneso, 35, 16146 Genova, Italy
E-mail: Angela.Locoro@unige.it

Jérôme David, Jérôme Euzenat
INRIA & LIG
655 avenue de l'Europe, 38330 Montbonnot Saint Martin, France
E-mail: Jerome.Euzenat@inria.fr, E-mail: Jerome.David@inria.fr

1 Introduction and motivations

The Semantic Web relies on the expression of formalized knowledge on the Web. Data is expressed in the framework of ontologies (theories describing the vocabulary used for expressing data). However, due to the decentralisation of the Web, ontologies may be heterogeneous and have to be reconciled. One way to reconcile ontologies is to find correspondences between their entities. This is called ontology matching [11] and the resulting set of correspondences is called an alignment. Each correspondence relates entities from each of the ontologies with a particular relation, e.g., equivalence, subsumption.

Context-based ontology matching works by taking advantage of intermediate resources to which the two ontologies to be matched can be connected. This is in contrast with content-based matchers, which compare the content of ontologies for matching them, whereas context-based matching use relationships, called anchors, between the entities of the ontologies to be matched and other ontologies on the web. For instance, in Figure 1, Beef from the Agrovoc thesaurus and Food from the NAL thesaurus are anchored to the concepts with the same names in the TAP ontology. Then because Food subsumes Beef in TAP, it is assumed that Food from NAL also subsumes Beef from Agrovoc.

Context-based matchers have already been shown beneficial [21, 18]. However, there is a wide latitude in their design: They depend on the type of resources to be considered (ontologies, encyclopedia, fully informal resources, etc.), how relations are obtained within these resources (asserted, inferred, etc.), how many will be considered (the first one that provides a result or as many as possible), how entities are anchored (simple or complex matchers), and how results are combined when there are several correspondences for the same pair of entities (by vote, by conjunction, etc.). We provide a general framework highlighting these aspects.

hal-00975284, version 1 - 8 Apr 2014

The goal of this paper is to better understand the influence of some of these parameters on the quality of the resulting alignments. For that purpose, we design a flexible context-based matcher that offers various ways to parameterise its behaviour. This renders explicit the various options and allows us to combine them. In order to evaluate the impact of these options on the matching task, we took the following steps:

- determine atomic parameters that constitute these configurations;
- develop a context-based matcher that can be configured to adopt some of the most significant variations of the context-based matching approach;
- design experiments to assess the advantages of each configuration;
- compare the results to establish their strengths and weaknesses.

These experiments confirm that not restricting the intermediate ontologies to be used increases F-measure, although this is the main factor affecting speed. They also establish that extending the search for new paths provides more correspondences and increases F-measure. In addition, through the use of algebras of relations, we were able to analyse precisely some of these improvements, and show that increasing the path length provides new correct and non precise correspondences, but few incorrect correspondences. This suggests proper ways of dealing with this problem. We also compared the results to those of two state-of-the-art matchers: LogMap [17] and YAM++ [19]. This shows that simple context-based matching finds at least as many correspondences.

The paper is organised as follows: §2 introduces and synthesises the state-of-the-art in context-based matching. §3 presents the architecture of a system, generalising Scarlet, along with the main operations of context-based matching and how they have been parameterised. §4 introduces the experimental set-up, whereas §5 discusses the experiment results. §6 concludes and outlines future directions for this work.

2 Context-based matching

Ontology matching must identify relations between ontology entities from two ontologies. These are returned as correspondences of the form $\langle e, r, e' \rangle$ such that e is an entity from the first ontology, e' is an entity from the second ontology and r is the relation assumed to hold between them. Often, matchers associate a measure of their confidence with each correspondence they return. In the following, we consider correspondences between named ontology entities (classes, properties, etc.). Relations may be subsumption ($<$ and $>$ and their reflexive versions $\leq$ and $\geq$), equivalence ($=$) or disjointness ($\perp$) between these entities.

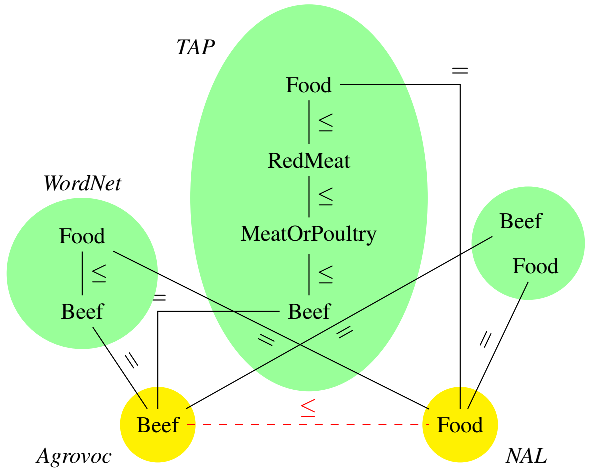

Fig. 1 Scarlet example: several results are returned and must be aggregated (adapted from [21]). Two paths are found relating Beef in Agrovoc to Food in NAL and WordNet. The aggregation of their relations indicates that the former is more specific than the latter.

Context-based matching contrasts with content-based matching. Matching ontologies with content-based techniques compares ontology entities (classes, properties) by relying only on its internal content, such as their annotations, structures, and/or semantics. For the same purpose, context-based matching also uses the context of these ontologies, e.g., resources that they annotate, and message exchanges between agents that use them. For instance, Figure 1 shows two entities from the Agrovoc (FAO)[1] and NAL (US DoA)[2] thesauri that had to be matched in the *food* test case of OAEI-2007 [12]. When considering concepts `Beef` and `Food`, the use of ontologies found on the Web, such as the TAP[3] ontology, helps deduce that `Beef` is less general than `Food`. The same result can also be obtained with the help of WordNet since `Beef` is a hyponym (is a kind) of `Food`. Thus, multiple sources of background knowledge can simultaneously help.

2.1 Early work on context-based ontology matching

Context can take different forms, such as a set of resources such as web pages or pictures that have been annotated with the concepts of an ontology [23]. It can also be some general purpose resource such as a dictionary (WordNet is very often used in ontology matchers).

We concentrate here on systems that use ontological resources as context for matching. By ontological resources,

[1] `http://www.fao.org/aims/ag_intro.htm`.
[2] `http://www.nal.usda.gov/`.
[3] `http://139.91.183.30:9090/RDF/VRP/Examples/tap.rdf`.

hal-00975284, version 1 - 8 Apr 2014

we mean ontologies or knowledge bases, e.g., formalised data sets. Even with this restriction, several context-based ontology matchers have been elaborated over the years:

– using domain specific ontologies, e.g., in the field of anatomy [24, 1];
– using upper-level ontologies [18, 16];
– using linked data as background knowledge [15, 14];
– using all the ontologies available on the Semantic Web, such as in the work on Scarlet [21].

By focusing on a specific domain, such as in [1] and [24], authors were able to provide deeper insights on ontology concept similarities, especially based on the analysis of its respective structural relations, i.e., not only hierarchical, but also relational in its broadest sense (for example by means of the *partOf* relation), or by approximating matching measures when different local hierarchies contain the same concept or group of concepts.

In [18] general purpose upper ontologies are exploited to match ontologies by relating entities if and only if they have the same upper level context. GeRoMeSuite has been extended to select several intermediate ontologies before performing matching [20].

The BLOOMS system [15] is a first attempt to use Linked Open Data (LOD)[4] for schema-level matching. It tries to connect categories coming from two schemas, transform them in trees of senses for each concept to be matched, and compare such trees of senses for discovering hierarchical relations between such concepts. Its evolution, BLOOMS+ [16], exploits the Proton upper-level ontology to enhance the LOD schema-level matching task.

Scarlet [21] tries to find a relation between two concepts by using all the ontologies on the Web for discovering relational paths that connect them. It is presented in more details in §2.2. In [14], a macro scale analysis of thousands of mapped ontologies is carried out in order to detect morphological features as well as power distribution laws in the resulting graphs. In this way, some hints on what exists now and on how to organise and evolve existing knowledge on the Web by means of forthcoming ontologies are provided.

The difficulty of context-based matching is a matter of balance: adding context provides new information, and hence, helps increase recall, but this new information may also generate incorrect correspondences which decrease precision.

As can be observed, there are various ways to use ontological resources for context-based ontology matching. Many options can be taken concerning the type of resource to be used or the way it is connected to the ontologies to be matched. Our goal is to explore these options. For that purpose, we decided to extend an existing ontology matcher.

[4] `http://linkeddata.org/.`

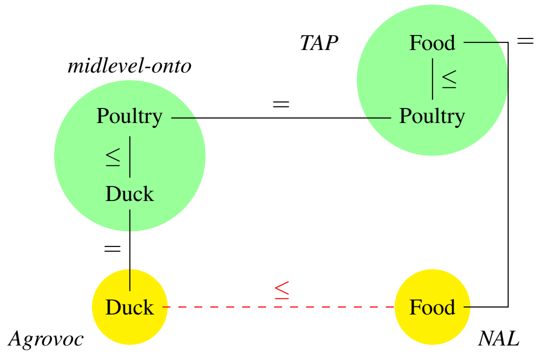

Fig. 2 Scarlet composition example (adapted from [21]). There is no intermediate ontology providing a correspondence between Duck and Food. However, two intermediate ontologies (midlevel-onto and NAL) provide a path between these concepts through the Poultry concept. The relations along this path show that Duck is more specific than Food.

2.2 Scarlet

Our starting point was Scarlet[5] [21] because it already took into account the versatility of context-based matching.

Scarlet [21, 22] is an ontology matcher that operates by contextualising ontologies with ontologies that can be found on the Web. But Scarlet is more complex than this definition, since it involves selecting ontologies for context, matching entities from the initial ontologies and those of the context, and composing the relations obtained after matching. The rationale behind Scarlet is that using more ontologies improves the results. The problem raised by the heterogeneity of ontologies is solved by taking advantage of these many heterogeneous ontologies, which is based on the following principles:

– using the ontologies on the Web as context;
– composing the relations obtained through these ontologies: this covers reasoning within the ontology for deducing the relations between entities (Figure 1) or reasoning across ontologies (Figure 2).

In more details, Scarlet processing roughly consists of the following steps:

1. harvest ontologies on the Web with either Swoogle [8] or Watson [4];
2. select those which are related to the ontologies to match: usually this is achieved by selecting, for each pair of named entities, the ontologies that contain both names;
3. find anchors between the ontologies to match and those that have been selected: here Scarlet uses simple string equivalence;
4. compose the relations between entities through the intermediate ontologies: this is done by returning the relation found in the ontology (see Figure 2);

[5] `http://scarlet.open.ac.uk/.`

5. aggregate the obtained results (see Figure 1).

