Semantic Web 0 (0) 1
IOS Press


# Automatic evaluation of complex alignments: an instance-based approach

Elodie Thiéblin *, Ollivier Haemmerlé, Cássia Trojahn
*Institut de Recherche en Informatique de Toulouse, Toulouse, France*
*E-mails: elodie@thieblin.fr, ollivier.haemmerle@irit.fr, cassia.trojahn@irit.fr*

**Abstract.** Ontology matching is the task of generating a set of correspondences (i.e., an alignment) between the entities of different ontologies. While most efforts on alignment evaluation have been dedicated to the evaluation of simple alignments (i.e., those linking one single entity of a source ontology to one single entity of a target ontology), the emergence of complex approaches requires new strategies for addressing the problem of automatically evaluating complex alignments (i.e., those composed of correspondences involving logical constructors or transformation functions). This paper proposes a benchmark for complex alignment evaluation composed of an automatic evaluation system that relies on queries and instances, and a dataset about conference organisation. This dataset is composed of populated ontologies and a set of competency questions for alignment as SPARQL queries. State-of-the-art alignments are evaluated and a discussion on the difficulties of the evaluation task is provided.

Keywords: ontology matching, complex alignment, evaluation, benchmark

## 1. Introduction

Ontology matching is the task of generating a set of correspondences (i.e., an alignment) between the entities of different ontologies. This is the basis for a range of other tasks and applications, such as data integration, ontology evolution, and query rewriting. While the field has fully developed in the last decades, most works are still dedicated to the generation of simple correspondences (i.e., those linking one single entity of a source ontology to one single entity of a target ontology). However, simple correspondences are insufficient for covering the different kinds of heterogeneities (lexical, semantic, conceptual) in the ontologies to be matched. More expressiveness is achieved by complex correspondences, which can better express the relationships between entities of different ontologies. For example, the piece of knowledge that a conference paper has been accepted can be represented as a class IRI *ekaw:Accepted_Paper* in a source ontology, or as a class expression representing the papers (the range of *cmt:hasDecision* is *cmt:Paper*) having a decision of type *cmt:Acceptance* in a target ontology. The correspondence $\langle$*ekaw:Accepted_Paper,* $\exists$*cmt:hasDecision.cmt:Acceptance,* $\equiv$*, 1*$\rangle$ expresses an equivalence between the two representations of "accepted paper", with a confidence value of 1.

Earlier works in the field have introduced the need for complex ontology alignments [1, 2], and different approaches for generating them have been proposed in the literature afterwards. These approaches rely on diverse methods, such as correspondence patterns [3–5], knowledge-rules [6], statistical methods [7–9], competency questions for alignment [10], genetic programming [11] or still path-finding algorithms [12]. In others fields, such as relational databases, different approaches have been proposed so far [13, 14], however, covering less expressive knowledge representation languages and models. The reader can refer to [15] for a survey on complex matching. While works on complex ontology matching have been mostly dedicated to the development of approaches able to generate complex alignments, there is still a lack of benchmarks[1]

*Corresponding author. E-mail: elodie@thieblin.fr.

[1]Following the definition of "benchmark" as a standard by which something can be measured or judged (from the Ameri-

on which the approaches can be systematically evaluated. On the one hand, most existing matching proposals have been manually evaluated [3], usually in terms of precision, or on approach-tailored datasets [9] on which recall is calculated. On the other hand, most efforts on systematic evaluation are still dedicated to matching approaches dealing with simple alignments. Although a large spectrum of matching cases has been proposed so far in the Ontology Alignment Evaluation Initiative campaigns (OAEI)[2], e.g., involving synthetically generated or real world datasets with large and domain-specific ontologies, these datasets are mostly limited to simple alignments. Recently, the first OAEI complex track was proposed [16] opening new perspectives for the automatic evaluation in the field.

In this paper, a benchmark for evaluating complex alignments is proposed. This benchmark is composed of a dataset involving ontologies, populated with controlled and shared instances, reference competency question queries, and an automatic evaluation system. "Controlled" or "regularly" populated instances mean that every entity (class or property) concerned by the alignment (as for the CQAs) should have at least one instance in both ontologies. While classical benchmarks in the field [17, 18] rely on reference alignments and measurements of compliance between the generated and reference alignments (usually using classical precision and recall as evaluation metrics), here we propose a set of competency questions as reference. A competency question expresses, through a SPARQL query, the knowledge an alignment should cover between the source and target ontologies [19]. In particular, we propose two evaluation measures. While the *CQA coverage* measure relies on pairs of equivalent SPARQL queries (source and target queries) and measures how well an evaluated alignment covers these queries, the *intrinsic precision* compares the instances of the correspondences members. Intrinsic precision balances the CQA coverage like precision balances recall in information retrieval.

The contribution of this paper is manifold:

- we discuss the challenges of automatic evaluation of complex alignments with respect to classical evaluation workflows in the literature;
- we propose an automatic approach for evaluating complex alignments, which is based on competency questions for alignment in the form of SPARQL queries as references, and comparison of instances;
- we propose a dataset with controlled instance population and competency questions for alignment on which the alignments are evaluated;
- we evaluate state-of-the-art complex alignments on the proposed dataset and discuss their main strengths and weaknesses.

The automatic evaluation system and the populated datasets (and the scripts to generate them) are published under LGPL license[3].

The rest of this paper is organised as follows. The background on complex ontology matching and competency question for alignment are introduced in Section 2. Related works are discussed in Section 3. An evaluation workflow is proposed to analyse existing evaluation strategies (Section 4). This workflow is then used as basis for the evaluation system we propose here (Section 5). Next, the methodology followed to create the dataset and the dataset itself are detailed in Section 6. Evaluation of existing complex alignments over the benchmark is discussed in Section 7. Finally, conclusions and future work are presented in Section 8.

## 2. Background

Before introducing the notions of complex alignment and competency questions, the ontologies and their instances that will be used in the rest of this paper are introduced. The ontologies *cmt* and *ekaw* come from the Conference dataset [18]. Their fragments are depicted in Figures 1 and 2 using the format proposed in [20].

### 2.1. Complex ontology alignment

Ontology matching (as in [21]) is defined as the process of generating an alignment $A$ between two ontologies: a source ontology $o$ and a target ontology $o'$. $A$ is directional, denoted $A_{o\to o'}$. $A_{o\to o'}$ is a set of correspondences $\langle e, e', r, n\rangle$. Each correspondence expresses a relation $r$ (e.g., equivalence ($\equiv$), subsumption ($\sqsupseteq$, $\sqsubseteq$)) between two members $e$ and $e'$, and $n$ expresses the level of confidence $[0..1]$ in this corre-

can Heritage® Dictionary of the English Language, Fifth Edition. S.v. "benchmark." Retrieved January 7 2019 from https://www.thefreedictionary.com/benchmark), an alignment **benchmark** is considered composed of a dataset and an evaluation system.

[2]http://oaei.ontologymatching.org/

[3]https://framagit.org/IRIT_UT2J/conference-dataset-population

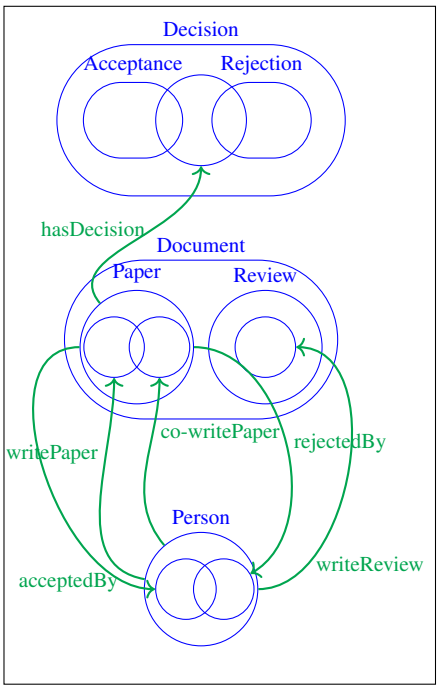


Fig. 1. Fragment of the *cmt* ontology used in the running examples.

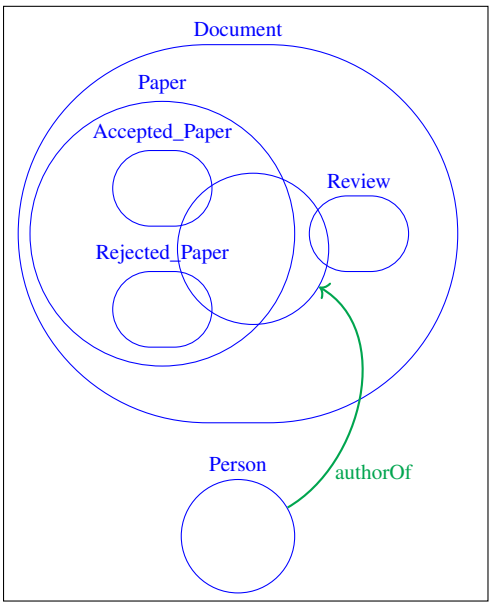


Fig. 2. Fragment of the *ekaw* ontology used in the running examples.

spondence. A member can be a single ontology entity (class, object property, data property, individual) of respectively $o$ and $o'$ or a more complex construction which is composed of some entities using constructors or transformation functions (as in the examples in the following). From that, two types of correspondences are considered depending on the type of their members [22]:

- a correspondence is **simple** if both $e$ and $e'$ are single entities (represented as IRIs): $\langle$*ekaw:Paper, cmt:Paper,* $\equiv$*, 1*$\rangle$
- a correspondence is **complex** if at least one of $e$ or $e'$ involves a constructor or a transformation function, respectively: $\langle$*ekaw:Accepted_Paper,* $\exists$ *cmt:hasDecision.cmt:Acceptance,* $\equiv$*, 1*$\rangle$ and $\langle$*concatenation(edas:hasFirstName, " ", edas:hasLastName), cmt:name,* $\rightarrow$*, 1*$\rangle$

A simple correspondence is usually noted (s:s), and a complex correspondence can be (s:c) if its source member is a single entity, (c:s) if its target member is a single entity or (c:c) if both members are complex entities. An approach which generates a complex alignment will be referred as "complex matching approach", "complex matching system" or "complex matcher" in the rest of this paper.

### 2.2. *Competency questions for alignment (CQAs)*

In ontology authoring, in order to formalise the knowledge needs of an ontology, competency questions (CQs) have been introduced as *ontology's requirements in the form of questions the ontology must be able to answer* [23]. As defined in [10, 19], a competency question for alignment (CQA) is a competency question which should (in the best case) be covered by two or more ontologies, i.e., it expresses the knowledge that an alignment should cover in the best case (if both ontologies' scopes can answer the CQA). The first difference between CQA and CQ is that the scope of the CQA is limited by the intersection of its source and target ontologies' scopes. The second difference is that this maximal and ideal alignment's scope is not known *a priori* (as it is the purpose of the alignment). As the ontology authoring competency questions (CQs) [24], a CQA can be expressed in natural language or as SPARQL SELECT queries.

Inspired from the predicate arity in [24], the notion of **question arity**, which represents the arity of the expected answers to a CQA was introduced in [10]:

- A *unary* question expects a set of instances or values, e.g., "Which are the accepted papers?" *(paper1), (paper2).*
- A *binary* question expects a set of instances or value pairs, e.g., "What is the decision on a paper?" *(paper1, accept), (paper2, reject).*

– A *n-ary* question expects a tuple of size 3 or more, e.g., "What is the decision associated with the review of a given paper?" *(paper1, review1, weak accept), (paper1, review2, reject)*.

## 3. Related work

Evaluation of matching systems is carried out over an **evaluation dataset**, usually composed of a set of ontologies, a reference alignment, and potentially different inputs (e.g., queries, instances, partial alignment). The generated alignment is then evaluated by an **evaluation system** which gives a score to the alignment produced by the system. Different evaluation dimensions can be considered in the process (that applies for both simple and complex evaluation):

**Tool-oriented** This dimension refers to the evaluation of the system performance in terms of run-time and memory usage. It is often performed over ontologies of different sizes and levels of expressiveness. Most OAEI tracks adopt this kind of evaluation.

**Controlled input** Evaluation of the generated alignment given different (and controlled) inputs. Such an evaluation was proposed for the GeoLink and Hydrography datasets of the OAEI Complex track [16]. Given a list of entities, the system should be able to find the correct (complex) construction involving these entities.

**Output-oriented** Evaluation of the output alignment itself over a dataset. This evaluation can be intrinsic or extrinsic. With the former, the quality of an alignment can be measured based on its intrinsic characteristics, as in [25] who evaluates the quality of an alignment over its logical coherence or in [26] where a good alignment should not violate the conservativity principle. With the latter, the evaluation is usually based on the compliance of the generated alignment with respect to a reference one (i.e., applying precision and recall metrics).

**Task-oriented** The quality of an alignment can also be assessed regarding its suitability for a specific task or application. Considering that ontology alignments are, in practice, constructed for a given application or with a given task in mind, it would be useful to set up experiments that do not stop at the delivery of the alignment but carry on to the application or task for which the alignment was constructed.

In the following, the main related works considering these evaluation dimensions are discussed.