When no ontology contains the pair of terms, another implemented variation was to use several ontologies and to bridge them in order to increase the chances to find the pair of terms (see Figure 2).

This can become a very complex procedure so it is restricted to finding, for each pair of ontologies, the intersection between the entities subsuming one term and those subsumed by the other, which helps quickly find subsumption relations (see Figure 4).

Three variants of Scarlet have been experimented against Agrovoc (FAO) and NAL (US DoA). The considered variants were:

$S1$ works with only one intermediate ontology at a time: it retrieves the ontologies covering both candidate terms from both ontologies, and delivers all the correspondences that it finds between matched concepts (Figure 1);

$S1'$ is like S1 but it stops at the first correspondence that it finds;

$S2$ implements path search in the graph of ontologies (Figure 2), but only through direct subsumers (and no subsumees).

In all cases, the search for anchors was provided by strict string matching on terms as bags of words, and candidate ontologies were provided by Swoogle. Because of the lack of a full reference alignment in the data set, results were manually assessed and only reported on precision. They provide an average value of 70% precision. This is expected with the given anchoring strategy, indeed, anchoring with string equivalence usually provides high precision. This result has even been improved by using word-sense disambiguation techniques, which allow for better discriminating similar terms [13]. However, this is rather good given that Scarlet returns subsumption relations.

We went on by further generalising the Scarlet approach.

2.3 A generalised view of context-based matching

Because context-based matching is very versatile, we synthesise its behaviour in a generalised view that aims at covering and extending existing matchers. For that purpose, we decompose the context-based matching process in 7 steps described in Figure 3:

Ontology arrangement preselects and ranks the ontologies to be explored as intermediate ontologies. The preselection may retain all the ontologies from the Web or ontologies belonging to a particular type, such as upper ontologies, domain dependent ontologies, e.g., medical or biological ontologies, competencies, popular ontologies, recommended ontologies, or any customised set of ontologies.
The ordering may be based on the likeliness for the ontology to be useful, usually measured by a distance. Such a distance may be based on the proximity of the ontology with the ontology to be matched [5], the existence of alignments between them [6], or the availability of quickly computable anchors.

Contextualisation, or anchoring, finds anchors between the ontologies to be matched and the candidate intermediate ontologies. These anchors are obtained through an ontology matching method or by using existing alignments. They can be correspondences of any types including various relations and confidence measures. In principle, any ontology matching method may be used for anchoring; in practice, this is usually a fast method because anchoring is only a preliminary step.

Ontology selection restricts the candidate ontologies that will actually be used. This selection relies usually on the computed anchors by selecting those ontologies in which anchors are present.

Local inference obtains relations between entities of a single ontology. It may be reduced to logical entailment. It may also use weaker procedures, especially when intermediate resources have no formal semantics, e.g., thesauri. It could then be replaced by the use of asserted relations of the ontologies or relations obtained through composing existing ones.

Global inference finds relations between two concepts of the ontologies to be matched by concatenating relations obtained from local inference and correspondences across intermediate ontologies

Composition determines the relation holding between the source and target entities by composing the relations in the path (sequence of relations) connecting them. The composition method may be functional ($= \cdot =$ is $=$), order-based ($< \cdot \leq$ is $<$) or relational ($\perp \cdot \geq$ is $\perp$).

Aggregation combines relations obtained between the same pair of entities. It can either simply return all correspondences or return only one correspondence with an aggregated relation. Aggregation itself can be based on various methods such as relation aggregation operators (e.g., conjunction), popularity (selecting the relation which is obtained from the most paths) or confidence (selecting the relation with the highest confidence).

These steps extend those provided in the descriptions of Scarlet [21]: contextualisation was called anchoring, selection was considered, local and global inference as well as composition were gathered in a set of "derivation rules" and aggregation was called combining. GeRoMeSuite has also identified the arrangement (called selection), anchoring, local inference (including composition), and aggregation steps [20] to which a consistency check is added. This presentation provides a finer decomposition of context-based match-

hal-00975284, version 1 - 8 Apr 2014

hal-00975284, version 1 - 8 Apr 2014

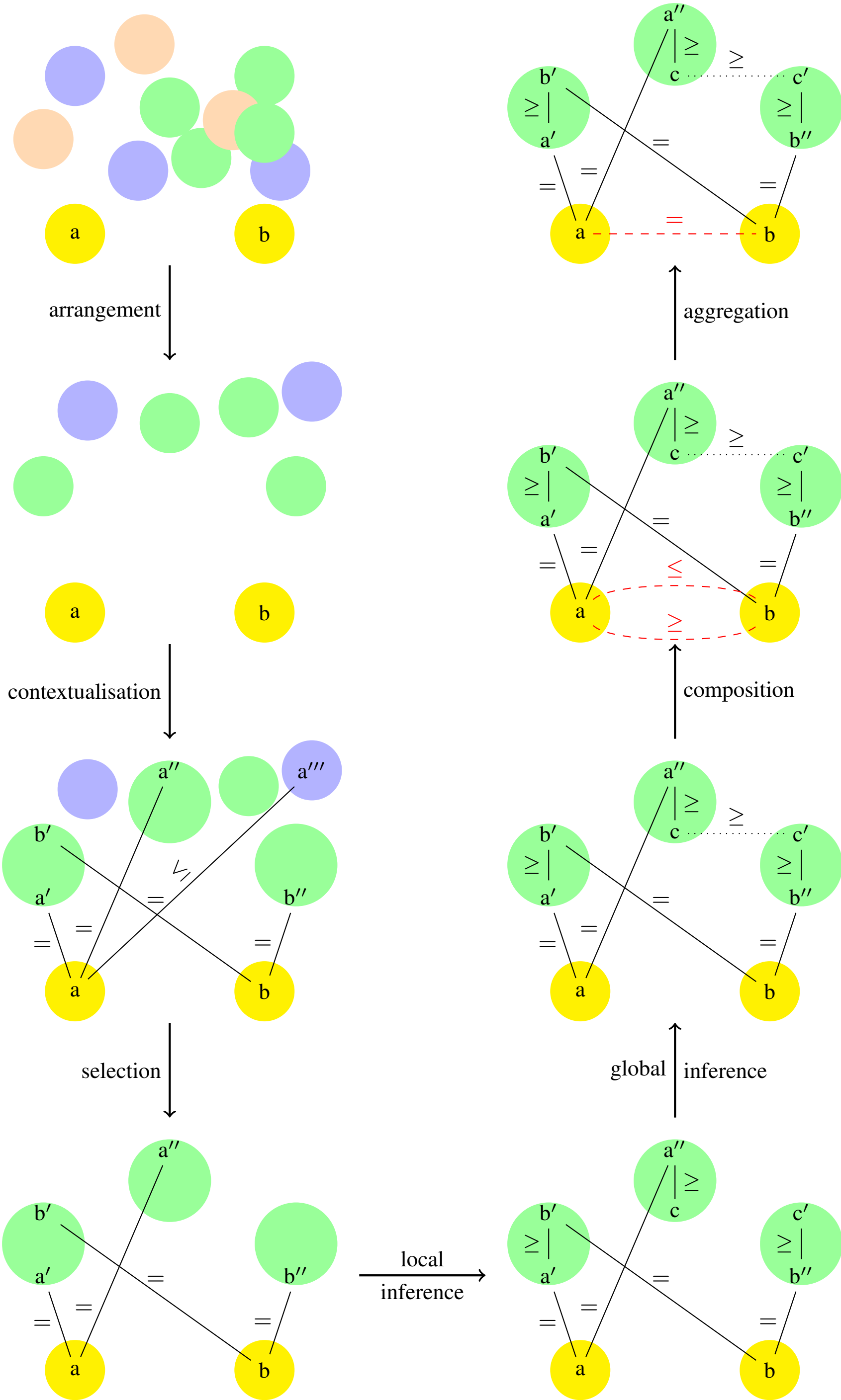

Fig. 3 The different steps of context-based matching.

ing that can be used for instantiating differently each (optional) step.

We may see context-based matching under a fully logical point of view: local and global inference are replaced by entailment tests and composition and aggregation are replaced by logical deduction. In such a case, beyond anchoring, matching is reduced to reasoning in a network of ontologies. Hence, when the technology of reasoning in networks of ontologies will be fully developed, it will be possible, in principle, to reduce the seven steps to anchoring and reasoning. Matchers such as LogMap [17] currently apply this, but only between the two ontologies to match.

Such a framework is intellectually very seducing and mostly compatible with the framework proposed above. Indeed, local inference, relation composition and relation aggregation are approximations of their logical counterpart. Only global inference may be too local for fully approximating entailment in a network of ontologies.

3 Path-driven context-based matching

A new version of Scarlet, named Scarlet 2.0, has been developed along the framework of the previous section. Its characteristics are as follows:

- it still takes advantage of Watson [4, 3] giving access to the ontologies of the Web;
- like the initial Scarlet, it uses intensively a path traversal strategy,
- it uses algebras of relations for expressing the relationships between concepts,
- it offers precise parameterisation, so as to study the influence of their values.

We describe this approach as path-driven because the implementation uses the notion of paths, i.e., it considers ontologies and alignments as graphs whose ontology entities are the nodes and the statements and correspondences are the edges. In this setting, matching two concepts consists of (a) finding a path in this graph between them, and (b) computing the relation carried by this path. For instance, in Figure 1, there are two paths, one of which is agrovoc:Beef $=$ tap:Beef $\leq$ tap:MeatOrPoultry $\leq$ tap:ReadMeat $\leq$ tap:Food $=$ nal:Food. The composition of the relations in the edges of this path yields $\leq$ as the relation between agrovoc:Beef and nal:Food.

The reason for considering the same restricted framework as Scarlet is that it is possible to control precisely the way the algorithm explores the search space (through ontology selection or limitation of its exploration). Introducing more sophisticated methods, either for anchoring or for inferring, remains mostly possible. We avoided it in order to obtain clear initial observations in the presence of simple methods.

3.1 General overview and parameters

We describe below the techniques implemented in Scarlet 2.0 with respect to the framework of Section 2.3. The parameters governing the behaviour of the system are identified (in *italics*) and their further values are provided in Table 1.

Ontology arrangement does not do any preselection and potentially considers all ontologies from the Web as provided by Watson.

Contextualisation, uses a simple matching method. This step is parameterised by the ontology *matching method* used for anchoring. It does not take advantage of confidence measures. Scarlet 2.0 can use any matcher implementing the Alignment API[6]. In this experiment, we will only use a simple token-based string equality (each label is reduced to a set of tokens which are compared with string equality).

Ontology selection is governed by two thresholds on the number of anchors that have to be found between the ontologies to be matched. A first parameter called *minimum local anchors*, is the minimal number of pairs of ontology entities that have anchors in an ontology. A second parameter, *minimum global anchors*, is the minimal total number of anchors found in an intermediate ontology. Obviously, if the first value is greater than or equal to the second one, then the second one is useless. If both values are 0, then all ontologies are selected.

Local inference is implemented by local path exploration: it traverses an intermediate ontology to retrieve paths, i.e., sequences of asserted relations between entities. In this implementation, it will attempt at finding paths between anchors, or finding subsumption paths of a given length around anchors (for global inference). This exploration process uses three parameters: (1) the *maximum local path length* for restricting the length of the exploration; (2) the *exploration type* for determining which types of relations are followed; (3) the *selection method* for selecting which paths between a pair of entities have to be retained, e.g., the first one, the shortest one, all of them.

Global inference is implemented by global path exploration, i.e., it generates paths between two concepts of the ontologies to be matched by concatenating various local paths from distinct ontologies, such that the concept at the end of each local path is anchored to the concept at the beginning of the next local path. The *maximum global path length* parameter determines the maximal number of ontologies that may be traversed to return a relation between two entities. If this is 0, then the algorithm is in the case of classical (content-based) ontology matching, and matching will be reduced to anchoring. Like before, the *selection method* indicates which paths are selected, e.g., the first one, the shortest one, or all of them. The graphs traversal algorithm is further presented in §3.2.

Composition In this approach, the *composition method* used for composing relations is the standard composition of algebras of relations (see §3.3 for details).

Aggregation relies on an *aggregation method* for aggregating the relations obtained between the same pair of entities. This is either an algebraic operation such as conjunction or disjunction, e.g., the conjunction between $\leq$ and $\geq$ is $=$, though their disjunction is $<, =, >$, or pop-

[6] `http://alignapi.gforge.inria.fr/`.

hal-00975284, version 1 - 8 Apr 2014

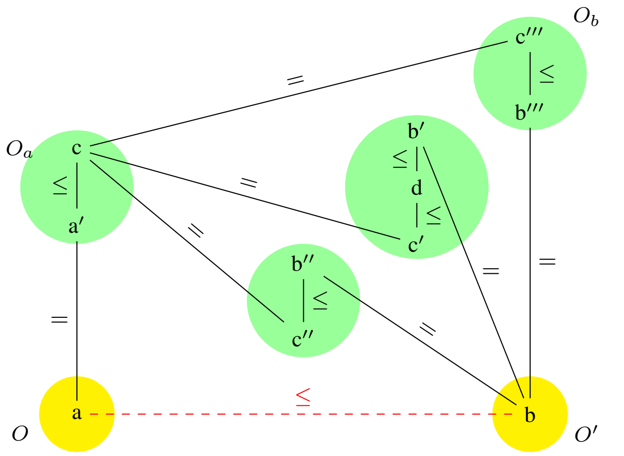

Fig. 4 2-context traversal. Of the three paths found between a and b, two return the $\leq$ relation and the third one returns all the possible relations (Γ).

ularity aggregation, which selects the relation obtained from the most paths.

Table 1 summarises the parameters identified at each step of this process and the different values that they can take. It also provides approximate values for reproducing the original Scarlet strategies.

We present in more detail three aspects of this procedure: graphs traversal (§3.2), relation composition and aggregation using algebras of relations (§3.3), and minimal path reduction (§3.4).

3.2 Global inference through context traversal

For all pairs of concepts for which a correspondence could not be found in any of the intermediary ontologies used during the context-matching operation, global inference can connect the paths obtained in several context ontologies. We call:

0-context traversal content-based matching;
1-context traversal context-based matching using only one context ontologies;
n-context traversal context-based matching using at most n intermediate ontologies.

We describe the behaviour of 2-context traversal, which traverses two intermediary ontologies. Given two concepts $a \in o$ and $b \in o'$ and their respective set of intermediary ontologies O_a and O_b to which they are anchored, for any pair of ontologies $\langle o_a, o_b \rangle \in O_a \times O_b$, the 2-context traversal algorithm looks if there exists an anchor $\langle c_a, =, c_b \rangle$ between them such that:

1. c_a is either a subsumer or a subsumee of $a \in o_a$, found by exploring o_a until a given path length;
2. c_b is either a subsumer or a subsumee of $b \in o_b$, found by exploring o_b until a given path length.

Once such an anchor is found, the path selection, composition and aggregation for the pair put in correspondence are applied in the usual way. A visual sketch of the algorithm is depicted in Figure 4.

The difference between the 2-context traversal strategy and strategy $S2$ of Scarlet lies in that the intermediate concept c is now searched through the whole length of a local path instead of stopping at the direct subsumers or subsumees of the concepts to be matched.

3.3 Composing paths and aggregating correspondences

One of the benefits of an approach exploring different paths for finding relations is that it may return several possible relations between two entities. Such relations may confirm or contradict each other and this has to be considered in the aggregation step. For this reason, we use an algebra of alignment relations [10] that structures the set of possible relations.

Such an algebra of relations is based on a set of jointly exhaustive and pairwise disjoint relations Γ. This means that for any pair of concepts, there exists exactly one relation in Γ, which characterises their relative positions. Algebras of relations allow for expressing uncertainty through the use of subsets of Γ, which are interpreted disjunctively. For instance, the often used relation $\leq$, standing for "subsumed by" or "equals to", really stands for $\{<, =\}$, the disjunction of "subsumed by" and "equals to". The complete lattice of the 31 disjunctive relations of such an algebra is reported in [10]. For our purposes, some of them have been named. Table 2 shows the complete list of named relations and some of the disjunctive combinations, along with their symbol, a description of their interpretation, and the short label used to refer to them in the rest of the paper.

The benefits of algebras of relations is that they provide well-defined aggregation operators: conjunction or disjunction of such relations correspond to set intersection and union, respectively. Hence, the relations obtained through traversing the graph of ontologies may be aggregated with:

conjunction if we consider that each path provides an exact, but non precise, relation and that several paths contribute precising it. When the conjunction gives the $\emptyset$ relation, the resulting correspondence is inconsistent.
disjunction if we consider that each path provides a possible relation without excluding the others.

An alternative aggregation method, independent from the algebra, is the popularity method, which retains the most frequent relation in the set of correspondences between a pair of entities. If several relations have the same popularity, then they are disjunctively aggregated.

hal-00975284, version 1 - 8 Apr 2014

Step	Parameter	Value	$S1$	$S'1$	$S2$
Arrangement	ontologies	**web**	$\surd$	$\surd$	$\surd$
		upper-level ontologies			
		specific domain ontologies			
		specific ontology			
Contextualisation	matching method	a matching method (**=token based similarity**)	string equality		
Selection	minimum local anchors	positive integer (**=0**)	0	0	0
	minimum global anchors	positive integer (**0-10**)	0	0	0
Local Inference	**maximum local path length**	positive integer (**[0..4]**)	∞	∞	1
	exploration type	**subsumption**	$\surd$	$\surd$	$\surd$
		disjointness			
		complete			
	selection method	**all**	$\surd$		$\surd$
		first		$\surd$	
		shortest			
Global Inference	**maximum global path length**	positive integer (0,**1**,**2**)	1	1	2
	selection method	**all**	$\surd$		
		first		$\surd$	
		shortest			
Composition	composition method	functional			
		order-based			
		relational	$\surd$	$\surd$	$\surd$
Aggregation	**aggregation method**	none	$\surd$		$\surd$
		conjunctive			
		disjunctive			
		popularity			

Table 1 List of the possible parameters at each step, whose combination generates a new matcher. Bold parameters are those which will vary in this study; bold values indicate either the (default) value that is used throughout the study or the different values experimented.

Set of relations	Short Label (symbol)	Description
$=$	equiv ($=$)	equivalence relation
$<$	subClass ($<$)	strict subsumption relation
$>$	superClass ($>$)	strict inverse subsumption relation
$\between$	overlaps ($\between$)	overlaps relation
$\perp$	disjoint ($\perp$)	disjoint relation
$>,=$	subsumesOrEqual ($\geq$)	subsumes or equivalent relations
$<,=$	subsumedOrEqual ($\leq$)	is subsumed or equivalent relations
$>,\between$	subsumesOverlap	subsumes or overlaps relations
$<,\between$	subsumedOverlaps	is subsumed or overlaps relations
$>,\between,\perp$	notSubsumed ($\not\leq$)	is not subsumed relation
$<,\between,\perp$	notSubsumes ($\not\geq$)	does not subsume relation
$>,<,\between,=$	notIncompatible ($\not\perp$)	not disjoint relation
$\dots$		other combinations obtained by disjunction or conjunction
$<,>,\between,=,\perp$	all (Γ)	all relations

Table 2 Relation symbols that may result from a composition or aggregation operation for the algebra of alignment relations. The first part of the table features the 5 base relations between concepts.

Algebras of relations also provide a composition operation ($\cdot$), usually based on a table. For instance, $\{>,=\}\cdot\{>\}$ is $\{>\}$ and $\{>,=\}\cdot\{<\}$ is $\not\perp$. This operation is important in context traversal. These traversals return paths which carry sequences of relations between concepts. The composition operator reduces this sequence to a relation preserving as much information as possible.

For instance, a real path found by the system is:

BodyOfWater $=$ BodyOfWater $\geq$ FreshWaterLake $\leq$ Lake $=$ Lake

whose composition brings to the notIncompatible relation ($\not\perp$). Intuitively, this means that if BodyOfWater and Lake are two concepts with a sub-concept in common, viz., FreshWaterLake, they should not be disjoint (because in this algebra concepts are assumed non empty). Thanks to composition, the information that the two concepts are not disjoint is preserved.

hal-00975284, version 1 - 8 Apr 2014

hal-00975284, version 1 - 8 Apr 2014

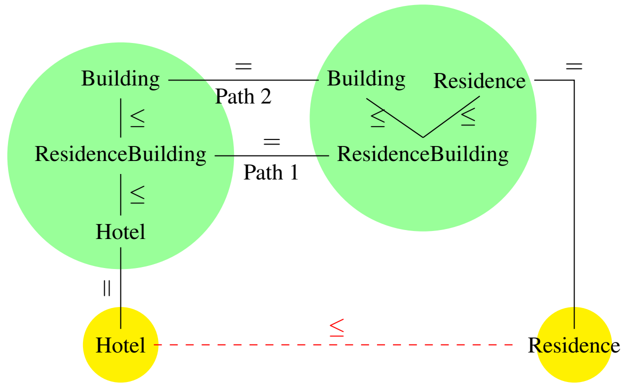

Fig. 5 Two alternative paths between the same pair or concepts. One path is the extension of the other.