### 3.1. Complex alignment evaluation metrics

Most works on alignment evaluation address the evaluation of simple alignments using a reference alignment or a sample of it. This is what has been done in the context of the OAEI campaigns. With respect to the evaluation of complex alignments, they have been evaluated manually, usually in terms of precision [3, 4, 8, 9], or on specific datasets in order to compute recall. In particular, the approach adopted in [8, 9] estimated their recall based on a recurring pattern (*Class by attribute-value*) between DBpedia and Geonames. They estimated the number of occurrences of this pattern between these ontologies and calculated the recall based on this estimation. In [12] a set of reference correspondences between two ontologies was manually created, involving few reference correspondences from which only two could not be expressed with simple correspondences. In [9] the authors proposed an algorithm to create an evaluation dataset that is composed of a synthetic ontology containing 50 classes with *Class-by-attribute-value* correspondences with DBpedia and 50 classes with no known correspondences with DBpedia. Both ontologies are populated with the same instances. In [27], inspired from [14], the approach for discovering complex attribute correspondences (i.e., {First Name, Last Name} = {Author}) between web interfaces is evaluated using *target accuracy* (that includes target precision and target recall) as metric. It evaluates how similar the generated alignment is with respect to a set of manually collected ones, using the notion of synonym attribute sets.

As discussed in [10] (inspired from [28]), alternative metrics of *accuracy* and *top−x accuracy* have been also applied in evaluation settings in which the number of correspondences is predefined, e.g., there is one correspondence for each entity of the target schema/ontology. The accuracy is calculated as the percentage of predefined questions having a correct answer. A "question" in this context could be a source entity to be matched and the "answers" the correspondences having this entity as source member. Some approaches output various answers for each question, e.g., a ranked list of correspondences for each source entity. In this case the top-$x$ accuracy is the percentage of questions whose correct answer is in the top-$x$ answers to the question. For example, top-3 accuracy

is the fraction of source entities for which the correct correspondence is in the three best correspondences generated by the system. Alternatively, the approach in [29], to evaluate complex correspondences between agronomic ontologies is based on manually comparing the results of the reference queries and queries automatically rewritten with the help of the complex alignments.

### 3.2. Complex alignment benchmarks

As discussed above, complex matchers are usually evaluated on custom evaluation alignment sets, usually covering the specificities of the approach to be evaluated. Recently, the first complex benchmark has been introduced in the OAEI campaigns [16]. The track consists of four datasets from different domains and considering different evaluation strategies:

**Complex conference** a consensual complex alignment was created using the query rewriting methodology from [22]. Each generated correspondence is manually classified as true positive or false positive, with respect to a reference alignment. The evaluated and reference correspondences are (s:c). In 2019, the benchmark presented in this paper has been used to automatically evaluate complex alignments.

**Hydrography and GeoLink** a set of ontologies on the hydrography domain and a pair of ontologies from GeoScience (more details bout the GeoLink dataset are provided in [30]). The matchers are evaluated following three subtasks: i) finding all entities which appear in a given correspondence, ii) finding the right construction involving those entities, and iii) finding the complex correspondences from scratch. Only the first subtask was implemented in the OAEI 2018 campaign [31], and the evaluation was automatically carried out using classical precision and recall (all alignments were simple equivalences). In 2019, a close metric to relaxed precision and recall [32] has been applied to entity identification and relationship identification tasks.

**Taxon** a set of CQAs over agronomic knowledge bases is rewritten with the evaluated alignments. Each rewritten query is manually classified as semantically equivalent to the source query or not. A "Query Well Rewritten" metric measures the percentage of CQA which had at least a semantically equivalent query after the rewriting process. Each correspondence of the evaluated alignment is also manually classified as true positive or false positive without a reference.

In 2018, only two systems, AMLC [5] and CANARD [33], were able to generate complex correspondences for those datasets. In 2019, a new system has been proposed, AROA[4]

### 3.3. Task-oriented benchmarks

Regarding task-oriented evaluation, [21] argued that different task profiles can be established to explicitly compare matching systems for certain tasks, such as ontology evolution or query answering, that have different constraints in terms of coverage and runtime. One such task-oriented evaluation approach was introduced in the OAEI in 2015 at the *OA4QA* track[5] [34], which focused on the task of query answering. This track used a synthetically populated version of the *Conference* dataset and a set of manually constructed queries over these $\mathcal{A}boxes$. A given query, such as *Q(x):=Author(x)* expressed using the vocabulary of the *cmt* ontology, was executed over the merged ontology $cmt \cup ekaw \cup A$, where $A$ is an alignment between *cmt* and *ekaw*. Precision and recall were calculated with respect to model answer sets, i.e., for each ontology pair and query *Q(x)*, and for each alignment $A$ computed by each matching system. An alternative approach for evaluating query answering without using instances was proposed by [35], where queries are compared without instance data, by grounding the evaluation on query containment.

In [36], an "end-to-end" evaluation in which a set of queries are rewritten using an evaluated alignment is proposed. The results of the queries are manually classified by relevance for a user on a 6-point scale. This evaluation was performed with two rewriting systems. If a source member $e$ does not appear in any correspondence of the alignment, the *upwards* rewriting system will use super-classes of $e$ which appear as source member in the alignment's correspondences and the *downwards* system will use subclasses of $e$. Three alignments were evaluated. For each alignment, 20 concepts were randomly selected to be queried and evaluated.

[4] http://oaei.ontologymatching.org/2019/results/complex/index.html

[5] http://www.cs.ox.ac.uk/isg/projects/Optique/oaei/oa4qa/index.html

While the task-based evaluation is pertinent for both simple and complex alignments, some tasks tend to have higher expressiveness requirements, such as query rewriting and ontology merging, as discussed in [22]. Complex alignments for query rewriting have been the focus of the work of [37][6], applied to a few pairs of ontologies. More recently, complex correspondences have been exploited for the task of query rewriting for federating agronomic taxonomy knowledge on the LOD [29] cloud. This dataset is the one used in the OAEI *Complex* track, on the ability to rewrite SPARQL queries using these alignments. The queries written for the source ontology were rewritten automatically using (s:s) or (s:c) correspondences and the system described by [38], and manually for (c:c) correspondences.

In fact, the query rewriting task can be seen as one of the main applications for complex alignments, and evaluation approaches based on this task are highly relevant. In the case of simple alignments, a naive approach for rewriting SPARQL queries can be to simply replace the IRI of an entity of the initial query by the IRI of the corresponding entity in the alignment, as described in [39]. For complex alignments, such a naive approach is not enough, as the semantics of the alignment itself has to be taken under consideration. [40] proposed an approach for writing specific SPARQL CONSTRUCT queries, but most query rewriting systems still rely on simple or (s:c) complex correspondence and fail in covering highly expressive (c:c) correspondences.

### *3.4. Positioning*

Contrary to works focusing on manually evaluating alignments, in terms of precision as in [3, 4], calculating recall on recurring patterns as in [8, 9], or relying on a sample of reference correspondences [12], we proposed here an evaluation benchmark that considers queries as references and relies on metrics based on query coverage (as for recall) and intrinsic precision (as for precision without a reference alignment). These metrics are detailed in Section 5. This is an automation of the evaluation process carried manually in [29]. Our approach requires, however, datasets populated in a controlled manner, differently from the datasets in [30].

As [34], we have queries as references instead of reference alignments. Close to ours, the evaluation in [34] relies on a synthetically populated version of the *Conference* dataset. However, their queries are executed over a merged ontology and limited to simple alignments. Here, the queries are executed over different populated ontologies. As [36], here a set of queries are rewritten using an evaluated alignment. However, their evaluation process relies on manually classifying the query results. While the approaches from [34, 36] are limited to simple alignments, query rewriting with complex alignments mostly address (s:c) correspondences [37, 38, 41].

Table 1 summarizes the existing alignment evaluation benchmarks that are close to our proposal (**CQA** benchmark, marked in bold in Table 1). Automation for (c:c) correspondences is still an open issue in the field. The proposal here is to automatise the evaluation process by shifting the problem to the comparison of instances, as detailed in the following sections.

## 4. Alignment evaluation workflow

As discussed above ontology alignment evaluation is often performed by comparing a generated alignment to a reference one. Most of the OAEI tracks use this kind of evaluation. However, the reference can also take other forms such as merged ontologies with their transitive closure, or equivalent queries (*i.e.*, a query over the source ontology and its equivalent for the target ontology). Even though these types of evaluation are developed and automated for simple alignments, automated evaluation of complex alignments is still addressed to a lesser extent [16]. The purpose of this section is to identify the difficulties inherent to complex alignment evaluation and discuss how they can be overcome. For that, we start by dissecting the alignment evaluation process into a generic workflow in Section 4.1. We then present the specificities of simple (Section 4.2) and complex (Section 4.3) alignment evaluation, together with a detailed example.

### *4.1. Generic workflow*

We analyse here the alignment evaluation process with a reference, regardless of its type. Figure 3 presents the generic workflow resulting from this analysis. This workflow applies for both simple and complex alignment evaluation. Overall, the steps followed in the evaluation process are:

[6] http://www.music.tuc.gr/projects/sw/sparql-rw/

Table 1
Comparison of ontology alignment evaluation benchmarks. The *Type of corresp.* column represents the form of the most expressive correspondences dealt with by the benchmarks – (c:c) is more complex than (s:c), which is more complex than (s:s).

| *Benchmark* | *Type of evaluation* | *Type of reference* | *Type of corresp.* |
|---|---|---|---|
| OA4QA [34] | Automatic (precision/recall) | Query | (s:s) |
| Query rewrite [36] | Manual | Query | (s:s) |
| Patterns evaluation [9] | Manual | Alignment | (s:c) |
| Patterns evaluation [8] | Manual | Alignment | (s:c) |
| Thieblin 2018 [22] | Manual | Alignment | (s:c) |
| GeoLink 2018 [30] | Automatic (precision/recall) /Manual | Alignment | (c:c) |
| Hydrography 2018 [30] | Automatic (precision/recall)/Manual | Alignment | (c:c) |
| GeoLink 2019 | Automatic (relaxed precision/recall) | Alignment | (c:c) |
| Hydrography 2019 | Automatic (relaxed precision/recall) | Alignment | (c:c) |
| Taxon [29] | Manual | Query | (c:c) |
| **CQA benchmark** | **Automatic (CQA coverage/intrinsic precision)** | **Query** | **(c:c)** |

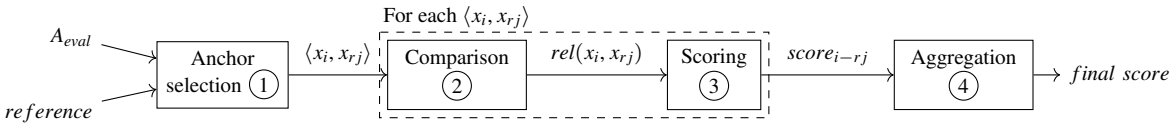


Fig. 3. Evaluation process of the alignment $A_{eval}$ with a generic $reference$.

**1 Anchor selection** The anchor selection step consists of outputting a pair of comparable objects $\langle x_i, x_{rj} \rangle$. $x_i$ is an object related to the evaluated alignment $A_{eval}$ and $x_{rj}$ is an object related to the reference $reference$. The objects depend on the type of reference. For example, if the reference is an alignment, $x_i$ is a correspondence ($c_i$) from $A_{eval}$, $x_{rj}$ is a correspondence ($c_{rj}$) from the reference alignment. If the reference is equivalent queries, $x_i$ can be a query derived from $A_{eval}$ and $x_{rj}$ a reference query.

**2 Comparison** The purpose of the comparison step is to output a relation $rel(x_i, x_{rj})$ for each pair previously obtained $\langle x_i, x_{rj} \rangle$. The relation can be an equivalence (*i.e.*, $x_i \equiv x_{rj}$), a subsumption, an overlap, a disjoint, *etc.* (this list can be extended according to the type of comparison performed). A similarity value can be associated with the relation. The comparison can be syntactic, semantic or instance-based as developed in Sections 4.2 and 4.3. For correspondence comparison (if the reference is an alignment), $x_i = c_i = \langle e_i, e'_i, r_i, n_i \rangle$ and $x_{rj} = c_{rj} = \langle e_{rj}, e'_{rj}, r_{rj}, n_{rj} \rangle$. Each element of an evaluated correspondence should be compared to its counterpart in the reference correspondence. $rel(c_i, c_{rj})$ can be decomposed into the relations between the elements of $c_i$ and $c_{rj}$: source members ($e_i$, $e_{rj}$), target members ($e'_i$, $e'_{rj}$), relations ($r_i$, $r_{rj}$) and confidence values ($n_i$, $n_{rj}$). A similarity score can be added to each relation between components.

$$rel(x_i, x_{rj}) = \begin{cases} rel(e_i, e_{rj}) \\ rel(e'_i, e'_{rj}) \\ rel(r_i, r_{rj}) \\ rel(n_i, n_{rj}) \end{cases} \quad (1)$$

**3 Scoring** The scoring step associates a score with each relation found in the previous step. Thus, the scoring functions are directly impacted by the relation $rel(x_i, x_{rj})$ found between the objects. Different scoring metrics have been proposed in the literature. The purpose of this section is not to be exhaustive but rather to give insights on how the comparison step impacts the correspondence pair scores. The classical score, used in the classical precision and recall metrics is:

$$classical\ score = \begin{cases} 1 & \text{if } x_i = x_{rj} \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

It may happen however that an alignment is very close to the expected result and another one is quite remote from it, although both share the same precision and recall. The reason for this is that standard metrics only compare two sets of objects (strict syntactic comparison) without considering if these are close or remote to each other. It may be helpful for users to know whether the found objects are close to the expected one and easily repairable or not. It is thus necessary to measure the proximity between objects instead of their strict equality. In order to better discriminate such systems a relaxed precision and recall measures were defined which replace the set intersection by a distance [32].

$$relaxed\ prec\ score = \begin{cases} 1 & \text{if } x_i \leqslant x_{rj} \\ 0.5 & \text{if } x_i > x_{rj} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

$$relaxed\ rec\ score = \begin{cases} 1 & \text{if } x_i \geqslant x_{rj} \\ 0.5 & \text{if } x_i < x_{rj} \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

The score can also be that which was associated with the relation found in the previous step. For example, if the comparison was syntactical and based on an edit distance, the edit distance value associated with $rel(x_i, x_{rj})$ can directly be used as $score_{i-rj}$. When using an instance-based comparison, a percentage of relevant instances can be associated with $rel(x_i, x_{rj})$ as in [36]. We call the scores which use the similarity value obtained during the comparison phase, the **comparison value** scoring functions. When dealing with correspondences, their confidence value can also be incorporated into the score, as in the weighted precision and recall metrics.