3.4 Minimal path reduction in path concatenation

During the path exploration procedure, it may happen that paths are extensions of shorter paths. Figure 5 shows an example of two such paths for the same pair of concepts, one path (Path 2) being the extension of the other (Path 1). They are:

1. Hotel $=$ Hotel $\leq$ ResidenceBuilding $=$ ResidenceBuilding $\leq$ Residence $=$ Residence, which by composition $(= \cdot \leq \cdot = \cdot \leq \cdot =)$ yields $\leq$
2. Hotel $=$ Hotel $\leq$ ResidenceBuilding $\leq$ Building $=$ Building $\geq$ ResidenceBuilding $\leq$ Residence $=$ Residence, which by composition $(= \cdot \leq \cdot \leq \cdot = \cdot \geq \cdot \leq \cdot =)$ yields Γ.

If both paths are retained, they will be aggregated:

– conjunction gives the final correspondence $\langle$Hotel $\leq$ Residence$\rangle$;
– disjunction gives the final correspondence $\langle$Hotel Γ Residence$\rangle$;
– popularity based aggregation would result in both final correspondences $\langle$Hotel $\leq$ Residence$\rangle$, and $\langle$Hotel Γ Residence$\rangle$, as they are equally occurring. In this case a disjunction of both the final relations is computed, the final correspondence being $\langle$Hotel Γ Residence$\rangle$.

So, there is a risk of having non precise correspondences if all such paths are gathered and it is preferable to select them. There may be several ways to do it:

– select the one that goes across less ontologies: both paths traverse two ontologies, so in this example both paths 1 and 2 would be selected;
– select the shortest one: the former is the shortest, so in this example the path selected would be path 1;
– select the most precise one: the former is the most precise one because $\leq \subseteq \Gamma$.

In our case, a procedure for always selecting the shortest path between the source and the target concepts is applied. Hence, in our experiments the path selected would be Path 1.

4 Experiments

A benefit of having a parameterised matcher is to be able to evaluate the various configurations of this matcher and to determine in which context they perform well. We describe below a series of tests that have been used for evaluating these parameters.

4.1 Data

Since the proposed context-based matcher is able to discover more expressive correspondences than those provided by classical ontology matching benchmarks such as OAEI, we build a specific test. It is based on two ontologies and a reference alignment between them. The two ontologies are:

Places_selection.owl: a fragment of the Schema.org[7] ontology, in particular of the sub-module under the more general concept *Places*, that has been reduced by hand. It contains 38 concepts;
geofile_onto_rev.owl: a reduction by hand of the geofile-ont.owl[8]. It contains 35 concepts.

The reduction by hand of the original size of the two ontologies was necessary either for matching two ontologies with a comparable number of concepts and for handling a manual reference alignment. A reference alignment between *Places* and *Geofile* was created by ourselves. It contains 94 correspondences classified as follow:

– 8 correspondences with equivalence relation;
– 43 correspondences with subsumes relation;
– 43 correspondences with subsumed-by relation.

The ontologies used for the experiments and the reference alignment are available online[9].

4.2 Relational precision and recall

As stated in [9], classical precision and recall used for evaluating matching systems are defined on sets and do not take the semantics of alignments into account. For instance, with the classical precision and recall measures, an evaluated correspondence $\langle a, \leq, b\rangle$ will be considered as a false positive against a reference correspondence $\langle a, =, b\rangle$.

[7] `http://schema.org/docs/schemaorg.owl.`
[8] `http://www.daml.org/2001/02/geofile/geofile-ont.`
[9] `http://www.disi.unige.it/person/LocoroA/download/tests/scarlet20/.`

Since the returned alignments are based on algebras of relations, we propose to use more appropriate precision and recall called relational precision and recall.

In our experiments, we adopt an asymmetric interpretation of evaluated and reference alignments. For any pair of concepts which does not belong to the reference alignment, we assume that the relation holding between them is $\perp$, i.e., they are disjoint. Nevertheless, if a pair of concepts does not belong to the evaluated alignment, we consider that the relation holding between these concepts is Γ, which means that the relation is unknown.

In order to not artificially increase precision or recall, we only retain correspondences which are:

- not Γ for the evaluated alignment;
- not $\perp$ for the reference alignment.

Given that a correspondence in an alignment A may have the form:

$$\langle e_a, R_a, e'_a\rangle \text{ with } R_a \subseteq \Gamma$$

and a correspondence of the reference alignment R has the form:

$$\langle e_r, R_r, e'_r\rangle \text{ with } |R_r| = 1$$

Relational precision and recall is based on the following formulas:

$$P(A,R) = \frac{|\{\langle e, R_a, e'\rangle \in A; \langle e, \{r_r\}, e'\rangle \in R \;\&\; r_r \in R_a\}|}{|A|}$$

$$R(A,R) = \frac{\sum_{\langle e,R_a,e'\rangle \in A; \langle e,\{r_r\},e'\rangle \in R \;\&\; r_r \in R_a} \frac{1}{|R_a|}}{|R|}$$

These are both instances of the equation [7]:

$$R(A,R) = \frac{\sum_{\langle e,R_a,e'\rangle \in A; \langle e,R_r,e'\rangle \in R} \frac{|R_r \cap R_a|}{|R_a|}}{|R|}$$

Because it does not aggregate correspondences, the "No aggregation" strategy always have a larger number of correspondences. Such additional correspondences prevent from interpreting the alignment and artificially increase precision [7]. Hence we will not compute precision and recall on these alignments.

Stemming from these measures, each correspondence in an alignment is classified with respect to the reference alignment, as follows:

correct correspondence, when $R_a = \{r_r\}$;
non precise correspondence, when $\{r_r\} \subset R_a$;
conflicting correspondence, when $\{r_r\} \cap R_a = \emptyset$;
incorrect correspondence, when the reference alignment does not contain correspondence between the same two entities.

For instance, if we consider a reference alignment R containing two correspondences $\{\langle$Place $\{>\}$ GeographicArea$\rangle$, $\langle$City $\{=\}$ City$\rangle\}$, then with respect to this reference alignment, the correspondence:

- $\langle$Place $\{>\}$ GeographicArea$\rangle$ is correct;
- $\langle$Place $\{>, =\}$ GeographicArea$\rangle$ is non precise;
- $\langle$Place $\{=\}$ GeographicArea$\rangle$ is conflicting;
- $\langle$City $\{>, =\}$ GeographicArea$\rangle$ is incorrect.

The incorrect category is not strictly needed. Indeed, if the absence in the reference means $\perp$, then the incorrect correspondences may be interpreted according to the three other categories. However, this category is useful for analysing the results.

It is noteworthy that this way of defining precision and recall is grounded logically, i.e., precision is approximating correctness and recall is approximating completeness. However, because of the disjunctive interpretation of relations, this leads to counterintuitive consequences: the larger the relation, i.e., the most imprecise, the more likely it will be correct, because the disjunction will be a consequence of the reference alignment. For example, $\langle$Place Γ GeographicArea$\rangle$, although being imprecise, is still correct, as $\leq \subseteq \Gamma$. Hence, precision will be higher if correspondences are less precise. This is, in fact, a problem with the name of "precision", but the behaviour is the one expected.

4.3 Questions and parameters

Here are the basic questions that we expect to answer through these experiments. We express them through the parameter settings that have been used for the experiments according to Table 1.

Q1 What is the influence of the selected ontologies and, in particular, is pruning the set of ontologies a priori based on a minimal number of anchors worthwhile? To answer this question, we compare the results obtained with no selection (minimum local and global anchors set to 0) and those obtained by restricting the candidate ontologies to those containing at least 10 anchors globally.

Q2 What is the impact of the path length between intermediary ontologies? We run different series of tests, with different maximum local path length varying from 0 to 4.

Q3 What is the impact of using paths across more intermediary ontologies? To answer this question the experiments have been run by using 1-context traversal as well as 2-context traversal (this last one just for all those pairs of concepts for which the exploration of single intermediary ontologies did not return any correspondence). For achieving this, we have set the maximum global path

hal-00975284, version 1 - 8 Apr 2014

length to 1 and 2, i.e., traversing 1 and 2 intermediate ontologies, respectively.

Q4 What is the impact of the aggregation methods? We have used all possible values for aggregation methods: none, conjunctive, disjunctive, and popularity.

In all the experiments, the ontology selection is the Web, i.e., Watson is used for retrieving any relevant ontology on the Web, the anchoring method is token-based, e.g., concepts are selected as anchors when at least one word from their local name is contained in the local name and label of the concept in the source ontology, and the composition method is relational.

In each case, the accuracy is measured in terms of precision, recall and F-measure evolution, that is as a function that measures them when varying the parameter settings.

4.4 Material

As briefly introduced in Section 4.3, Table 1 reports the whole of Scarlet 2.0 parameters settings, which are configurable at each step of the context-based approach, and whose combination generates a new possible matcher. Bold parameters in Table 1 are those exploited in the experiments conducted for this research study. In particular, the combination of all the configurations chosen for this prototypical study yields to 80 different matchers, which are a subset of those reported in Table 1, and whose main parameter values are detailed in Table 3. A summary of the selection criterion for the present experimental setting, along each context-based step, with a brief motivation, is reported below, in form of descriptive list. Such list unifies the information reported in Table 1 and Table 3, in order to provide a complete view of the choices adopted for these experiments:

- *Arrangement* parameter *ontologies* has been set to *Web*, in order to explore as many intermediate ontologies as possible;
- *Contextualisation* parameter *matching method* has been set to *token-based*, in order to keep the anchoring phase as simple and correct as possible;
- *Exploration* parameter *minimum global anchors* has been set to *0*, which corresponds to "All Ontologies" and to *10*, which corresponds to "Selected Ontologies", in order to test the performance of using all ontologies or a possible user-selection of ontologies;
- *Local inference* parameters *maximum local path length* (which corresponds to "PL"), *exploration type*, and *selection method* have been set to *0, 1, 2, 3, 4*, *subsumption*, and *all*, respectively, in order to explore all possible local paths, along the subsumption relation, while testing the performance of different path lengths;
- *Global inference* parameters *maximum global path length* and *selection method* have been set to *1, 2* and *all*, respectively, in order to test the performance of 1-context traversal (contextualisation with 1 intermediate ontology), as well as of 2-context traversal (contextualisation with 2 intermediate ontologies);
- *Composition* parameter *composition method* has been set to *relational*, in order to test the relational composition of different context-explored local paths;
- *Aggregation* parameter *aggregation method* has been set to *none, conjunctive, disjunctive, popularity*, in order to test each aggregation method.

maximum global path length	**minimum global anchors**	**maximum local path length (PL)**	**aggregation method**
1	0	0	none
(1-context traversal)	(All ontologies)	1	conjunctive
2	10	2	disjunctive
(2-context traversal)	(Selected ontologies)	3	popularity
		4	

Table 3 List of test value combinations used in the experiments.