The variety of scoring functions and all their possible combinations point out that there is not one consensual way to measure the compliance of an alignment with regard to a reference. There is no "best scoring function" or "best metric". It depends on what the evaluation is supposed to measure. For example, if the evaluation measures how well an alignment allows for retrieving all results for a given query, regardless of the precision, a *recall-oriented* score can be applied [32]. If the purpose of the evaluation is to measure the exactitude of an alignment, then a classical function (1 if correct, 0 if incorrect) can be applied.

④ **Aggregation** The scores are locally and globally aggregated to give the *final score*. The aggregations can be performed with different functions: best match, average, weighted average, *etc.* The local aggregation aggregates all scores for a given object. There can be different local aggregations. For example, there can be an aggregation over the evaluated object and one over the reference object. The global aggregation aggregates all the locally-aggregated scores. For example, if the local aggregation was performed over the reference object, all the reference objects were given a score. The reference object scores can be aggregated into a final score. A final score locally aggregated over the evaluated objects is often referred to as the *precision* score. A final score locally aggregated over the reference objects is often referred to as the *recall* score.

The differences between simple and complex alignment evaluation lie in the Anchor selection ① and Comparison ② steps. We detail how they are performed for simple alignments in Section 4.2 and what are the challenges for their application to complex alignments in Section 4.3.

### *4.2. Workflow for simple alignment evaluation*

*Anchor selection* As introduced above, the anchor selection step consists of outputting a pair of comparable objects. In a simple alignment, each correspondence consists of a pair of URIs linked by a relation and potentially a confidence. The anchor selection can be performed by outputting all pairs of correspondences whose source member or target member are equivalent. As the source and target members of simple correspondences are URIs, an exact string match between the URIs is sufficient. Consider the evaluated correspondences:

- $c_1$=⟨ *o:Paper* , *o′:Paper* , ≡ ⟩
- $c_2$=⟨ *o:Paper* , *o′:Document* , ≡ ⟩

and the reference correspondence:

- $c_{r1}$=⟨ *o:Paper* , *o′:Paper* , ≡ ⟩

The pairs $\langle c_1, c_{r1}\rangle$ and $\langle c_2, c_{r1}\rangle$ are formed by comparing their source member to that of the reference correspondence. In the case of reference queries, the anchoring phase consists of translating a source query into a target query, using the evaluated alignment. That means that the evaluated alignment is used for generating a query in terms of the target ontology translating a query in terms of the source ontology. The output pair consists of the generated query and the reference target one. For simple alignments, the query rewriting system can consist in replacing each URI from the source query by an equivalent found in the evaluated alignment. For instance, the reference source query $q_{rs}$ `SELECT ?x WHERE{?x a o1:Paper.}` gives:

- $q_1$ with $c_1$
  `SELECT ?x WHERE{?x a o2:Paper.}`
- $q_2$ with $c_2$
  `SELECT ?x WHERE{?x a o2:Document.}`

The reference query in this scenario is $q_{rt}$: `SELECT ?x WHERE{?x a o2:Paper.}`. The pairs $\langle q_1, q_{rt}\rangle$ and $\langle q_2, q_{rt}\rangle$ are formed.

*Comparison* The purpose of the comparison step is to output a relation $rel(x_i, x_{rj})$ for each pair previously obtained $\langle x_i, x_{rj}\rangle$. In general, the comparison of the objects can be performed in a syntactic, semantic or instance-based manner.

A **syntactic** comparison compares the string representations of the objects. When dealing with simple alignments, the correspondences member URIs are compared. As the URIs are strings, a syntactical comparison is enough. This kind of comparison is the most common in the OAEI simple tracks. This kind of comparison is limited to stating whether objects (correspondences, queries, *etc.*) are equivalent, syntactically similar or different. In the example correspondence, the source, target members, and relations of $c_1$ and $c_{r1}$ are syntactically equivalent. $c_2$ is syntactically different from $c_{r1}$ because their target member differ. $q_1$ is syntactically equivalent to $q_{rt}$, $q_2$ is not.

A **semantic** comparison is based on reasoning rules. [32] propose to compute whether a simple evaluated correspondence is more specific or more general than the reference one based on taxonomic inference. $c_1$ is semantically equivalent to $c_{r1}$. $c_2$ is more general than $c_{r1}$ because *o′:Paper* is a subclass of *o′:Document*. In [35], a comparison between queries without instances can be performed based on inference rules. The semantic comparison does not depend on an ontology population. It can rely on existing reasoners and would work with every construction possible of the same axiom (inverse of inverse property, equivalent classes, *etc.*). $q_1$ is semantically equivalent to $q_{rt}$, $q_2$ is more general than $q_{rt}$.

An **instance-based** comparison is based on the interpretation of the objects in knowledge bases where the aligned ontologies have an associated $\mathcal{A}box$. The instance-based comparison only needs comparing sets of URIs. There is no expressiveness restriction for the evaluated alignment. The syntactic form of the correspondence does not matter. Therefore it can be used in the same manner for simple or complex correspondences. However, it fully relies on the ontologies' $\mathcal{A}box$. If the $\mathcal{A}box$ contains errors, or is irregular, the comparison results can be erroneous. For instance, if the target ontology $o'$ is only populated with *o′:Paper* instances (if there are no *o′:Document* instances which are not *o′:Paper*), then $c_2$ (resp. $q_2$) could be found equivalent to $c_{r1}$ (resp. $q_{rt}$). Complementary, another example of irregular population is the case where the *o′:acceptedBy* property is only instantiated for *o′:paper1* whereas every accepted paper is supposed to be *o′:acceptedBy* someone.

### 4.3. Workflow for complex alignment evaluation

The first issue when dealing with complex alignment evaluation is the creation of the reference. In [22], the proposed reference alignments were limited to (s:s) and (s:c) correspondences. This way, a certain completude could be ensured: an evaluated correspondence which would fall into the reference alignment creation criteria could be classified as correct or incorrect. When dealing with no restriction on the shapes of the correspondences, it becomes hard to prove that a reference alignment covers every possible correct correspondence.

*Anchor selection* As introduced before, the anchor selection step consists of outputting a pair of comparable objects. In comparison with simple alignments, complex correspondence members are not limited to URIs. They therefore require more than a simple syntactic match. When dealing with a reference alignment, the (s:c) or (c:s) correspondences can be anchored on their simple member. For example, the (s:c) evaluated correspondence $\langle$ *o:AcceptedPaper* , $\exists o'$*:acceptedBy*.$\top$ , $\equiv \rangle$ can be put in pair with the reference(s:c) correspondence $\langle$ *o:AcceptedPaper* , $\exists o'$*:hasDecision.o′:Acceptance* , $\equiv \rangle$ because their source members are the same URI (*o:AcceptedPaper*).

However, this kind of anchoring is not as easily applicable for (c:c) correspondences.

In the case of reference queries, the anchoring phase consists of translating a source query based on the evaluated alignment. A query rewriting system dealing with complex correspondences is thus needed. As introduced in Section 3, query rewriting systems only deal with (s:c) correspondences [37, 38, 41]: they translate a source URI into an equivalent construction based on the correspondence. Dealing with (c:s) and (c:c) correspondences for query rewriting remains a challenge, as further developed in Section 5.

*Comparison* The purpose of the comparison step is to output a relation for each pair of objects previously obtained. As for simple alignments, the comparison can be syntactic, semantic or instance-based.

A **syntactic** comparison for complex correspondences could measure how much effort should be done to transform an evaluated correspondence into the reference one. However correspondences which use different constructors, or different levels of factorisation can express the same meaning. A syntactic comparison also depends on the language in which the correspondences are expressed. Such a comparison strongly depends on the way the reference correspondences, queries, *etc.* are expressed.

For example, $\langle$ *o:Author* , $\exists o'$*:authorOf*.$\top$ , $\equiv \rangle$ is semantically equivalent to the correspondence $\langle$ *o:Author* , $\exists o'$*:writtenBy*$^-$.$\top$ , $\equiv \rangle$. However, these two correspondences use different URIs in their constructors and thus are syntactically different. The correspondences $\langle$ *o:AcceptedPaper* , $\exists o'$*:acceptedBy*.$\top$ , $\equiv \rangle$ and $\langle$ *o:AcceptedPaper* , $\geqslant 1\ o'$*:acceptedBy*.$\top$ , $\equiv \rangle$ are equivalent but expressed using different constructors (respectively an existential restriction or a cardinality restriction over the $o'$*:acceptedBy* property). They are also syntactically different. A factorisation problem would consist in verifying that $\langle$ *o:paperWrittenBy* , *dom*($o'$*:Paper*) $\sqcap$ $o'$*:writes*$^-$ , $\equiv \rangle$ and $\langle$ *o:paperWrittenBy* , ($o'$*:writes* $\sqcap$ *range*($o'$*:Paper*))$^-$ , $\equiv \rangle$ are equivalent correspondences. The *inverse* constructor is factorised in the second correspondence. A syntactic comparison of queries is faced with the same problems: syntactically different SPARQL queries can share the same semantics.

A **semantic** comparison would then be an alternative solution. However, the expressiveness of the evaluated alignment with a semantic comparison is limited to $\mathcal{SROIQ}$ (the decidable fragment of OWL [42]). Correspondences with transformation functions could not be compared with such a comparison. The semantic query comparison proposed by [35] is based on query containment which can be based on inferences. However, it is also limited with regard to queries with transformation functions.

As discussed in Section 4.2, **instance-based** comparison is applicable for both simple and complex alignments. However, it requires the knowledge bases to be regularly populated. Hence, for this kind of comparison, the desiderata for instance data is that the ontologies to be matched have ideally to be regularly and consistently populated with common instances.

In the following sections, examples of reference alignment-based evaluation and reference query-based evaluation are presented.

### 4.4. *Reference alignment based evaluation*

For sake of simplicity, the following examples are presented with (s:c) correspondences. Consider the following alignments generated by a system:

$c_{11}$=$\langle$ *ekaw:Accepted_Paper,*
$\exists$ *cmt:hasDecision.cmt:Acceptance,* $\equiv$, *1*$\rangle$
$c_{12}$=$\langle$ *ekaw:Accepted_Paper,*
$\exists$ *cmt:hasDecision.{icmt:decision1},* $\equiv$, *1*$\rangle$
$c_{13}$=$\langle$ *ekaw:Accepted_Paper,*
$\exists$ *cmt:acceptedBy.*$\top$, $\equiv$, *1*$\rangle$
$c_{14}$=$\langle$ *ekaw:Accepted_Paper, cmt:Paper,* $\equiv$, *1*$\rangle$
$c_{15}$=$\langle$ *ekaw:Accepted_Paper,*
$\exists$ *cmt:hasDecision.cmt:Acceptance,* $\sqsupseteq$, *1*$\rangle$
$c_{21}$=$\langle$ *ekaw:authorOf, cmt:writePaper,* $\equiv$, *1*$\rangle$
$c_{41}$=$\langle$ *ekaw:Paper,*
$\exists$ *cmt:hasDecision.cmt:Decision,* $\equiv$, *1*$\rangle$

Two populations of the *cmt* ontology are considered, as presented in Table 2.

The reference alignment is an alignment containing the following correspondences:

$c_{r1}$=$\langle$ *ekaw:Accepted_Paper,*
$\exists$ *cmt:hasDecision.cmt:Acceptance,* $\equiv$, *1*$\rangle$
$c_{r2}$=$\langle$ *ekaw:authorOf, cmt:writePaper* $\sqcup$
*cmt:co-writePaper* $\sqcup$ *cmt:writeReview,* $\sqsupseteq$, *1*$\rangle$
$c_{r3}$=$\langle$ *ekaw:Rejected_Paper,*
$\exists$ *cmt:hasDecision.cmt:Rejection,* $\equiv$, *1*$\rangle$

*Anchoring* In this example, the anchoring phase is performed on an exact match (string match) of the two correspondences source members. All the pairs $\langle c_{1k}, c_{r1}\rangle, k \in [1..5]$ are considered (e.g., $\langle c_{11}, c_{r1}\rangle$) together with the pair $\langle c_{21}, c_{r2}\rangle$.