Hence, a total of 80 different alignments is obtained from matching the two ontologies presented in §4.1 against the above listed configuration criteria[10].

In the next section, we present the results of these experiments and interpret them.

5 Results and discussion

Experiment results are first considered globally (§5.1) in order to answer the questions in §4.3. Then we proceed to analyse the results from two further perspectives:

1. the type of correspondences found by the various strategies (§5.2);
2. the kind of relations used in these correspondences (§5.3).

5.1 Global analysis

Table 6 summarises the maximum precision and the maximum recall obtained for each group of experiments, for each path length (PL), and for each of the two approaches "All Ontologies" and "Selected Ontologies" (see Table 3 for the parameter values that correspond to such approaches).

As 2-context traversal does not change the results with respect to 1-context traversal, in what follows (with the exception of Table 7), we only report those relative to the

[10] The tests have been conducted in parallel on two different machines: a Toshiba Notebook, with Windows 7 64-bit OS, Processor Intel Core i5 2.27 GHz, 4 GB RAM and an HP Pavillon Notebook with Intel Core Duo T2250 processor, 1.73 GHz of clock, 2 GB of RAM, and Windows 7 32-bit OS.

hal-00975284, version 1 - 8 Apr 2014

part of experiments where we set the *maximum global path length* to 1, i.e., 1-context traversal. Section 5.2.2 reports a dedicated experiment discussing the (non) impact of 2-context traversal.

5.1.1 Baseline

The main scope of this paper is to compare several variants of context-based matching and to observe the influence of different parameters. However, it is worth having a preliminary comparison with other available approaches. These comparisons do not aim at claiming that context-based matching is the best method for any type of matching problem. They rather illustrate that on this particular pair of ontologies, that has been selected to evaluate parameters of context-based matching, this approach brings benefit.

For that purpose we compared:

– two very simple matching techniques available in the Alignment API: edna, which computes the edit distance on names, and substring, which computes the substring distance on names;
– two efficient and sophisticated matchers which performed very well in the OAEI 2012 evaluation: LogMap [17], which relies heavily on semantic techniques, and YAM++ [19], which combines several similarities in an adaptive way,
– Scarlet 2.0 baseline, which uses the simplest parameters: local path length = 0, hence no context exploration.

Tool / Method	Precision	F-Measure	Recall	Precision$^=$	F-Measure$^=$	Recall$^=$
LogMap	1.0	.10	.05	1.0	.77	.63
YAM++	.86	.12	.06	.86	.80	.75
Edna	.31	.09	.05	.31	.42	.63
Substring	.5	.11	.06	.50	.60	.75
Scarlet 2.0 baseline	1.0	.10	.05	1.0	.77	.63

Table 4 Precision, recall and F-measure obtained by matching the ontologies with LogMap, YAM++, edna and substring, and comparing them with Scarlet baseline (PL0). The first results are those obtained with the relational measure against the reference alignment, the second results (marked $^=$) are obtained with standard measures against the reference alignment reduced to the equivalence correspondences.

Table 4 shows the results of these methods for the setting used in the current experiment, i.e., relational measures presented in §4.2 and the reference alignment containing various relations in correspondences. It also displays the results with standard precision and recall and the reference alignment reduced to those correspondence having the equivalence relation. This second measure is assumed to be more favourable to content-based matchers. In both cases, the results of the baseline Scarlet are comparable to those of the best content-based matchers. Only YAM++ is slightly better than all the other matchers.

Table 5 shows for each approach the details of found correspondences, which should be compared to those of Table 11 to have a general idea of the differences. This illustrates the wide extent of new correspondences that context-based matching considers.

Correspondence	LogMap	YAM++	edna	substring	Scarlet baseline
State = State	=	=	=	=	=
Hospital = Hospital	=	=	=	=	=
City = City	=	=	=	=	=
Airport = Airport	=	=	=	=	=
Country = Country	=	=	=	=	=
BodyOfWater = WaterArea				=	
CivicStructure < Infrastructure			=	=	
SportsActivityLocation < Location				=	
Place > Lake			=		
Place > GeographicArea		=			
LakeBodyOfWater = Lake		=			

Table 5 A comparative analysis between Scarlet 2.0 baseline and other matchers. For each correspondence in the reference alignment, with the correct relation, the table reports whether each tool also found a correspondence, and the relation found.

Time performances are not comparable because the matchers have not been launched in comparable conditions. However, they are clearly at the advantage of content-based matchers. See §5.1.4 and 5.1.5 for a discussion about the runtime performances of our implementation.

5.1.2 Selecting ontologies increases precision, but decreases recall and F-measure (Q1)

From Figure 6 to 8, we observe that selecting the ontologies decreases recall to the point that F-measure decreases. As Scarlet already shown [21]: using all the Web increases recall. Results also show that precision increases when we select ontologies and when the path length is greater than 1. Selecting ontologies allows for retaining only the more relevant ontologies.

5.1.3 Path length increases recall to the point that it increases F-measure (Q2, Q3)

From Figure 6 to 8, we observe, in most of the cases, that path length increases recall to the point that F-measure increases. This is especially true from PL 0 to PL 1 and from

hal-00975284, version 1 - 8 Apr 2014

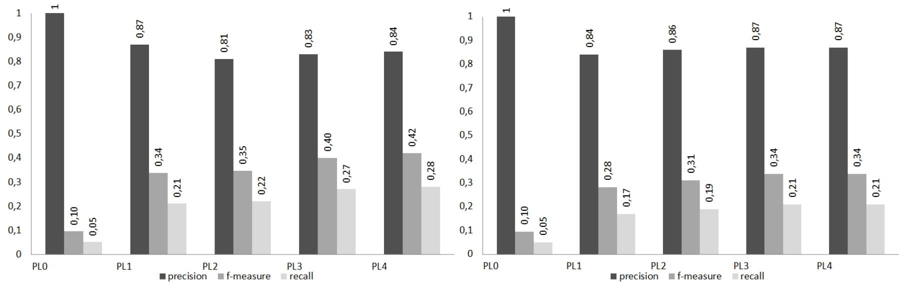

Fig. 6 precision and recall for "All Ontologies" (1-context and 2-context traversal) and "Selected Ontologies" (1-context and 2-context traversal) and conjunctive aggregation of alignments.

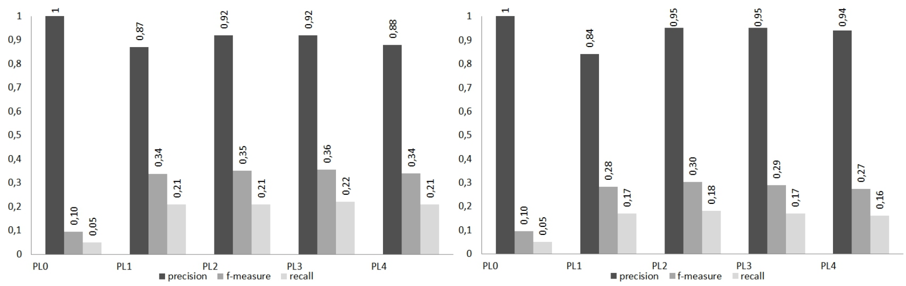

Fig. 7 precision and recall for "All Ontologies" (1-context and 2-context traversal) and "Selected Ontologies" (1-context and 2-context traversal) and disjunctive aggregation of alignments.

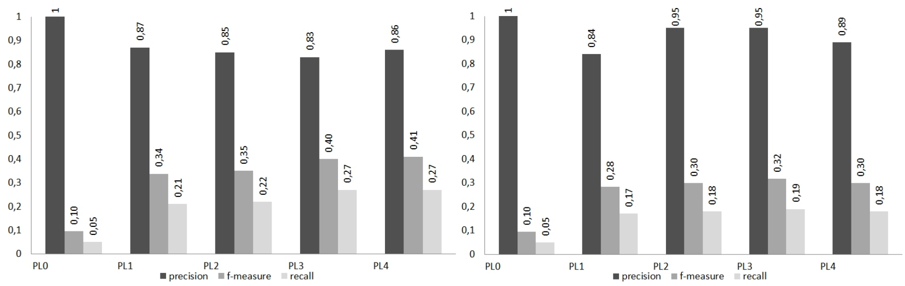

Fig. 8 precision and recall for "All Ontologies" (1-context and 2-context traversal) and "Selected Ontologies" (1-context and 2-context traversal) and popularity aggregation of alignments.

Ontologies	Measure	**PL0**	**PL1**	**PL2**	**PL3**	**PL4**
ALL	Max Prec	1	.87	.92	.92	.88
	Max Rec	.05	.21	.22	.27	.28
SELECTED	Max Prec	1	.84	.95	.95	.94
	Max Rec	.05	.17	.19	.21	.21

Table 6 Maximum Precision and Maximum Recall obtained per path length, for each of the two approaches "All Ontologies" and "Selected Ontologies".

PL 2 to PL 3. Selecting ontologies smoothes this trend. These results confirm the relevance of such context-based matching approach.

5.1.4 Selecting ontologies decreases processing time (Q1)

Runtime performances of the systems were compared, in terms of how long the version with all the ontologies did take, if compared to the version with only selected ontologies (see Table 7 for details). The results show a better time performance for the approach that operates on a selection of ontologies. For each experiment of the series named "All Ontologies", 327 intermediate ontologies were searched in Watson and exploited in the context-matching algorithm, while 41 intermediate ontologies were those applied to the "Selected Ontologies" series. The main differences in the results time performance are only due to the number of total ontologies used for each approach (see also next section).