Table 2
D1 and D2 datasets with different populations.

| | |
|---|---|
| D1 | icmt:paper1 cmt:hasDecision icmt:decision1 .<br>icmt:decision1 a cmt:Acceptance .<br>icmt:paper1 cmt:acceptedBy icmt:person1 .<br>icmt:paper2 cmt:hasDecision icmt:decision2 .<br>icmt:decision2 a cmt:Acceptance .<br>icmt:paper2 cmt:acceptedBy icmt:person1 .<br>icmt:paper3 cmt:hasDecision icmt:decision3 .<br>icmt:decision3 a cmt:Acceptance .<br>icmt:paper3 cmt:acceptedBy icmt:person1 .<br>icmt:paper4 cmt:hasDecision icmt:decision4 .<br>icmt:decision4 a cmt:Rejection .<br>icmt:paper4 cmt:rejectedBy icmt:person1 .<br>icmt:paper5 cmt:hasDecision icmt:decision5 .<br>icmt:decision5 a cmt:Rejection .<br>icmt:paper5 cmt:rejectedBy icmt:person1 .<br>icmt:paper6 cmt:hasDecision icmt:decision6 .<br>icmt:decision6 a cmt:Rejection .<br>icmt:paper6 cmt:rejectedBy icmt:person1 . |
| D2 | icmt:paper1 cmt:hasDecision icmt:decision1 .<br>icmt:decision1 a cmt:Acceptance .<br>icmt:paper1 cmt:acceptedBy icmt:person1 .<br>icmt:paper2 cmt:hasDecision icmt:decision2 .<br>icmt:decision2 a cmt:Acceptance .<br>icmt:paper3 cmt:hasDecision icmt:decision3 .<br>icmt:decision3 a cmt:Acceptance . |

The evaluated correspondence $c_{41}$ was not anchored to any of the reference correspondences as *ekaw:Paper* is never the source member of these correspondences. The reference correspondence $c_{r3}$ was not anchored to any of the evaluated correspondences, as none of them have *ekaw:Rejected_Paper* as source member.

In the following steps, for sake of simplicity, only the pairs $\langle c_{1k}, c_{r1}\rangle, k \in [1..5]$ will be further developed in the examples.

*Correspondence comparison.* The compared correspondences have the same source member (anchoring step) and the confidence of all of them is 1.0. Therefore, only the target members and relations of the correspondences need to be compared. The comparison techniques considered for the target member comparison in this example are the following:

**Syntactic** Basic string comparison of the target member expression. Possible relation output: $\equiv$, $\perp$

**Semantic** Inference-based comparison of the target member expressions. Possible relation output: $\equiv$, $\sqsubseteq$, $\sqsupseteq$, $\perp$

**Instance-based** Comparison of the instance described by the target members. Possible relation output: $\equiv$, $\sqsubseteq$, $\sqsupseteq$, $\perp$, overlap

For the correspondence relation comparison, we chose to prefer equivalence relation ($\equiv$) than subsumption relations ($\sqsubseteq$, $\sqsupseteq$). This is what has been done for all comparison techniques (syntactic, semantic and instance-based). In Table 3, the relations between the correspondences $c_{1k}, k \in [1..5]$ and $c_{r1}$ are shown. In this table, the relation between the correspondences based on the non-equivalent part of the correspondence elements is represented. The relation between the correspondences is expressed using $=$,$>$,$<$,$\neq$ instead of $\equiv$, $\sqsubseteq$, $\sqsupseteq$, $\perp$ because they also include comparison between non-axiomatic elements such as the confidence value and the correspondence relation.

$$rel(c_i, c_{rj}) = \begin{cases} rel(e'_i, e'_{rj}) & \text{if } r_i = r_{rj} \\ rel(r_i, r_{rj}) & \text{if } e'_i \equiv e'_{rj} \end{cases}$$

Table 3
Relations between the evaluated correspondences $c_i$ and the reference correspondence $c_{r1}$.

| | | $c_{11}$ | $c_{12}$ | $c_{13}$ | $c_{14}$ | $c_{15}$ |
|---|---|---|---|---|---|---|
| syntactic | | $c_{11} = c_{r1}$ | $c_{12} \neq c_{r1}$ | $c_{13} \neq c_{r1}$ | $c_{14} \neq c_{r1}$ | $c_{15} > c_{r1}$ |
| semantic | | $c_{11} = c_{r1}$ | $c_{12} < c_{r1}$ | $c_{13} \neq c_{r1}$ | $c_{14} > c_{r1}$ | $c_{15} > c_{r1}$ |
| instance | D1 | $c_{11} = c_{r1}$ | $c_{12} < c_{r1}$ | $c_{13} = c_{r1}$ | $c_{14} > c_{r1}$ | $c_{15} > c_{r1}$ |
| | D2 | $c_{11} = c_{r1}$ | $c_{12} < c_{r1}$ | $c_{13} < c_{r1}$ | $c_{14} = c_{r1}$ | $c_{15} > c_{r1}$ |

As for the syntactic comparison (that syntactically compared the target members of the correspondences), the semantic comparison found that $c_{13}$ is wrong. The latter is due to the fact that no axiom in *cmt* states that: $\exists$*cmt:hasDecision.cmt:Acceptance* $\equiv \exists$*cmt:acceptedBy.*$\top$ Therefore, a reasoner would not find that $c_{13}$ and $c_{r1}$ are equivalent in a semantic way. The instance-based comparison shows that D1 was suited for the comparison but D2 was not. Indeed, D2 only contains accepted paper instances, therefore, no discrimination can be done at instance level between a paper and an accepted paper. Moreover, the instances are not consistently populated: only one paper out of the 3 has a *cmt:acceptedBy* relation. The same information ("accepted paper") is not homogeneously represented in the instances.

*Scoring* The classical, relaxed precision-oriented (relaxed prec) and recall-oriented (relaxed rec) have been calculated for each pair of correspondences, according to the results of the comparison step. The resulting scores are presented in Table 4.

Table 4
Scores over the calculated relations from Table 3.

| | | $c_{11}$ | $c_{12}$ | $c_{13}$ | $c_{14}$ | $c_{15}$ |
|---|---|---|---|---|---|---|
| Classical | semantic | 1 | 0 | 0 | 0 | 0 |
| | syntactic | 1 | 0 | 0 | 0 | 0 |
| | instance D1 | 1 | 0 | 1 | 0 | 0 |
| | instance D2 | 1 | 0 | 0 | 1 | 0 |
| Relaxed prec | semantic | 1 | 1 | 0 | 0.5 | 0.5 |
| | syntactic | 1 | 0 | 0 | 0 | 0.5 |
| | instance D1 | 1 | 1 | 1 | 0.5 | 0.5 |
| | instance D2 | 1 | 1 | 1 | 1 | 0.5 |
| Relaxed rec | semantic | 1 | 0.5 | 0 | 1 | 1 |
| | syntactic | 1 | 0 | 0 | 0 | 1 |
| | instance D1 | 1 | 0.5 | 1 | 1 | 1 |
| | instance D2 | 1 | 0.5 | 0.5 | 1 | 1 |

*Aggregation* To show the aggregation process, the reference alignment is $A_{ref} = \{c_{r1}, c_{r2}, c_{r3}\}$, and the evaluated alignment is $A_{eval} = \{c_{11}, c_{12}, c_{21}, c_{41}\}$. In the anchoring phase, the pairs: $(c_{11}, c_{r1})$, $(c_{12}, c_{r1})$, and $(c_{21}, c_{r2})$ were output. $c_{r3}$ was not paired with any evaluated correspondence. $c_{41}$ was not paired with any reference correspondence.

The pair scores considered in this step are the ones listed in Table 5: $score(c_{11}, c_{r1}) = 1$, $score(c_{12}, c_{r1}) = 0.5$, $score(c_{21}, c_{r2}) = 0.2$. As no evaluated correspondence $c_i$ was paired with more than one reference $c_{rj}$, no evaluated correspondence aggregation needs to be performed. The reference correspondence $c_{r1}$ was paired with more than one evaluated correspondence. The local aggregation for the reference correspondence $c_{r1}$ can be 0.75 with the average or 1.0 with the best-match.

Table 5
Local aggregation for the evaluated and reference correspondences. The values chosen for the global aggregation are shown in bold. The average of the locally aggregated scores for the evaluated and reference is shown as respectively *average (eval aggreg)* and *average (ref aggreg)*.

| Correspondence | average | best-match |
|---|---|---|
| $c_{11}$ | **1** | 1 |
| $c_{12}$ | **0.5** | 0.5 |
| $c_{21}$ | **0.2** | 0.2 |
| $c_{41}$ | **0** | 0 |
| average(eval aggreg) | **0.43** | 0.43 |
| $c_{r1}$ | 0.75 | **1** |
| $c_{r2}$ | 0.2 | **0.2** |
| $c_{r3}$ | 0 | **0** |
| average(ref aggreg) | 0.32 | **0.40** |

The global aggregation aggregates the locally aggregated scores for all the correspondences. Assuming that the average function was chosen for the evaluated reference correspondence and the best-match function was chosen for the reference correspondence local aggregation (in bold in Table 5). Then, an average function is chosen to be applied for the global aggregation. Two scores are obtained:

– global evaluated correspondence score: **0.43**. The aggregation over the evaluated correspondences gives what is usually referred to as the *precision* of an alignment.

– global reference correspondence score: **0.4**. The aggregation over the reference correspondences gives what is usually referred to as the *recall* of an alignment.

These two scores are traditionally combined into their harmonic mean called F-measure.
$final\ score(A_{eval}) = 2 \times \frac{precision \times recall}{precision + recall} = \mathbf{0.41}$
This score translates the fact that the evaluated alignment is perfectible as it does not cover all expected correspondences (low recall) and contains wrong correspondences (low precision).

### 4.5. *Reference queries*

The evaluation process using reference queries differs for the anchoring and comparison steps. Instead of a reference alignment, a set of equivalent queries is provided, as in the example below:

$q^s_{r1}$ =
```
SELECT ?s WHERE {
  ?s a ekaw:Accepted_Paper}
```
$q^t_{r1}$ =
```
SELECT ?s WHERE {
  ?s cmt:hasDecision ?o.
  ?o a cmt:Acceptance.}
```

*Anchoring* The anchoring phase consists in rewriting the reference source query $q^s_{r1}$ into a target evaluated query based on the evaluated alignment $q^t_{1k}$. Because of that, the anchoring phase depends on the employed rewriting system. Let us assume that the system of [38] was chosen. This system only deals with (s:c) correspondences and does not consider the correspondence relation nor the correspondence confidence value in the process. Each $c_{1k}, k \in [1..5]$ correspondence from the running example can be used to rewrite $q^s_{r1}$.

$q^t_{11}$=
```
SELECT ?s WHERE {
  ?s cmt:hasDecision ?o.
  ?o a cmt:Acceptance.}
```
$q^t_{12}$=
```
SELECT ?s WHERE {
  ?s cmt:hasDecision
  icmt:decision1.}
```
$q^t_{13}$=
```
SELECT ?s WHERE {
  ?s cmt:acceptedBy ?o.}
```
$q^t_{14}$=
```
SELECT ?s WHERE {
  ?s a cmt:Paper.}
```
$q^t_{15}$=
```
SELECT ?s WHERE {
  ?s cmt:hasDecision ?o.
  ?o a cmt:Acceptance.}
```

The pairs of queries which are output are all the $\langle q^t_{1k}, q^t_{r1}\rangle, k \in [1..5]$. In this example, only one correspondence was necessary to rewrite $q^s_{r1}$. However, more than one correspondence can be necessary to rewrite a query. For example, the following query may need 3 correspondences (one per *ekaw* IRI) to be rewritten using the rewriting system [38].
```
SELECT ?s WHERE{
?s a ekaw:Accepted_Paper.
?s ekaw:hasReviewer ?o.
?o a ekaw:PC_Member.}
```

*Comparison* The comparison techniques considered for the query comparison in this example are manually performed:

**Syntactic** Basic string comparison of the queries. Possible relation output: $\equiv$, $\perp$

**Semantic** Inference-based comparison of the queries content. Possible relation output: $\equiv$, $\sqsubseteq$, $\sqsupseteq$, $\perp$

**Instance-based** Comparison of the query results. Possible relation output: $\equiv$, $\sqsubseteq$, $\sqsupseteq$, $\perp$, overlap

Table 6 presents the relation between the queries based on the comparison techniques. In comparison with Table 3, the query rewritten based on $c_{15}$ is equivalent to $q^t_{r1}$ because the relation of $c_{15}$ was not taken into account in the rewriting process.

*Scoring and aggregation* The scoring and aggregation steps are the same for a reference alignment or reference queries. The scores aggregated over the evaluated queries would show how precise the alignment is with regard to these queries. However, such a score would also depend on the rewriting process (e.g., if the rewriting process brings noise, the precision would be impacted). The scores aggregated over the reference queries would show how many reference queries the alignment can cover, i.e. its suitability with regard to a query rewriting application.

Automating the evaluation of complex alignments using such queries as references and instance-based comparison is discussed in the following.