5.1.5 Increasing path length, and traversal of ontologies, does affect time performances only minimally (Q2, Q3)

Other aspects that are observable in Table 7 are relative to the 2-context traversal procedure, and the path length increase on the correspondence found. The time increase remains moderate. This is particularly true when compared with the time difference between selection of ontologies and no selection of them. The calls to the Watson search service, the retrieval, and the caching of an intermediate ontology are

hal-00975284, version 1 - 8 Apr 2014

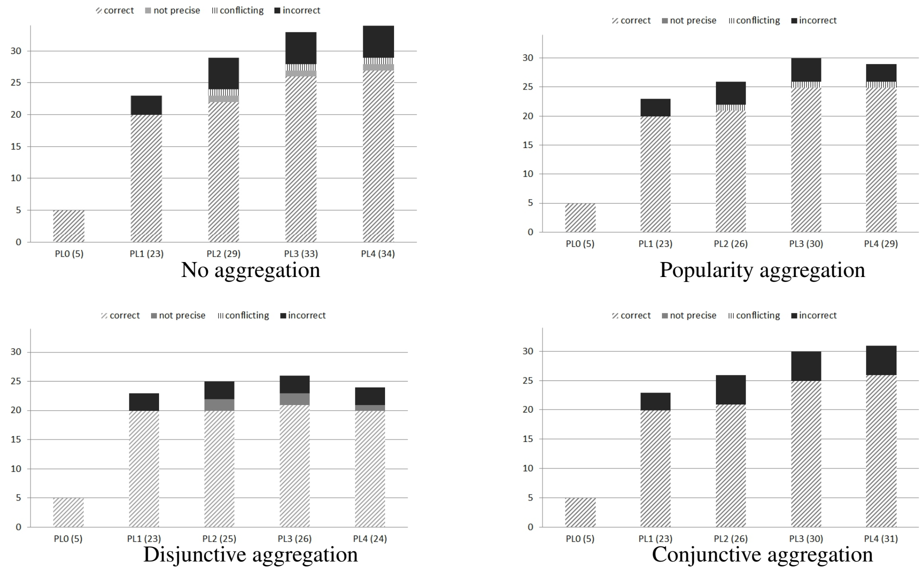

Fig. 9 Number of correspondences found distributed in correct, non precise, conflicting and incorrect with respect to the reference alignment for "All Ontologies, 1-context traversal". The non precise correspondences are in fact the non degenerated imprecise correspondences, i.e., those which are not the whole Γ.

	PL0	PL1	PL2	PL3	PL4
ALL ONTO					
1-context traversal	98	102	106	116	123
2-context traversal	102	106	110	119	128
SELECTED ONTO					
1-context traversal	12	15	17	21	22
2-context traversal	14	18	18	29	42

Table 7 Time Performance in minutes for "All Ontologies" and "Selected Ontologies" experiments.

the more computationally expensive operations of the algorithm. Such a huge penalty was already noted in [20] and could be improved by caching.

5.1.6 Aggregation methods are influential, though only marginally (Q4)

From Figure 6 to 8, we observe the following rankings among the aggregation methods:

- precision: disjunction $>$ popularity $>$ conjunction;
- recall: conjunction $>$ popularity $>$ disjunction;
- F-measure: conjunction $>$ popularity $>$ disjunction.

Disjunctive aggregation outperforms other methods on precision. Since disjunctive aggregation increases the imprecision of correspondence relations, it obtains the best results in terms of precision (see explanation in §4.2). According to precision definition, the more imprecise the relation, the more chances it has to be counted as a true positive by precision. For example, given two correspondences such as:

- $\langle$Place $\{>, =\}$ geographicArea$\rangle$
- $\langle$Place $\{<, =\}$ geographicArea$\rangle$

their aggregation through disjunction, i.e., $\{>, =\} \cup \{<, =\} = \{<, =, >\}$, yields to the final correspondence $\langle$Place $\{<, =, >\}$ geographicArea$\rangle$. Given the correspondence in the reference alignment, i.e. $\langle$Place $\{>\}$ geographicArea$\rangle$, it results that the final correspondence is "non precise" (it contains the relation $\{>\}$).

In counterpart, conjunctive aggregation tends to reduce correspondence broadness, hence augmenting the chances of being incorrect. By following the same example as above, and aggregating the two correspondences relations through conjunction the result is $\{>, =\} \cap \{<, =\} = \{=\}$. The final correspondence, i.e., $\langle$Place $\{=\}$ geographicArea$\rangle$, is then conflicting, i.e., not correct. An opposite example where conjunction reduces correspondence imprecision by yielding a correct correspondence is instead the following:

- $\langle$Place $\{<, >\}$ geographicArea$\rangle$
- $\langle$Place $\{>, =\}$ geographicArea$\rangle$

which becomes, once aggregated with conjunction, the correct correspondence, i.e., $\{<, >\} \cap \{>, =\} = \{>\}$, hence $\langle$Place $\{>\}$ geographicArea$\rangle$.

In term of recall and F-measure, conjunction has better results than disjunction and popularity.

Even if popularity has an intermediate position, it cannot outperform conjunction. See the example above where conjunction brings the correct correspondence, whereas popularity would disjunctively aggregate both relations, thus yielding a non precise correspondence.

hal-00975284, version 1 - 8 Apr 2014

Correspondence	Number of paths
$\langle$Canal Γ Airport$\rangle$	3
$\langle$Airport $<$ InternationalAirport$\rangle$	3
$\langle$BodyOfWater Γ Airport$\rangle$	1
$\langle$Canal Γ Sea$\rangle$	2
$\langle$Museum Γ School$\rangle$	2
$\langle$Mountain Γ City$\rangle$	1
$\langle$BodyOfWater Γ Sea$\rangle$	4
$\langle$Mountain Γ WaterArea$\rangle$	1

Table 8 Correspondences and number of paths found for each of them with the 2-context traversal only.

5.2 Correspondence-level analysis

We took advantage of algebras of relations for performing a more precise analysis of the results. Separating the returned correspondences into correct, non precise, conflicting and incorrect (see Section 4.2 for details of this classification) allows for evaluating better the effects of parameters.

Figures 9 shows how many correspondences have been found based on this classification, for "All Ontologies", for each aggregation method, and 1-context traversal. Table 9 gives the numbers of correspondences having relation Γ, which have been discarded after the aggregation step. These correspondences have been removed from alignments because they would artificially increase precision and recall values. Nevertheless, having these numbers helps understand which method generates imprecise correspondences.

The expectation with respect to the aggregation methods is that conjunctive aggregation increases conflicts and disjunctive aggregation increases non precise correspondences at the expense of correct correspondences. To show this with an example, consider the two correspondences:

– $\langle$City $\{<,=\}$ Town$\rangle$
– $\langle$City $\{<\}$ Town$\rangle$

whose aggregation by conjunction yields to a conflicting correspondence, i.e., $\{<,=\} \cap \{<\} = \{<\}$, when compared to the correct correspondence of the reference alignment, i.e., $\langle$City $\{=\}$ Town$\rangle$, whereas it becomes a non precise correspondence if aggregated through disjunction ($\{<,=\} \cup \{<\} = \{<,=\}$).

A first question concerns the proportion of this effect.

5.2.1 Using all ontologies retrieves more correct correspondences (Q1)

According to Figure 10, the method that uses all ontologies found between 22% and 35% more correct correspondences than those that select ontologies. This trends increases with the length of the path.

The highest number of correct correspondences is given by the approach that uses all ontologies, with conjunctive aggregation, and both 2-context and 1-context traversal, for a total of 26 correct correspondences. This means that, contrary to the intuition, adding more ontologies reduces imprecision when used with conjunctive aggregation.

5.2.2 2-context traversal does not improve the results (Q3)

The 2-context traversal method did not bring any improvement in terms of new correspondences discovered. It had been run only on those pairs of concepts for which no correspondence was found when exploiting 1-context traversal.

In order to evaluate whether the 2-context traversal could be used at least for refinements purposes, an extra experiment was carried out, exploring paths for all pair of concepts, including those for which a local path was already found. The results of this experiment are reported in Table 8. A total of eight correspondences were obtained. So, going through several ontologies may provide new paths between entities. However, if compared to those of Table 11, those tend to be less precise (only one is precise and it is contradictory to the correct one). This can be explained because the knowledge found in two different ontologies is usually less tightly related than the one found in one single ontology.

Since what weakens 2-context traversal is the increasing number of anchors, this reveals that the quality of context-based matching strongly depends on them. It may then be wise to generate anchors with strong confidence, rather than approximate ones.

It would be worth testing the 2-context traversal performance with more sophisticated criteria. For example, exploit it as a second iteration step of the algorithm, or with a better tuning on the initial set of intermediate ontologies, or with an advanced exploration task.

5.2.3 Increasing path length, always increases correct correspondences (Q2, Q3)

From Figure 9, we observe that increasing path length increases the number of correct correspondences. As we have already seen, 2-context traversal does not change anything.

5.2.4 Path length increases non precise and incorrect correspondences (Q2)

From Table 9, we observe that increasing path length increases the number of correspondences having relation Γ (the most imprecise one). This kind of relation appears from PL 2 on, and its amount is more than twice at PL 4. For example the paths:

– BodyOfWater $\geq$ String $\leq$ InanimateString-Natural $\geq$ Sea
– BodyOfWater $\geq$ Lake $\leq$ Individual $\geq$ Ocean $\leq$ Sea

hal-00975284, version 1 - 8 Apr 2014

	PL0	PL1	PL2	PL3	PL4
no agg	-	-	26	37	62
conj	-	-	24	32	54
disj	-	-	26	37	62
pop	-	-	24	34	58

Table 9 Number of correspondences bearing the whole Γ relation obtained after composition and aggregation (for "All ontologies, 1-context traversal")

are long paths (> 2) whose composition yields Γ.

Figure 9 also shows that other kinds of non precise correspondences are introduced from PL 2 on. For example, the path BodyOfWater $\geq$ FreshWaterLake $\leq$ Lake, found at PL2, yields $\not\perp$.

Figures 9 and 10 show that increasing path length also increases the number of incorrect correspondences. In this test, this number becomes stable from PL 2 on, and is neglectable in comparison with the amount of correct correspondences. For example, the paths:

- Continent $\leq$ LandRegion $\geq$ Island
- Continent $\leq$ LandArea $\geq$ Island
- Continent $\geq$ TrueContinent $\leq$ LandBoby $\geq$ Island
- Mountain $\leq$ Protrusion $\leq$ PartiallyTangible $\geq$ AirportPhysical $\geq$ Airport

are all examples of incorrect correspondences (see also Section 5.3.1 for an analysis of incorrect correspondences that result from incorrect ontologies).

5.2.5 Disjunctive aggregation increases non precise correspondences at the expense of correct ones (Q4)

Disjunctive aggregation decreases the number of correct correspondences and increases the number of non precise correspondences (in the technical sense introduced in Section 4.2). From Table 9, we observe that disjunctive aggregation does not introduce more non precise correspondences with Γ relation, when compared to the no aggregation method. Nevertheless, it often happens that, in the raw results (no aggregation), there are both a non precise (Γ) and a correct correspondence for the same pair of entities. In this context, the disjunctive aggregation returns a fully non precise correspondence (with Γ). This leads to decreased correct correspondences. For example, in the two correspondences:

- $\langle$Place Γ geographicArea$\rangle$
- $\langle$Place $>$ geographicArea$\rangle$

the former is non precise and the latter is correct. Once aggregated through disjunction, the resulting relation is less precise, because $\Gamma \cup \{>\} = \Gamma$, hence the final correspondence is $\langle$Place Γ geographicArea$\rangle$.

Furthermore, Figure 9 shows that disjunction, in general, introduces more non precise correspondences, if compared to conjunction and popularity; the figures in Table 9 confirm the trend depicted in the above example: correspondences aggregated through disjunction tend to preserve the ones with the Γ relation, at the expense of correct ones.

5.2.6 Popularity based and conjunctive aggregation methods favor precise correspondences (Q4)

Beside PL 0 and PL 1, where the results for the different aggregation methods are the same, Table 9 and Figure 9 show that, for longer paths, both conjunction and popularity remove many non precise correspondences at the expense of a few correct ones.

In each case, there is no perfect aggregation method that could both reduce incorrect and non precise correspondences while preserving correct ones. Hence, a mix of these strategies may improve the situation. One may consider two particular options:

- select correspondences in function of the length of the path for obtaining them, i.e., using the Scarlet $S1'$ strategy;
- select correspondences depending on the type of relations they provide.

This prompted us to analyse more precisely the relations within correspondences.

5.3 Relation-level analysis

The second option is considered below, where a further analysis is conducted to assess the different kinds of relations found in the experiments, for each different kind of correspondences.

The aim of this analysis is to observe whether it is worth continuing the exploration of ontologies by increasing path length. We analyse the types of relation that are obtained for each path length.

Figure 10 shows, on the left, the results for the "All Ontologies, 1-context traversal" approach, and on the right, those of the "Selected Ontologies, 1-context traversal" approach. They provide the following observations (Q1, Q2, Q3):

- **All equivalences are found in the shortest path, and longer paths do not bring any new equal correspondence.** This sharp trend emerges in correlation between the length of the path and the type of relations as well as their correctness. This is also related to the absence of equivalence statements in our ontologies, with few exceptions that we have not encountered during our experiments. This is certainly due to following mostly subsumption assertions in the retrieved ontologies.

hal-00975284, version 1 - 8 Apr 2014

hal-00975284, version 1 - 8 Apr 2014

– **The subsumption relations are found, for all the results, already at PL 1, but nearly double from PL 3 on.** The highest number of subsumption relations are found at PL 4 (12 and 10 if we consider using all ontologies - 8 if we consider only a selection of ontologies).

Path length	0	1	2	3	4
Correct					
=	+5				
(<)		+7	+2	+3	
(>)		+8		+1	+1
Non Precise					
NewNonPrecise			+1	+1	
Conflicting					
NewConflicting			+1		

Table 10 Number of new correct, non precise and conflicting correspondences per path length.

5.3.1 Correct correspondences are found in shorter path lengths, but additional path lengths bring new correct correspondences (Q1, Q2, Q3)

Starting from these considerations, a further analysis was conducted to assess the status of correspondences found in longer paths. Table 10 summarises this analysis for the "All Ontologies, 1-context traversal, No aggregation" (the ones synthesised in Figure 10).

As expected, the path length with the most new correct correspondences is PL 1, for a total of 15 new correct correspondences with respect to PL 0. However, Table 10 shows that new correct correspondences are also found at PL 2 (+2), PL 3 (+4), and PL 4 (+1). No refinements were found, i.e., path length does not improve the precision of the non precise relations, but instead adds non precise relations, e.g., Γ. On the contrary, precise correspondences may become less precise. In particular, for the subsumption relations, popularity aggregation behaves slightly better than other aggregation methods (it only loses one subsumption correspondence, with respect to the no aggregation, and only at PL 4). In the end, incorrect correspondences do not become correct as paths get longer. This suggests that the only strategy is to accept *more precise* correspondences obtained from longer paths and to conjunct them with those obtained on shorter paths.

Table 11 shows each pair of concepts of the reference alignment for which a path has been found, along with each path length. In addition, the number of paths found (Column ♯Paths) and the number of different final relations found for each pair (Column ♯Rels) are reported. The values are cumulative, i.e., longer path lengths contain all the paths of shorter path lengths.

The last three columns of the table represent:

– the size of the final relations found for each correspondence (colum ♯BaseRels);
– if the paths are different when more than one path is found (Column ≠Path);
– if the paths found have a different sequence of relations (Column ≠SeqOfRel), i.e., $\langle <, =, <, = \rangle$ is the same sequence of relations as $\langle <, =, <, = \rangle$ but not as $\langle <, =, < \rangle$ nor as $\langle <, <, =, < \rangle$ which all provide the same relation $(<)$.

The correct relation, i.e., the one reported between each pair of concepts, is equal to the relation found if the column ♯BaseRels is = 1, or is among the relations reported, if the column ♯BaseRels is > 1.

On a total of 30 unique concept pairs, 23 have more than one path, and for 19 of them the paths found are the same, i.e., the different paths are made of the same concept labels. For 11 of them the paths were different, and for 4 of them the sequence of relations that compose the paths were also different.

Further considerations emerge from Table 11, as well as from the correspondences that are not in the table, i.e., the incorrect ones. They are the following:

5.3.2 Relation between remote entities are found at longer path lengths

Some correspondences are only found at long path lengths. For example, $\langle Place > MedicalFacility \rangle$ is found by exploiting at least path length 3; $\langle Hospital < MedicalFacility \rangle$ is already available in path length 1. $MedicalFacility$ can be considered semantically "closer" to $Hospital$ than to $Place$: the first two concepts describe objects of Medical type at different levels of detail, whereas $Place$ is a more general object (it may describe a $Hospital$ as well as a $City$). With few exceptions, we observe that associations between concepts of large abstraction differences are found by exploring longer paths.

5.3.3 Incorrect correspondences may reveal incorrect ontologies

When examining the incorrect correspondences, some of them are clearly not correct, such as $\langle Store = State \rangle$. Many others are found to encode part-whole relations using subsumption relations, such as $\langle City \leq State \rangle$. This correspondence is not correct because if a city may be part of a state, cities are *not* states. This type of mistake was already noted in [13]. For this reason, the incorrect correspondences were further divided by hand into:

– non expressible with the current algebra of relations, but with other semantic relations such as *partOf* and *hasPart*;

hal-00975284, version 1 - 8 Apr 2014

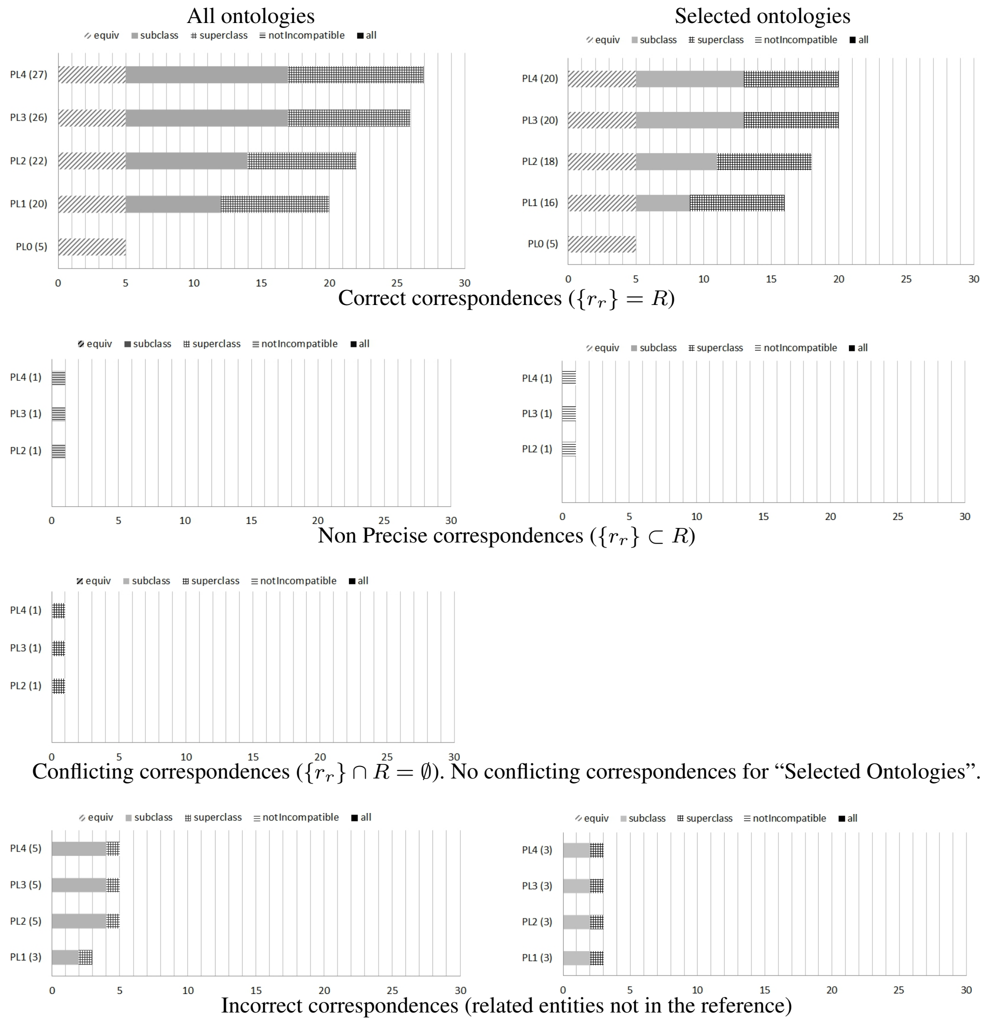

Fig. 10 Number of correct, non precise, conflicting and incorrect correspondences found per path length and kind of relation for "1-context traversal, No aggregation".

– wrong, i.e., relation that should not have been found between the two entities.

We analysed by hand the set of incorrect correspondences obtained by using all ontologies (Figure 10, all ontologies, path length 4) and found that 4 of the 5 incorrect correspondences were non expressible relations. If the non precise correspondences of Table 9 are added to this computation, 38% of them result to be non expressible relations! Such correspondences indicate two things:

– there are relations between entities of these ontologies which cannot be expressed in the algebra of relation that we have used;
– there are ontologies which use the subsumption relation for encoding these (and this is why such relations show up).