## 5. Automatic evaluation of complex alignments

From the analysis of the evaluation workflow introduced above, we have identified that **anchor selection** and **comparison** are the most difficult steps to automate for complex alignment. Instance-based comparison (of correspondences/queries, etc.) is, so far, the easiest comparison method to automatize. How-

Table 6
Relations between the evaluated queries $q^t_{1k}$ and the reference correspondence $q^t_{r1}$.

| | | $q^t_{11}$ | $q^t_{12}$ | $q^t_{13}$ | $q^t_{14}$ | $q^t_{15}$ |
|---|---|---|---|---|---|---|
| syntactic | | $q^t_{11} = q^t_{r1}$ | $q^t_{12} \neq q^t_{r1}$ | $q^t_{13} \neq q^t_{r1}$ | $q^t_{14} \neq q^t_{r1}$ | $q^t_{15} = q^t_{r1}$ |
| semantic | | $q^t_{11} = q^t_{r1}$ | $q^t_{12} < q^t_{r1}$ | $q^t_{13} \neq q^t_{r1}$ | $q^t_{14} > q^t_{r1}$ | $q^t_{15} = q^t_{r1}$ |
| instance | D1 | $q^t_{11} = q^t_{r1}$ | $q^t_{12} < q^t_{r1}$ | $q^t_{13} = q^t_{r1}$ | $q^t_{14} > q^t_{r1}$ | $q^t_{15} = q^t_{r1}$ |
| | D2 | $q^t_{11} = q^t_{r1}$ | $q^t_{12} < q^t_{r1}$ | $q^t_{13} < q^t_{r1}$ | $q^t_{14} = q^t_{r1}$ | $q^t_{15} = q^t_{r1}$ |

ever, as stated before, this comparison must be done over controlled instances, and a complex alignment dataset fulfilling such requirements does not exist. Here, a benchmark to evaluate complex alignments is proposed, including i) an evaluation system implementing instance-based comparison and using equivalent queries as references and ii) a dataset with controlled instances. Using equivalent SPARQL CQA as reference would ensure that the two compared objects are equivalent because they model the same piece of knowledge.

With respect to i), we propose two evaluation measures. While the *CQA coverage* measure relies on pairs of equivalent SPARQL queries (source and target queries) and measures how well an evaluated alignment covers these queries, the *intrinsic precision* compares the instances of the correspondences members. Intrinsic precision balances the CQA coverage like precision balances recall in information retrieval. With respect to ii) a methodology based on CQAs, as introduced in [10], is proposed to synthetically populate ontologies. This methodology was applied to five ontologies of the well-known Conference dataset [18].

In the following, the CQA coverage metric is detailed (Section 5.1), followed by the description of the intrinsic metric (Section 5.2).

### 5.1. CQA coverage metric

With this evaluation strategy, the reference is a set of equivalent CQAs in the form of SPARQL queries. An evaluated alignment $A_{eval}$ will be used to rewrite each source CQA. The rewritten queries will then be compared to the reference target CQA. The comparison of the queries is instance-based and a value is associated with each query relation based on the common part of the evaluated query and target CQA instances. The scoring metric chosen is the one keeping the comparison relation value. A best-match aggregation is locally performed over the reference queries. The locally aggregated scores are then aggregated by an average. In the following, each step of the proposed evaluation process is described.

#### 5.1.1. Source CQA anchoring

As stated above, the reference in this kind of evaluation is a set of equivalent CQAs as SPARQL queries. Each source CQA $cqa_s$ has an equivalent target CQA $cqa_t$. In the anchoring step, each source $cqa_s$ is rewritten using the generated alignment $A_{eval}$. The rewriting phase outputs all the possible rewritten target queries from the rewriting systems as the set $Q_t$. For each query $q_t$ in $Q_t$, a pair $(q_t, cqa_t)$ is formed.

Two rewriting systems have been considered. None of these systems consider the correspondence relation or correspondence value. The first system is the one from [38]. Each triple of $cqa_s$ is rewritten using $A_{eval}$. When the predicate or object of the triple appears as the source member of a correspondence in $A_{eval}$, the target member of this correspondence is transformed into a SPARQL subgraph and put in the triple's place in the query. This system only deals with (s:c) correspondences. If a triple can be rewritten with different correspondences, all the possible combinations are added into $Q_t$. For example, consider the CQA:

```
SELECT ?s WHERE{
?s a ekaw:Accepted_Paper.}
```

which contains *ekaw:Accepted_Paper* which is the source member of the correspondences $c_{1k}, k \in [1..5]$.

The rewritten query using the $c_{11}$ correspondence is:

```
SELECT ?s WHERE{
?s cmt:hasDecision ?o.
?o a cmt:Acceptance.}
```

This rewriting system cannot however work the other way around. For example, the CQA

```
SELECT ?s WHERE{
?s cmt:hasDecision ?o.
?o a cmt:Acceptance.}
```

cannot be rewritten with $c_{11}$.

The second system is based on instances and has been developed in the context of this paper. The instances $I^{cqa}_s$ of $cqa_s$ are retrieved from the source ontology. For each correspondence $c$ of $A_{eval}$, the source member is transformed into a query and which retrieves the set of instances $I_s$ over the source ontology. If $I_s \equiv I^{cqa}_s$, then, the target member of $c$ is transformed into a query and added to $Q_t$. For example the CQA:

```
SELECT ?s WHERE{
?s a ekaw:Accepted_Paper.}
```

retrieves a set of accepted paper instances in the *ekaw* ontology. This set of instances is then compared to the set of instances described by the source member of each correspondence. In this case, *ekaw:Accepted_Paper* describes the same instances as the source member of all the $c_{1k}, k \in [1..5]$. Therefore, the target member of each correspondence can be transformed into a query. For $c_{11}$, the output query is

```
SELECT ?s WHERE{
?s cmt:hasDecision ?o.
?o a cmt:Acceptance.}
```

This rewriting system allows queries such as

```
SELECT ?s WHERE{
?s cmt:hasDecision ?o.
?o a cmt:Acceptance.}
```

to be rewritten too using the inverse of $c_{11}$ for example (the inverse of a correspondence is its equivalent except that the source member becomes the target member and vice-versa).

Out of the existing rewriting systems dealing with complex correspondences, the one described in [38] deals with the most types of constructions. So far, the proposed instance-based rewriting system is one of the few systems able to deal with (c:c) correspondences. However, it is a feature of the system that (c:c) cannot be combined together.

#### 5.1.2. Comparison

The instances $I_t^{cqa}$ of $cqa_t$ are retrieved over the target ontology. The instances $I_t$ of $q_t$ are retrieved over the target ontology. $I_t$ and $I_t^{cqa}$ are compared and the query precision ($QP$) and query recall ($QR$) are associated as value with the relation $rel(q_t, cqa_t)$ (subsumption, overlap, equivalence, etc.) between the two queries.

$$QP = \frac{|I_t \cap I_t^{cqa}|}{|I_t|} \qquad QR = \frac{|I_t \cap I_t^{cqa}|}{|I_t^{cqa}|}$$

$$rel(q_t, cqa_t) = \begin{cases} \equiv \text{ if } QR = 1 \text{ and } QP = 1 \\ \sqsubseteq \text{ if } QR \leqslant 1 \text{ and } QP = 1 \\ \sqsupseteq \text{ if } QR = 1 \text{ and } QP \leqslant 1 \\ \mathit{overlap} \text{ if } 0 < QR \leqslant 1 \text{ and } 0 < QP \leqslant 1 \\ \bot \text{ if } QR = 0 \text{ and } QP = 0 \end{cases}$$

#### 5.1.3. Scoring

The relation (associated with the query precision and query recall values) between $cqa_t$ and $q_t$ is transformed by an harmonic mean into a query F-measure score:

$$Fmeasure = 2 \times \frac{QR \times QP}{QR + QP}$$

The query F-measure (equally balancing precision and recall) was preferred over other metrics to be the scoring function as it is commonly used in alignment evaluation to aggregate the results of precision and recall. However, users may prefer one score than other, depending on alignment usage or manipulation. This was an implementation choice, as a matter of facilitating the comparison of the evaluated alignments.

#### 5.1.4. Aggregation

As the rewriting phase outputs all the possible queries regardless of the correspondence relation, a lot of noise can be introduced. Moreover, the same query can be output by both rewriting systems. Therefore, for each $cqa_t$, the query $q_t$ with the best query F-measure score is kept. The best-match aggregation prevents the final score to suffer from the noise introduced by the query rewriting systems. If a $cqa_s$ could not be rewritten by the alignment, its query precision, query recall and query F-measure scores are 0.0. The global aggregation method is the average function. The final output of the evaluation system is an average query precision, query recall and query F-measure score for the evaluated alignment.

### 5.2. Intrinsic precision

The CQA coverage evaluation locally aggregates the results over the CQA and not the rewritten queries because of the noise added by the rewriting systems. In return, an alignment with all the possible correspondences (correct and erroneous) between the source and target ontologies would obtain a good CQA coverage score. To counterbalance the CQA coverage score, we propose to measure the **intrinsic instance-based Precision** of an alignment.

For each correspondence $c_i$ in the evaluated alignment, the instances $I_s$ represented by the source member are compared to the instance $I_t$ represented by the target member. Each correspondence is then classified as an *equivalent*, *subsumed*, *overlapping*, or *disjoint*, given the relation between $I_s$ and $I_t$, or *empty* if $I_s = I_t = \emptyset$. Therefore, a correspondence can be *empty* if both its members are either unsatisfiable entities or non populated entities.

Different precision scores are given for each type of correspondence member relation: the *equivalent* precision measures the percentage of correspondences whose members are exactly populated with the same instances, the *subsumed* precision measures the percentage of correspondences whose members subsume one another, the same goes for *overlapping* and *not disjoint* which consider correct all correspondences except the *disjoint* ones.

## 6. CQA-based dataset

In this section, first the methodology followed to create the evaluation dataset (populated ontologies and associated CQAs) is presented (Section 6.1). Then, the OAEI Conference dataset (Section 6.2) is described, followed by the population of its ontologies from real-life data (Section 6.3). Finally, the set of evaluation CQAs extracted from the CQAs used for the dataset population is discussed (Section 6.4).

### 6.1. Dataset creation methodology

The purpose here is to create a dataset on which ontology matchers can be run and on which the evaluation described in the previous section can be performed. Therefore, the dataset must contain populated ontologies and a set of CQAs expressed as SPARQL queries over these ontologies.

The proposed methodology has the following main steps:

1. Create a set of CQAs based on an application scenario. Only unary and binary CQAs were considered in this work.
2. Create a pivot format (i.e., the bridge format used for representing in a uniform way the data extracted from the data sources) which covers all the CQAs from step 1.
3. For each ontology of the dataset, create SPARQL INSERT queries corresponding to the pivot format.
4. Instantiate the pivot format with real-life or synthetic data.
5. Populate the ontologies with the instantiated pivot format using the SPARQL INSERT queries.
6. Run a reasoner to verify the consistency of the populated ontologies. If an exception occurs, try to change the interpretation of the ontology and iterate over steps 3 to 5.
7. Based on SPARQL INSERT queries, translate the CQAs covered by two or more ontologies as SPARQL queries.

In this methodology, the interpretation of the ontologies is the same for ontology population and CQA creation. The creation of CQAs can be done by interviewing users and domain experts, as recommended in the NeOn methodology [43] for competency question authoring. The CQAs can also derive from the competency questions which were used to design the ontologies of the dataset. In this implementation, however, one expert created the CQAs. This set has been discussed with a second expert who judged the set exhaustive enough for covering the conference organisation scenario.

In [22], (c:c) correspondences were not included in the dataset hence no exhaustive coverage could be guaranteed. However, as CQAs represent basic pieces of knowledge, they can be exhaustively covered by an alignment regardless of the shape of the correspondences. Using the same list of CQAs for ontology population and evaluation also insures the consistency of the answers of the evaluation CQAs.

### 6.2. Conference dataset

The dataset used here is the Conference dataset[7] proposed in [44]. It has been widely used [18], especially in the OAEI campaigns where it is a reference evaluation track. It is composed of 16 ontologies on the conference organisation domain and simple reference alignments between 7 of these ontologies. These ontologies were developed individually. The motivation for the extension of this dataset is that the ontologies are real ontologies (as opposed to synthetic ones), they are expressive and largely used for evaluation in the field. The query-oriented evaluation benchmark OA4QA was also based on this dataset [34]. Furthermore, reference complex alignments for query rewriting and ontology merging tasks have been proposed over five ontologies of this dataset [22].

In the first OAEI complex track, an evaluation was proposed over a consensual complex alignment between three ontologies (*cmt*, *conference*, *ekaw*) [16]. Here, the five ontologies covered by [22] have been populated: *cmt*, *conference* (Sofsem), *confOf* (confTool), *edas* and *ekaw* (Table 7).

[7] http://oaei.ontologymatching.org/2018/conference/index.html
http://owl.vse.cz:8080/ontofarm/

Table 7
Number of entities by type of each ontology.

| | cmt | conference | confOf | edas | ekaw |
|---|---|---|---|---|---|
| Classes | 30 | 60 | 39 | 104 | 74 |
| Obj. prop. | 49 | 46 | 13 | 30 | 33 |
| Data prop. | 10 | 18 | 23 | 20 | 0 |

Even though this dataset has been largely used, it has only been partially populated. In the OA4QA track, only the classes covered by the 18 queries were populated and the creation of the synthetic $\mathcal{A}box$ has not been documented.

### 6.3. Populating the conference ontologies

In order to create the CQAs and re-interpret the Conference ontologies, the conference organisation scenario has been considered. First, the list of CQA has been established by examining a real-life use case: the Extended Semantic Web Conference 2018 edition. Second, the list of CQAs created from this use case has been extended by exploring the conference ontologies scope. The Extended Semantic Web Conference[8] (ESWC) is open review and its website provided a good base to analyse which information is needed for conference organisation. In order to create the artificial instances of the pivot format, the ESWC 2018 use case as well as data from Scholarly Data [45] were considered.

#### 6.3.1. Re-interpreting the ontologies with real-life data

As mentioned before, the first step of the process was to create a list of CQAs and re-interpret the ontologies under the perspective of a conference organisation application. By analysing the ESWC 2018 website, a first list of CQAs was created. The methodology was followed based on this first list of CQAs. The pivot format was instantiated with the website data.

While running the Hermit [46] reasoner in step 6 of the methodology, several exceptions were encountered. For most of them, the problem was with the interpretation of the ontology. For example, in the *cmt* ontology, *cmt:hasAuthor* is functional. Unlike primarily interpreted, this means that *cmt:hasAuthor* represents a "is first author of" relationship between a *cmt:Paper* and a *cmt:Author*. Then, the SPARQL INSERT queries have been modified in order to fit the new interpretation of the ontology.

Two exceptions have been detected, which could not be resolved by a change of interpretation. In that case, the original ontologies have been slightly modified:

- *cmt*: the relation *cmt:acceptPaper* between an *Administrator* and a *Paper* was defined as functional and inverse functional. This leads to an inconsistency when a conference administrator accepts more than one paper. *cmt:acceptPaper* has been changed to be only inverse functional.
- *conference*: *conference:Contribution_1st_author* was disjoint with *conference:Contribution_co-author*, which lead to an inconsistency when a person was at the same time the first author of a paper and the co-author of another paper. The disjunction axiom from the ontology has been then removed.

If a CQA was not exactly covered by an ontology, the ontology would not be populated with its associated instances. This results in an uneven population of equivalent concepts in the ontologies. For example, considering the *ekaw* and *cmt* ontologies, which both contain a *Document* class. *"What are the documents?"* was not a CQA whereas *paper, review, web site* and *proceedings* were the focus of CQAs. While *ekaw:Document* class has for subclasses *ekaw:Paper*, *ekaw:Review*, *ekaw:Web_Site* and *ekaw:Conference_Proceedings*, *cmt:Document* has only two subclasses *cmt:Paper* and *cmt:Review*. *ekaw:Document* will, by consequence of its subclasses, be populated with paper, review, website and proceedings instances whereas *cmt:Document* will be populated with paper and review instances only.

#### 6.3.2. Conference data analysis

In order to populate the conference ontologies and make it close to real scenarios, some figures from past conferences have been analysed. The information from ISWC 2018 and ESWC 2017 from Scholarly Data[9] complemented the ESWC 2018 website data for this analysis. Indeed, some information such as which program committee member reviewed which paper does not appear in Scholarly Data and the ESWC 2018 website did not show which person is affiliated to which organisation. Some points could be observed:

- percentage of accepted papers having at least a program committee member as author: 44% for ESWC 2017 and 59% for ISWC 2018

[8] https://2018.eswc-conferences.org/

[9] http://www.scholarlydata.org/

- distribution of the number of authors per submitted papers (ESWC 2018): 1 (6%), 2 (17%), 3 (29%), 4 (26%), 5 (9%), 6 (8%) ou 7-10 (2%)
- distribution of the number of collaborating institutions per accepted papers over scholarly data (global represents the statistics over all data from the scholarly data endpoint):

| nb inst. | global | ESWC 2017 | ISWC 2018 |
|---|---|---|---|
| 1 | 56% | 40% | 40% |
| 2 | 18% | 16 % | 30% |
| 3 | 10 % | 10 % | 17% |
| 4 | 6% | 7 % | 7% |
| 5 | 5% | 6% | 5% |
| 6+ | between 0 and 2 % | | |

- distribution of the number of authors per accepted papers over scholarly data:

| nb auth. | global | ESWC 2017 | ISWC 2018 |
|---|---|---|---|
| 1 | 12% | 7% | 13 % |
| 2 | 21% | 11% | 14% |
| 3 | 27% | 28% | 24% |
| 4 | 19% | 25% | 23% |
| 5 | 17% | 17% | 14% |
| 6 | 5% | 5% | 6% |
| 7+ | between 0 and 4 % | | |

### *6.3.3. Population of conference ontologies*

The first population of the ontologies with the ESWC 2018 data left some important knowledge unrepresented. For example, the concepts of external reviewer, presenter of a paper, and person affiliation, which appeared important for a conference organisation were not available on the website. Always in the perspective of conference organisation, the conference ontologies were browsed to complete the list of CQAs with useful concepts. The pivot format and associated SPARQL INSERT queries were also extended to cover the new list of CQAs. Then, the next step was to artificially generate the pivot format instantiation. For that, a score between 1 and 10 is given to each conference. This score pseudo-determines the number of submitted papers, program committee members, etc. as shown in Table 8.

The statistics from the ESWC 2018, ISWC 2018, ESWC 2017 datasets were globally reproduced: 50% of papers have at least a program committee member as author, the number of authors per paper is 1 (6%), 2 (17%), 3 (29%), 4 (26%), 5 (9%), 6 (8%) or 7-10 (2%), the number of collaborating institutions is around 1 (40%), 2(30%), 3 (17%), 4 (7%), 5 (5%) 6(2%). These statistics are pointers, as the generation process is pseudo-random, these figures may vary in practice. Some proportions were arbitrarily chosen: 20% of the submitted papers are poster papers, and 20% are demo papers, the regular paper acceptance rate is in [0.1 – 0.7] and a poster/demo paper acceptance rate is in [0.4 – 1.0], 20% of the reviews are done by an external reviewer.

Table 8
Number of submitted papers, pc members, etc. for a conference of size 1 and 10 (min – max values).

| Number of | Size 1 | Size 10 |
|---|---|---|
| submitted papers | 40 – 45 | 940 – 990 |
| people | 300 – 330 | 1830 – 2130 |
| pc members | 50 – 52 | 500 – 530 |
| oc members | 20 – 22 | 110 – 140 |
| sc members | 15 – 17 | 60 – 90 |
| institutions | 30 – 32 | 210 – 240 |
| tutorials | 1 – 2 | 10 – 11 |
| workshops | 1 – 2 | 19 – 20 |
| tracks | 1 | 6 |

In order to evaluate statistics-based matchers on the benchmark, different sets of population were considered for the ontologies. The idea is to provide the same conference ontologies but with partially overlapping set of instances (instances linked with *owl:sameAs*). To do so, 6 sets of instance population with a more or less important overlapping parts were created. Each ontology is populated with different conferences[10] (with absolutely no common instance between the conferences –no common person, no common paper, etc.). This ensures that there is a quantifiable common part and that the ontologies are consistent. As a result, 6 artificial datasets were created with 25 artificial conferences:

- 0 %: 5 different conferences per ontology
- 20 %: 1 common conference for all ontologies and 4 different conferences per ontology
- 40 %: 2 common and 3 different conferences
- 60 %: 3 common and 2 different conferences
- 80 %: 4 common and 1 different conference
- 100 %: 5 common conferences for all ontologies

Note that the percentage given in the name of the datasets is the percentage of common conference event instances per ontology. As the size of each conference is different, the percentage of common instances (papers, authors, etc.) will not be same. In Table 9, the minimum and maximum percentage of the common paper instances is given for each dataset.

[10] A *conference* here refers to the data related to a conference event.

Table 9

Percentage (min, max) of common submitted papers in the different datasets. The second line reads *"In the 20% dataset, the proportion of common paper instances is between 7 and 11 %"*. Which means that for one of the ontologies, the common part of paper instances represents 7% of all its paper instances. For another ontology, the common part of paper instances represents 11% of all its paper instances.

| Dataset | Min | Max |
|---|---|---|
| 0% | 0% | 0% |
| 20 % | 7% | 11 % |
| 40 % | 29% | 51% |
| 60 % | 40 % | 57% |
| 80 % | 57% | 84 % |
| 100 % | 100 % | 100 % |

Not all the ontology concepts were covered by the pivot CQAs. Table 10 shows the number of entities covered by the CQAs, *i.e.*, instantiated after the CQA-based population, in each ontology.

Table 10

Number of populated entities by ontology. Number of populated entities / number of entities in the original ontology.

| | cmt | conference | confOf | edas | ekaw |
|---|---|---|---|---|---|
| Classes | 26 / 30 | 51 / 60 | 29 / 39 | 43 / 104 | 57 / 74 |
| Obj. prop. | 43 / 49 | 37 / 46 | 10 / 13 | 17 / 30 | 26 / 33 |
| Data prop. | 7 / 10 | 13 / 18 | 10 / 23 | 11 / 20 | 0 / 0 |

### 6.4. CQA for evaluation creation

For the evaluation, the focus is on CQAs which can actually be covered by two or more ontologies. To write the CQAs which will be used in the dataset, the list of CQAs used for the population was trimmed:

- the CQAs which were only covered by one ontology
- some CQAs which were not considered relevant such as "What is the name of a reception?", the answer being an *rdfs:label* "Reception" for all reception instances.

The remaining CQAs were then written as SPARQL SELECT queries by adapting the SPARQL INSERT queries. Table 11 shows the number of CQAs which were covered by the pivot format, by each ontology (in the SPARQL INSERT queries) and which were transformed into SPARQL SELECT queries for the evaluation dataset. 278 SPARQL SELECT queries result from this process.

Table 11

Number of initial (pivot) CQAs covered by each ontology and number of evaluation (eval) CQAs covered by each ontology.

| | cmt | conference | confOf | edas | ekaw | total |
|---|---|---|---|---|---|---|
| pivot | 46 | 90 | 67 | 60 | 84 | 152 |
| eval | 34 | 73 | 54 | 52 | 65 | 100 |

## 7. Evaluation

Existing alignments over the conference dataset were evaluated with the proposed evaluation system. The dataset used for the evaluation is the 100 % dataset so that instance-based precision can be measured.

### 7.1. Evaluated alignments

Existing alignments between the Conference ontologies in EDOAL format[11] [39] have been evaluated. The EDOAL format was necessary so that the alignments could be processed by the rewriting systems. Five alignments have been evaluated. The number of ontology pairs (out of 10 pairs) that these alignments cover are indicated in the following.

**Query_rewriting** the query rewriting oriented alignment set[12] from [22]. It has been manually generated and is composed of 431 correspondences with 191 complex correspondences from 17 different patterns (some patterns are composite) - 10 pairs of ontologies

**Ontology_merging** the ontology merging oriented alignment set[12] from [22]. It has been manually generated and is composed of 313 correspondences with 54 complex correspondences from 9 different patterns (some patterns are composite) - 10 pairs of ontologies.

**ra1** the reference simple alignment[13] from the conference dataset [18]. This dataset is limited to simple alignments between 7 ontologies - 10 pairs of ontologies.

**Ritze_2010** the output alignment[12] from [4] (automatically generated) - complex correspondences found on 4 pairs of ontologies. This alignment is the smallest one as only one correspondence has been found for each pair.

**Faria_2018** the output alignment from [5] (automatically generated) - alignments between 3 pairs

[11] http://alignapi.gforge.inria.fr/edoal.html
[12] https://doi.org/10.6084/m9.figshare.4986368.v7
[13] http://oaei.ontologymatching.org/2018/conference/

publicly available. It is composed of two types of complex equivalence correspondences: those with attribute occurrence restriction and those with attribute domain restriction. These are the alignments available in the context of the OAEI 2018 campaign[14].

The ra1 alignment had been used as input by the systems of Ritze_2010 and Faria_2018. Ra1 has been added to these two alignments for the CQA coverage evaluation. The precision evaluation was made only on the complex correspondences (the output of the original approaches).

### 7.2. CQA coverage

The CQA coverage evaluation was run over all datasets in order to measure the standard deviation of the query precision, recall and f-measure between the datasets, as shown in Table 12. The standard deviation is maximal for Faria_2018 and Ritze_2010, but is still rather low ($10^{-3}$). As the standard deviation is low, the CQA coverage evaluation was performed over the 100% dataset so that the same dataset could be used for CQA coverage and instance-based precision evaluation (Table 13). Ritze_2010 and Faria_2018 both have better coverage than ra1 that they include. It means that the complex correspondences in these alignments are indeed a complement to the simple ones.

Globally, as shown in Table 13, the Query_rewriting alignments have a better coverage than the others. An exception for the edas-confOf pair could be noted. The Ontology_merging alignment outperforms the Query_rewriting one. This is explained by the choice made in the methodology for the creation of both alignments combined with the rewriting systems. In the Ontology_merging alignments, unions of properties were separated into individual subsumptions which were usable by the rewriting system, whereas in the Query_rewriting one, the subsumptions were unions. For example:

Query_rewriting correspondence:

$\langle$ *confOf:starts_on, edas:startDate* $\sqcup$ *edas:hasStartDate,* $\sqsupseteq$*, 1.0*$\rangle$
$\langle$ *confOf:Conference.confOf:starts_on.*$\top$*, edas:startDate,* $\equiv$*, 1.0*$\rangle$

Ontology_merging correspondences:

$\langle$*confOf:starts_on, edas:startDate,* $\sqsupseteq$*, 1.0*$\rangle$
$\langle$*confOf:starts_on, edas:hasStartDate,*$\sqsupseteq$*,1.0* $\rangle$

Therefore, when a query contained the *edas:hasStartDate* relation, the Ontology_merging correspondence could be used, but the Query_rewriting ones could not. The precision-oriented methodology prevented the addition of the two Ontology_merging correspondences to the Query_rewriting alignment.