6 Conclusion and future work

Context-based matching is based on the assumption that putting ontologies in the context of other ontologies may improve matching. In this paper, we provided a framework identifying important steps of context-based ontology matching and parameters that may influence its behaviour. We have conducted a pinpoint analysis on context-based matching by varying some of these parameters.

These experiments establish general observations on the behaviour of such systems, and confirm what was previously observed:

– Not restricting the considered ontologies provides significantly more correspondences than selecting them a priori and this increases F-measure, although precision decreases.
– Increasing global and local path length also provides more correspondences and increases F-measure; the ef-

hal-00975284, version 1 - 8 Apr 2014

Correspondence			PL0		PL1		PL2		PL3		PL4				
			♯Paths	♯Rels	♯Paths	♯Rels	♯Paths	♯Rels	♯Paths	♯Rels	♯Paths	♯Rels	♯BaseRels	≠Path	≠SeqOfRel
State	=	State	59	1	59	1	59	1	59	1	59	1	1		
Hospital	=	Hospital	21	1	21	1	21	1	21	1	21	1	1		
City	=	City	78	1	78	1	78	1	78	1	78	1	1		
Airport	=	Airport	51	1	51	1	51	1	51	1	51	1	1		
Country	=	Country	96	1	96	1	96	1	96	1	96	1	1		
BodyOfWater	>	Lake			1	1	2	2	3	2	3	2	4	√	√
City	<	Location			5	1	7	1	7	1	7	1	1	√	
BodyOfWater	>	Gulf			2	1	3	1	3	1	3	1	1	√	
State	<	Location			4	1	6	2	6	2	6	2	2	√	√
Place	>	Country			1	1	2	1	2	1	2	1	1	√	
Airport	>	InternationalAirport			3	1	4	1	4	1	4	1	1	√	
Place	>	City			2	1	2	1	2	1	2	1	1		
Place	>	State			2	1	2	1	2	1	2	1	1	√	
Hospital	<	MedicalFacility			2	1	2	1	2	1	2	1	1		
City	<	GeographicalArea			2	1	2	1	2	1	2	1	1		
BodyOfWater	>	Sea			2	1	2	1	3	2	4	2	4	√	√
Airport	<	AirLandingArea			2	1	2	1	2	1	2	1	1		
Continent	<	Location			1	1	1	1	1	1	1	1	1		
Place	>	GeographicArea			1	1	1	1	1	1	1	1	1		
State	<	GeographicArea			3	1	3	1	3	1	3	1	1		
Church	<	Infrastructure					2	1	2	1	2	1	1		
Country	<	Location					2	1	6	1	6	1	1	√	
Place	>	Gulf					1	1	1	1	1	1	1		
Mountain	<	Location							1	1	1	1	1		
Place	>	Infrastructure							1	1	2	1	1	√	√
Hotel	<	Infrastructure							2	1	2	1	1		
Place	>	MedicalFacility							1	1	1	1	1		
Airport	<	Location							1	1	1	1	1		
Place	>	Airfield									2	1	1	√	
Place	>	Sea									1	1	1		

Table 11 Number of paths (♯Paths) and corresponding number of relations (♯Rels) found between each pair of concepts in the reference alignment with respect to the path length variable. The last three columns indicate how many base relations are contained in the relations found (♯BaseRels), if the paths are different (≠Path), and if the sequence of relations in different paths are different (≠SeqOfRel).

fect of local path length increase is higher than that of global path length.
– Ontology selection is the main parameter impacting time performance.

Algebras of relations allowed for finely characterising the added benefits of these parameter values from the standpoint of the correctness of returned correspondences and the influence of the type of correspondences on this correctness. The observations are as follows:

– As paths get longer, new correct correspondences are still found;
– As paths get longer, correct correspondences may become non precise by additional relations;
– As paths get longer, incorrect correspondences do not become more correct and imprecise correspondences do not become more precise.

In summary, these experiments show once again that context-based ontology matching increases the quality of obtained results through multiplying sources of information. Even if conjunction obtains the best results, it seems that finer strategies could still improve the quality of alignments.

We plan to further develop the implementation and investigate more configurations in more situations. Developing and testing alternative aggregation strategies will also be an outcome of this work.

We disregarded confidence measures returned by matchers. They could be considered at each step of the framework and combined with relations [10,2] for refining the obtained results. Similarly, logical reasoning may be integrated within context-based matching.

Acknowledgement

We thank Mathieu d'Aquin (and Open university) for making Watson available to our experiments, and helping us using it. We also thank Duy Hoa Ngo (LIRMM, Montpellier) for running the test with YAM++ for us.

hal-00975284, version 1 - 8 Apr 2014

References

1. Aleksovski, Z.: Using background knowledge in ontology matching. Ph.D. thesis, Vrije Universiteit Amsterdam (2008)
2. Atencia, M., Borgida, A., Euzenat, J., Ghidini, C., Serafini, L.: A formal semantics for weighted ontology mappings. In: Proc. 11th International semantic web conference (ISWC), Boston (MA US). pp. 17–33 (2012)
3. d'Aquin, M., Baldassarre, C., Gridinoc, L., Angeletou, S., Sabou, M., Motta, E.: Characterizing knowledge on the semantic web with Watson. In: EON. pp. 1–10 (2007)
4. d'Aquin, M., Motta, E.: Watson, more than a semantic web search engine. Semantic web journal 2(1), 55–63 (2011)
5. David, J., Euzenat, J.: Comparison between ontology distances (preliminary results). In: Proc. 7th conference on international semantic web conference (ISWC), Karlsruhe (DE). pp. 245–260 (2008)
6. David, J., Euzenat, J., Sváb-Zamazal, O.: Ontology similarity in the alignment space. In: Patel-Schneider, P.F., Pan, Y., Hitzler, P., Mika, P., Zhang, L., Pan, J.Z., Horrocks, I., Glimm, B. (eds.) International Semantic Web Conference (1). Lecture Notes in Computer Science, vol. 6496, pp. 129–144. Springer (2010)
7. David, J., Euzenat, J.: On fixing semantic alignment evaluation measures. In: Shvaiko, P., Euzenat, J., Giunchiglia, F., Stuckenschmidt, H. (eds.) Proc. 3rd ISWC workshop on ontology matching (OM), Karlsruhe (DE). pp. 25–36 (2008)
8. Ding, L., Pan, R., Finin, T., Joshi, A., Peng, Y., Kolari, P.: Finding and ranking knowledge on the semantic web. In: Proc. 4th International Semantic Web Conference (ISWC). pp. 156–170 (2005)
9. Ehrig, M., Euzenat, J.: Relaxed precision and recall for ontology matching. In: Proc. of the Workshop on Integrating Ontologies, K-CAP 2005. CEUR Workshop Proceedings, vol. 156. CEUR-WS.org (2005)
10. Euzenat, J.: Algebras of ontology alignment relations. In: Sheth, A., Staab, S., Dean, M., Paolucci, M., Maynard, D., Finin, T., Thirunarayan, K. (eds.) The Semantic Web - ISWC 2008. LNCS, vol. 5318, pp. 387–402. Springer (2008)
11. Euzenat, J., Shvaiko, P.: Ontology Matching. Springer (2007)
12. Euzenat, J., Isaac, A., Meilicke, C., Shvaiko, P., Stuckenschmidt, H., Šváb, O., Svátek, V., van Hage, W.R., Yatskevich, M.: Results of the ontology alignment evaluation initiative 2007. In: Proc. 2nd International Workshop on Ontology Matching (OM) at the International Semantic Web Conference (ISWC) and Asian Semantic Web Conference (ASWC). pp. 96–132. Busan (KR) (2007)
13. Gracia, J., Lopez, V., d'Aquin, M., Sabou, M., Motta, E., Mena, E.: Solving semantic ambiguity to improve semantic web based ontology matching. In: Proc. 2nd ISWC Ontology Matching workshop (OM 2007), Busan (KR). pp. 1–12 (2007)
14. Hu, W., Chen, J., Zhang, H., Qu, Y.: How matchable are four thousand ontologies on the semantic web. In: Proc. 8th Extended Semantic Web Conference (ESWC). pp. 290–304 (2011)
15. Jain, P., Hitzler, P., Sheth, A., Verma, K., Yeh, P.: Ontology alignment for linked open data. In: Proc. 9th International Semantic Web Conference (ISWC). Lecture Notes in Computer Science, vol. 6496, pp. 401–416. Shanghai (CN) (2010)
16. Jain, P., Yeh, P., Verma, K., Vasquez, R., Damova, M., Hitzler, P., Sheth, A.: Contextual ontology alignment of LOD with an upper ontology: A case study with Proton. In: Proc. 8th Extended Semantic Web Conference (ESWC). pp. 80–92 (2011)
17. Jiménez-Ruiz, E., Cuenca Grau, B.: LogMap: Logic-based and scalable ontology matching. In: Proc. 10th International Semantic Web Conference (ISWC). Lecture Notes in Computer Science, vol. 7031, pp. 273–288. Springer (2011)
18. Mascardi, V., Locoro, A., Rosso, P.: Automatic ontology matching via upper ontologies: A systematic evaluation. IEEE Trans. Knowl. Data Eng. 22(5), 609–623 (2010)
19. Ngo, D., Bellahsene, Z.: YAM++: A multi-strategy based approach for ontology matching task. In: Proc. 18th International Conference on Engineering and Knowledge Management (EKAW). Lecture Notes in Computer Science, vol. 7603, pp. 421–425. Springer (2012)
20. Quix, C., Roy, P., Kensche, D.: Automatic selection of background knowledge for ontology matching. In: Proc. International Workshop on Semantic Web Information Management (SWIM), Athens (GR). p. 5. ACM (2011)
21. Sabou, M., d'Aquin, M., Motta, E.: Exploring the semantic web as background knowledge for ontology matching. J. Data Semantics 11, 156–190 (2008)
22. Sabou, M., d'Aquin, M., Motta, E.: Scarlet: Semantic relation discovery by harvesting online ontologies. In: ESWC. pp. 854–858 (2008)
23. Stumme, G., Mädche, A.: FCA-Merge: Bottom-up merging of ontologies. In: Proc. 17th International Joint Conference on Artificial Intelligence (IJCAI). pp. 225–234. Seattle (WA US) (2001)
24. Zhang, S., Bodenreider, O.: Experience in aligning anatomical ontologies. International Journal on Semantic Web and Information Systems 3(2), 1–26 (2007)