When closely looking at the results, many CQAs retrieving literals (titles, names, etc.) were not rewritten by the alignments. This is mainly explained because the *rdfs:label* property was introduced in the population phase when no labelling property was included in the original ontologies. The CQAs which needed (c:c) correspondences to be rewritten were not covered by the evaluated alignments. Indeed, these alignments are restricted to (s:s), (s:c) and (c:s) correspondences.

### 7.3. Intrinsic precision

Table 14 shows the precision of the alignments considering different sets of correspondences as correct. The *equivalent* precision is calculated by considering that only the correspondences whose members are *equivalent* are correct. The *subsumed* precision considers correct the correspondences whose members subsume one another (this includes the equivalent ones). The *overlapping* precision considers correct the correspondences with equivalent, subsumed or overlapping members. The *not disjoint* precision considers all correspondences whose members are not disjoint correct. The difference with the *overlapping* one is that an empty correspondence is correct in this case.

The real precision of the alignments is considered to be between the *equivalent* and the *not disjoint* values. The Query_rewriting, Ontology_merging alignments do not have a very good equivalent precision score (0.42 and 0.43). Indeed, their correspondences include a lot of subsumptions. For the subsumed, overlapping and not disjoint scores, their scores are much higher (0.94 and 0.91).

ra1 has a better equivalence score (0.56) than the other two manually created alignments because it originally contains only correspondence with an equivalence relation. However, given this score seems low for a reference alignment. This low score is partly due to the different CQA coverage of the ontologies in the population phase. For example, for the pair *cmt-edas*, the ra1 correspondence $\langle$*cmt:Document, edas:Document,* $\equiv$*, 1.0* $\rangle$ is a subsumption in the on-

[14]http://oaei.ontologymatching.org/2018/results/complex/conference/index.html

Table 12
Standard deviation and average of the query precision, query f-measure and query recall scores over the 6 datasets.

| | | Query_rewriting | Ontology_merging | ra1 | Faria_2018 | Ritze_2010 |
|---|---|---|---|---|---|---|
| Standard deviation | Precision | $1.45\times10^{-3}$ | $1.48\times10^{-3}$ | $6.75\times10^{-4}$ | $2.74\times10^{-3}$ | $1.64\times10^{-3}$ |
| | F-measure | $5.55\times10^{-4}$ | $7.95\times10^{-4}$ | $6.87\times10^{-4}$ | $2.65\times10^{-3}$ | $1.76\times10^{-3}$ |
| | Recall | $3.89\times10^{-4}$ | $1.17\times10^{-3}$ | $7.26\times10^{-4}$ | $2.63\times10^{-3}$ | $1.91\times10^{-3}$ |
| Average | Precision | 0.6963 | 0.6398 | 0.4283 | 0.4225 | 0.4852 |
| | F-measure | 0.6889 | 0.6338 | 0.4227 | 0.4164 | 0.4789 |
| | Recall | 0.7025 | 0.6528 | 0.4245 | 0.4164 | 0.4795 |

Table 13
Average of CQA f-measure for each pair of ontologies for each alignment on the 100% dataset.

| pair | Query_rewriting | Ontology_merging | ra1 | Faria_2018 | Ritze_2010 |
|---|---|---|---|---|---|
| cmt-conference | 0.70 | 0.57 | 0.31 | 0.45 | |
| cmt-confOf | 0.69 | 0.69 | 0.69 | | |
| cmt-edas | 0.65 | 0.65 | 0.41 | | 0.53 |
| cmt-ekaw | 0.65 | 0.64 | 0.25 | 0.42 | 0.34 |
| conference-cmt | 0.69 | 0.59 | 0.28 | 0.41 | |
| conference-confOf | 0.50 | 0.48 | 0.43 | | |
| conference-edas | 0.66 | 0.52 | 0.48 | | 0.48 |
| conference-ekaw | 0.48 | 0.45 | 0.33 | 0.36 | |
| confOf-cmt | 0.77 | 0.71 | 0.72 | | |
| confOf-conference | 0.73 | 0.56 | 0.45 | | |
| confOf-edas | 0.87 | 0.74 | 0.28 | | |
| confOf-ekaw | 0.83 | 0.72 | 0.51 | | 0.54 |
| edas-cmt | 0.73 | 0.67 | 0.43 | | 0.54 |
| edas-conference | 0.63 | 0.52 | 0.50 | | 0.50 |
| edas-confOf | 0.56 | 0.70 | 0.30 | | |
| edas-ekaw | 0.92 | 0.83 | 0.50 | | |
| ekaw-cmt | 0.66 | 0.65 | 0.27 | 0.46 | 0.36 |
| ekaw-conference | 0.51 | 0.46 | 0.34 | 0.38 | |
| ekaw-confOf | 0.74 | 0.74 | 0.45 | | 0.52 |
| ekaw-edas | 0.77 | 0.77 | 0.50 | | |
| **Average** | **0.69** | **0.63** | **0.42** | **0.41** | **0.48** |

Table 14
Different precision metrics over the alignments. The name of the precision metric is the relation between a correspondence member which is considered correct. For example, in the *equivalent* precision, the correspondences whose members were found equivalent is considered correct, the other correspondences not correct.

| Average Precision | Query_rewriting | Ontology_merging | ra1 | Faria_2018 | Ritze_2010 |
|---|---|---|---|---|---|
| equivalent | 0.42 | 0.43 | 0.56 | 0.65 | 0.75 |
| subsumed | 0.80 | 0.80 | 0.83 | 0.71 | 0.75 |
| overlapping | 0.90 | 0.86 | 0.92 | 0.71 | 0.75 |
| not disjoint | 0.94 | 0.91 | 0.96 | 0.71 | 0.75 |

tology population. *cmt:Document* has for subclasses *cmt:Paper* and *cmt:Review*, whereas *edas:Document* has for subclasses *edas:Paper*, *edas:Review*, *edas:Programme* and *edas:SlideSet* which were all populated. Therefore, even if the correspondence is correct with an equivalence relation, its instance interpretation is a subsumption. Note that the instance interpretation could also be an overlap is *cmt* had another subclass (e.g., Website) which did not appear in *edas*.

The low *equivalence* score of ra1 is also due to the different interpretation of the ontologies. For example, in the pair *cmt-confOf*, the ra1 correspondence $\langle$ *cmt:hasAuthor, confOf:writtenBy,* $\equiv$*, 1.0* $\rangle$ is a subsumption in the ontology population. *cmt:hasAuthor* was interpreted as the *"has 1st author"* relationship because of its functional property (Section 6.3.1).

Ritze_2010 has only equivalent or disjoint correspondences, therefore its precision scores are the same for all metrics. Faria_2018 achieves a good precision score overall (between 0.65 and 0.71).

Given the different population issues, the overlapping and not disjoint scores give a good representation of the alignment precision.

### 7.4. Discussion

Table 15 shows the results of the evaluation over the alignments. The CQA coverage and precision scores have been aggregated in an harmonic mean (called HM in Table 15). Overall, the Query_rewriting and Ontology_merging alignments have the better results. This is satisfactory given that these two alignments are complex reference alignments on this dataset. Even if ra1 has the best precision, its low CQA coverage (0.42) shows that a lot of CQAs from the benchmark need complex alignments to be covered. Faria_2018 and Ritze_2010 are compared to the other even if they do not contain the same number of pairs. Therefore, these numbers cannot be exactly compared to the others.

In the results of the OAEI 2018 [31], the precision measured for the Faria_2018 alignment was 0.54 (cf. Table 16). The instance-based precision gives the same result as the manual evaluation for the *cmt-ekaw* pair. For the other pairs, the gap is quite important. For the *cmt-conference* pair, this is probably due to a difference of interpretation of the ontologies. The *conference:Written_contribution* being considered as a superclass of *cmt:Paper* in the OAEI 2018 evaluation, but equivalent classes in the ontology population.

In the *conference-ekaw* pair, the $\langle \exists$*conference:was_a_track-workshop_chair_of. conference:Tutorial, ekaw:Tutorial_Chair,* $\equiv$*, 0.369*$\rangle$ was considered correct in the OAEI 2018 evaluation. However, an axiom of the *conference* ontology restrains the domain of *conference:was_a_track-workshop_chair_of* to *conference:Track* $\sqcup$ *conference:Workshop*. This has been taken into account in the ontology population and the correspondence was evaluated as disjoint for the evaluation system.

## 8. Conclusions and future work

This paper has presented an evaluation benchmark on which the approaches generating complex correspondences can be evaluated. In general, alignment evaluation is often performed by comparing a generated alignment to a reference one. It involves comparing the members of the correspondences generated by the systems to the members of the correspondences in the reference alignment. While this comparison is straightforward for simple alignments, this step becomes harder when dealing with complex correspondences. For example, these three correspondences can be considered as true positive: ($o$:AcceptedPaper,$\exists o'$:hasDecision.$o'$:Acceptance,$\equiv$), ($\exists o'$:accepted.{true},$\exists o''$:hasDecision.$o''$:Acceptance, $\equiv$), or ($o$:AcceptedPaper,$\exists o'$: acceptedBy.$\top$,$\equiv$).

While syntactic-oriented evaluation metrics (measuring the effort to transform a correspondence into another) would fail in covering the high space of possible combinations between constructors, semantic-oriented approaches would restrict the expressiveness of correspondences to those supported by current reasoners, leaving aside for instance, transformation functions. Hence, comparison of instance sets seems to be reasonable. Our proposal shifts the problem to the comparison of instances in a task of query rewriting targeting user needs. We proposed two evaluation measures. While the *CQA coverage* measure relies on pairs of equivalent SPARQL queries (source and target queries) and measures how well an evaluated alignment covers these queries, the *intrinsic precision* compares the instances of the correspondences members.

CQA coverage, in particular, requires however a way for rewriting the source query into the target query, in terms of the evaluated alignment. Such an evaluation however requires that the ontologies of the evaluation dataset are consistently populated and a system for rewriting the queries. With respect to the for-

Table 15
CQA coverage and equivalence, overlapping and not disjoint precision of the alignments, harmonic mean (HM) of the two scores.

| Metric | Query_rewriting | Ontology_merging | ra1 | Faria_2018 | Ritze_2010 |
|---|---|---|---|---|---|
| CQA Coverage | **0.69** | 0.63 | 0.42 | 0.41 | 0.48 |
| Precision overlapping | 0.90 | 0.86 | **0.92** | 0.71 | 0.75 |
| Precision not disjoint | 0.94 | 0.91 | **0.96** | 0.71 | 0.75 |
| HM overlap | **0.78** | 0.73 | 0.58 | 0.52 | 0.59 |
| HM not disjoint | **0.80** | 0.74 | 0.58 | 0.52 | 0.59 |

Table 16
Comparison of the OAEI 2019 and instance-based precision metrics over the Faria_2018 alignment. The not disjoint, subsumed and overlap precision scores are the same for this alignment.

| pair | OAEI 2018 | equivalent | not disjoint |
|---|---|---|---|
| cmt-conference | 0.4 | 1.00 | 1.00 |
| cmt-ekaw | 0.86 | 0.86 | 0.86 |
| conference-ekaw | 0.36 | 0.09 | 0.27 |
| Average | 0.54 | 0.65 | 0.71 |

mer, this problem has been addressed here by proposing an artificially and regularly populated dataset, as datasets with cross-ontology consistency may not be easy to find. The population process was guided by CQAs. We argue that the synthetic population ensures that each CQA is consistently populated across the ontologies. However, one can argue that in case the CQAs have different coverage for correspondences achieved through different patterns, this may have an impact on evaluation. As our evaluation is instance-based, two correspondences that do not exactly follow the same pattern but that represent the same piece of knowledge, will be considered to be comparable.

With respect to the query rewriting systems, most existing SPARQL rewriting system are limited (s:c) correspondences and dealing with (c:c) correspondences is still a challenge. A rewriting system which deals with such correspondences has been proposed here. However, it can not combine several (c:c) correspondence together. Instance-based rewriting could, however, be a new lead for this challenge. While the two systems have been manually evaluated in the task of rewriting queries, in the way discussed in [38], we did not evaluate the impact of each of the systems in the evaluation task. While it has to be done, we reduce their potential impact by choosing the best rewriting query, by selecting the one with the best f-measure. Another point is that these systems do not take into account correspondence relation and confidence within the rewrite process, what has to be addressed in the future.

The proposed approach has been applied for evaluating existing alignments. This system has also been applied for automating the evaluation of complex alignments in the OAEI 2019 campaign. The evaluation reported here shows that the reference alignments all have a good precision score and that complex alignments provide a better coverage of the CQAs than simple alignments. The evaluation of the alignments from two complex matchers shows that, even though both achieve a rather good precision, their CQA coverage is below 0.5. However, these results are far from the ones obtained with the original dataset and reported in OAEI campaigns, leaving a large room for improvements in the field. As our approach requires the alignments to be a priori known, it is suitable for scenarios such as the ones in OAEI. In that sense, as for the largely used artificial datasets, as the OAEI Benchmark, our dataset covers a lack of complex datasets under which an automatic evaluation can be carried in a controlled manner.

Evaluating complex ontology alignments, however, is a too broad challenge to be tackled with a single approach, as there are multiple aspects to take into account. A complementary approach to the instance-based one proposed in this paper could be an edit-distance approach that would reflect the effort involved in human validation. The approach should be also scalable, and avoid the need to do all correspondence comparisons. This could also be achieved by considering the possibility of computing minimal complex correspondences (or key complex correspondences, which can be used for computing all the other ones), in line with the work of [47]. In order to cover ontologies of various sizes and domains, developing a query generation system able to automatically generate queries adequate in coverage and scope to the evaluation of complex alignments could also help in the evaluation task.

## References

[1] P.R. Visser, D.M. Jones, T.J. Bench-Capon and M. Shave, An analysis of ontology mismatches; heterogeneity versus inter-

operability, in: *AAAI 1997 Spring Symposium on Ontological Engineering, Stanford CA., USA*, 1997, pp. 164–72.

[2] A. Maedche, B. Motik, N. Silva and R. Volz, MAFRA—A MApping FRAmework for Distributed Ontologies, in: *International Conference on Knowledge Engineering and Knowledge Management*, Springer, 2002, pp. 235–250.

[3] D. Ritze, C. Meilicke, O. Šváb-Zamazal and H. Stuckenschmidt, A pattern-based ontology matching approach for detecting complex correspondences, in: *4th ISWC workshop on ontology matching*, 2009, pp. 25–36.

[4] D. Ritze, J. Völker, C. Meilicke and O. Šváb-Zamazal, Linguistic analysis for complex ontology matching, in: *5th workshop on ontology matching*, 2010, pp. 1–12.

[5] D. Faria, C. Pesquita, B.S. Balasubramani, T. Tervo, D. Carriço, R. Garrilha, F.M. Couto and I.F. Cruz, Results of AML participation in OAEI 2018, *Ontology Matching* **2288** (2018), 125–131.

[6] S. Jiang, D. Lowd, S. Kafle and D. Dou, Ontology matching with knowledge rules, in: *Transactions on Large-Scale Data- and Knowledge-Centered Systems XXVIII*, Springer, 2016, pp. 75–95.

[7] R. Parundekar, C.A. Knoblock and J.L. Ambite, Linking and building ontologies of linked data, in: *ISWC*, Springer, 2010, pp. 598–614.

[8] R. Parundekar, C.A. Knoblock and J.L. Ambite, Discovering concept coverings in ontologies of linked data sources, in: *ISWC*, Springer, 2012, pp. 427–443.

[9] B. Walshe, R. Brennan and D. O'Sullivan, Bayes-ReCCE: A Bayesian Model for Detecting Restriction Class Correspondences in Linked Open Data Knowledge Bases, *International Journal on Semantic Web and Information Systems (IJSWIS)* **12**(2) (2016), 25–52.

[10] E. Thiéblin, O. Haemmerlé and C. Trojahn, Complex matching based on competency questions for alignment: a first sketch, in: *OM 2018 - 13th ISWC workshop on ontology matching*, 2018.

[11] B.P. Nunes, A. Mera, M.A. Casanova, K.K. Breitman and L.A.P. Leme, Complex Matching of RDF Datatype Properties, in: *Proceedings of the 6th International Conference on Ontology Matching-Volume 814*, CEUR-WS. org, 2011, pp. 254–255.

[12] H. Qin, D. Dou and P. LePendu, Discovering executable semantic mappings between ontologies, in: *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, Springer, 2007, pp. 832–849.

[13] R. Dhamankar, Y. Lee, A. Doan, A. Halevy and P. Domingos, iMAP: discovering complex semantic matches between database schemas, in: *Proceedings of the 2004 ACM SIGMOD international conference on Management of data*, ACM, 2004, pp. 383–394.

[14] B. He, K.C.-C. Chang and J. Han, Discovering complex matchings across web query interfaces: a correlation mining approach, in: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM Press, 2004, pp. 148–157. doi:10.1145/1014052.1014071.

[15] E. Thieblin, O. Haemmerle, N. Hernandez and C. Trojahn, Survey on complex ontology matching, *Semantic Web Journal* (2019).

[16] E. Thiéblin, M. Cheatham, C. Trojahn, O. Zamazal and L. Zhou, The First Version of the OAEI Complex Alignment Benchmark, in: *ISWC Posters and Demos*, Springer, 2018.

[17] J. Euzenat, M. Rosoiu and C.T. dos Santos, Ontology matching benchmarks: Generation, stability, and discriminability, *J. Web Sem.* **21** (2013), 30–48.

[18] O. Zamazal and V. Svátek, The Ten-Year OntoFarm and its Fertilization within the Onto-Sphere, *Web Semantics: Science, Services and Agents on the World Wide Web* **43** (2017), 46–53, ISSN 15708268. doi:10.1016/j.websem.2017.01.001.

[19] É. Thiéblin, Do Competency Questions for Alignment Help Fostering Complex Correspondences?, in: *Proceedings of the EKAW Doctoral Consortium 2018 co-located with the 21st International Conference on Knowledge Engineering and Knowledge Management (EKAW 2018), Nancy, France, November 13, 2018.*, 2018. http://ceur-ws.org/Vol-2306/paper8.pdf.

[20] G. Stapleton, J. Howse, A. Bonnington and J. Burton, A vision for diagrammatic ontology engineering, in: *International Workshop on Visualizations and User Interfaces for Knowledge Engineering and Linked Data Analytics*, CEUR-WS. org, 2014, pp. 1–13. http://eprints.brighton.ac.uk/13046/.

[21] J. Euzenat and P. Shvaiko, *Ontology Matching*, Springer Berlin Heidelberg, 2013. ISBN 978-3-642-38720-3 978-3-642-38721-0.

[22] E. Thiéblin, O. Haemmerlé, N. Hernandez and C. Trojahn, Task-Oriented Complex Ontology Alignment: Two Alignment Evaluation Sets, in: *The Semantic Web*, A. Gangemi, R. Navigli, M.-E. Vidal, P. Hitzler, R. Troncy, L. Hollink, A. Tordai and M. Alam, eds, Lecture Notes in Computer Science, Springer International Publishing, 2018, pp. 655–670. ISBN 978-3-319-93417-4.

[23] M. Grüninger and M.S. Fox, Methodology for the Design and Evaluation of Ontologies. International Joint Conference on Artificial Inteligence, in: *Workshop on Basic Ontological Issues in Knowledge Sharing*, Vol. 15, 1995.

[24] Y. Ren, A. Parvizi, C. Mellish, J.Z. Pan, K. van Deemter and R. Stevens, Towards Competency Question-Driven Ontology Authoring, in: *The Semantic Web: Trends and Challenges*, Vol. 8465, Springer, 2014, pp. 752–767. ISBN 978-3-319-07442-9 978-3-319-07443-6. doi:10.1007/978-3-319-07443-6_50.

[25] C. Meilicke and H. Stuckenschmidt, Incoherence as a basis for measuring the quality of ontology mappings, in: *Proceedings of the 3rd International Conference on Ontology Matching-Volume 431*, CEUR-WS. org, 2008, pp. 1–12. http://dl.acm.org/citation.cfm?id=2889699.

[26] A. Solimando, E. Jiménez-Ruiz and G. Guerrini, Detecting and Correcting Conservativity Principle Violations in Ontology-to-Ontology Mappings, in: *The Semantic Web – ISWC 2014*, Vol. 8797, P. Mika, T. Tudorache, A. Bernstein, C. Welty, C. Knoblock, D. Vrandečić, P. Groth, N. Noy, K. Janowicz and C. Goble, eds, Springer International Publishing, Cham, 2014, pp. 1–16. ISBN 978-3-319-11914-4 978-3-319-11915-1. doi:10.1007/978-3-319-11915-1_1. http://link.springer.com/10.1007/978-3-319-11915-1_1.

[27] W. Su, J. Wang and F. Lochovsky, Holistic Schema Matching for Web Query Interfaces, in: *Advances in Database Technology - EDBT 2006*, Y. Ioannidis, M.H. Scholl, J.W. Schmidt, F. Matthes, M. Hatzopoulos, K. Boehm, A. Kemper, T. Grust and C. Boehm, eds, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006, pp. 77–94.

[28] Y. An, X. Hu and I. Song, Learning to discover complex mappings from web forms to ontologies, in: *21st ACM International Conference on Information and Knowledge Management, CIKM'12, Maui, HI, USA, October 29 - November 02, 2012*, X. Chen, G. Lebanon, H. Wang and M.J. Zaki, eds, ACM, 2012, pp. 1253–1262. doi:10.1145/2396761.2398427.

[29] E. Thiéblin, F. Amarger, N. Hernandez, C. Roussey and C. Trojahn, Cross-Querying LOD Datasets Using Complex Alignments: An Application to Agronomic Taxa, in: *Research Conference on Metadata and Semantics Research*, Springer, 2017, pp. 25–37.

[30] L. Zhou, M. Cheatham, A. Krisnadhi and P. Hitzler, A Complex Alignment Benchmark: GeoLink Dataset, in: *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference, Monterey, CA, USA, October 8-12, 2018, Proceedings, Part II*, 2018, pp. 273–288. doi:10.1007/978-3-030-00668-6_17. https://doi.org/10.1007/978-3-030-00668-6_17.

[31] A. Algergawy, M. Cheatham, D. Faria, A. Ferrara, I. Fundulaki, I. Harrow, S. Hertling, E. Jiménez-Ruiz, N. Karam, A. Khiat, P. Lambrix, H. Li, S. Montanelli, H. Paulheim, C. Pesquita, T. Saveta, D. Schmidt, P. Shvaiko, A. Splendiani, É. Thiéblin, C. Trojahn, J. Vatascinová, O. Zamazal and L. Zhou, Results of the Ontology Alignment Evaluation Initiative 2018, in: *Proceedings of the 13th International Workshop on Ontology Matching co-located with the 17th International Semantic Web Conference, OM@ISWC 2018, Monterey, CA, USA, October 8, 2018.*, 2018, pp. 76–116. http://ceur-ws.org/Vol-2288/oaei18_paper0.pdf.

[32] M. Ehrig and J. Euzenat, Relaxed Precision and Recall for Ontology Matching, in: *Integrating Ontologies '05, Proceedings of the K-CAP 2005 Workshop on Integrating Ontologies, Banff, Canada, October 2, 2005*, 2005. http://ceur-ws.org/Vol-156/paper5.pdf.

[33] É. Thiéblin, O. Haemmerlé and C. Trojahn, CANARD complex matching system: results of the 2018 OAEI evaluation campaign, in: *OM@ISWC*, CEUR Workshop Proceedings, Vol. 2288, CEUR-WS.org, 2018, pp. 138–143.

[34] A. Solimando, E. Jiménez-Ruiz and C. Pinkel, Evaluating ontology alignment systems in query answering tasks, in: *Proceedings of the 2014 International Conference on Posters & Demonstrations Track-Volume 1272*, CEUR-WS. org, 2014, pp. 301–304.

[35] J. David, J. Euzenat, P. Genevès and N. Layaïda, Evaluation of Query Transformations without Data: Short paper, in: *Companion of the The Web Conference 2018 on The Web Conference 2018*, International World Wide Web Conferences Steering Committee, 2018, pp. 1599–1602.

[36] L. Hollink, M. Van Assem, S. Wang, A. Isaac and G. Schreiber, Two Variations on Ontology Alignment Evaluation: Methodological Issues, in: *5th European Semantic Web Conference*, 2008, pp. 388–401.

[37] K. Makris, N. Bikakis, N. Gioldasis and S. Christodoulakis, SPARQL-RW: transparent query access over mapped RDF data sources, in: *Proceedings of the 15th International Conference on Extending Database Technology*, ACM, 2012, pp. 610–613.

[38] E. Thiéblin, F. Amarger, O. Haemmerlé, N. Hernandez and C. Trojahn, Rewriting SELECT SPARQL queries from 1:n complex correspondences, in: *OM 2016 - 11th ISWC workshop on ontology matching*, 2016, p. 49.

[39] J. David, J. Euzenat, F. Scharffe and C. Trojahn dos Santos, The alignment API 4.0, *Semantic web* **2**(1) (2011), 3–10.

[40] J. Euzenat, A. Polleres and F. Scharffe, Processing Ontology Alignments with SPARQL, in: *2008 International Conference on Complex, Intelligent and Software Intensive Systems*, 2008, pp. 913–917. doi:10.1109/CISIS.2008.126.

[41] G. Correndo and N. Shadbolt, Translating expressive ontology mappings into rewriting rules to implement query rewriting, in: *6th Workshop on Ontology Matching*, 2011.

[42] I. Horrocks, O. Kutz and U. Sattler, The Even More Irresistible SROIQ., *Kr* **6** (2006), 57–67.

[43] M.C. Suárez-Figueroa, A. Gómez-Pérez and M. Fernández-López, The NeOn methodology for ontology engineering, in: *Ontology engineering in a networked world*, Springer, 2012, pp. 9–34.

[44] O. Šváb, V. Svátek, P. Berka, D. Rak and P. Tomášek, Ontofarm: Towards an experimental collection of parallel ontologies, *Poster Track of ISWC* **2005** (2005).

[45] A.G. Nuzzolese, A.L. Gentile, V. Presutti and A. Gangemi, Conference linked data: The scholarlydata project, in: *International Semantic Web Conference*, Springer, 2016, pp. 150–158.

[46] R. Shearer, B. Motik and I. Horrocks, HermiT: A Highly-Efficient OWL Reasoner., in: *OWLED*, Vol. 432, 2008, p. 91.

[47] V. Maltese, F. Giunchiglia and A. Autayeu, Save Up to 99% of Your Time in Mapping Validation, in: *On the Move to Meaningful Internet Systems, OTM 2010 - Confederated International Conferences: CoopIS, IS, DOA and ODBASE, Hersonissos, Crete, Greece, October 25-29*, 2010, pp. 1044–1060.