Semantic Rewriting of SPARQL Queries: The Key Role of Subsumption in Complex Ontology Alignments

Anicet Lepetit Ondo and Laurence Capus
Dep. of Computer and Software Engineering, Laval University, Quebec, G1V 0A6, Quebec, Canada

Keywords: Subsumption Relations, SPARQL Query Rewriting, Aligned Ontology, Large Language Models (LLMs).

Abstract: This article introduces an innovative method for rewriting SPARQL queries in the context of complex ontology alignment by leveraging hierarchical relations such as subClassOf and subPropertyOf. The method relies on generalization and specialization links between concepts to retrieve relevant results, even when strict equivalences are missing. In addition, the use of natural language, assisted by the GPT-4 model, helps address the syntactic complexity of SPARQL and facilitates interaction with ontologies. Unlike existing approaches that focus mainly on simple (s: s) or semi-complex (s: c) alignments based on equivalence between source and target entities, our method reinforces semantic matching by explicitly incorporating subsumption relations. It also integrates complex (c: c) correspondences, which are often overlooked in the literature, thereby improving both query coverage and accuracy. Experiments conducted on ontology datasets in the conference domain confirm the method's ability to capture a wide range of hierarchical relations. While the method is designed to be generic, further evaluations on large-scale ontologies are required to assess its robustness and generalizability.

1 INTRODUCTION

Formal ontologies hold a central place in the Semantic Web, as they enable the structuring and formalization of a domain's semantics in a machine-interpretable manner. However, the coexistence of multiple ontologies for the same domain introduces significant heterogeneity, whether syntactic, terminological, or conceptual, thus compromising data sharing and interoperability between systems. Ontology alignment constitutes a crucial response to this challenge by establishing correspondences between entities from different ontologies (Ondo et al., 2025 ; Amini et al., 2024). These correspondences may be simple (s: s), semi-complex (s: c), or complex (c: c), where "s" denotes an elementary entity and "c" a complex expression, depending on the nature of the entities being aligned.

Homogeneous querying of aligned complex ontologies represents a major challenge in interoperable Semantic Web systems. When a query is initially formulated on a source ontology, it must be rewritten to apply to a target ontology, taking into account syntactic, terminological, and conceptual heterogeneities between the two. While this rewriting can be relatively straightforward when based on elementary correspondences, it becomes significantly more complex when involving composite entities, that is, conceptual expressions formed from multiple interconnected entities (Ondo et al., 2025 ; Thiéblin et al., 2016). This complexity intensifies when alignments include hierarchical relations such as generalization and specialization. These relations are nevertheless essential to capture the semantic richness of ontologies, notably by establishing relevant links even in the absence of strict equivalences, including when they involve complex entities. However, they remain largely underexploited in existing approaches (Ondo et al., 2025 ; Thiéblin et al., 2021, 2016).

This article presents a method for homogeneous querying of aligned ontologies, finely exploiting subsumption relations, including complex correspondences (c: c). Experimental results demonstrate that this approach effectively incorporates various forms of hierarchical relations between source and target entities, thereby highlighting the structuring role of subsumptions in the semantic enrichment of alignments and promoting interoperability in heterogeneous environments.

The remainder of the article is organized as follows: we first present related work, followed by the

Ondo, A. L. and Capus, L.
Semantic Rewriting of SPARQL Queries: The Key Role of Subsumption in Complex Ontology Alignments.
DOI: 10.5220/0013676700004000

In *Proceedings of the 17th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2025) - Volume 2: KEOD and KMIS*, pages 99-107
ISBN: 978-989-758-769-6; ISSN: 2184-3228

proposed methodology, experimental results, and conclude with a discussion and research perspectives.

2 RELATED WORKS

Despite considerable advances over the past decade in the field of ontology alignments, whether simple alignments (of type (s: s)) or complex correspondences (of type (s: c) or (c: c)), research specifically focused on automatic rewriting mechanisms for SPARQL queries between a source ontology and a target ontology remains relatively scarce.

Indeed, the majority of these works have primarily focused on identifying and formalizing equivalence relations between ontological entities, based on simple (s: s) alignments. In this context, each entity in the source ontology is simply replaced by its equivalent in the target ontology by substituting the original IRI with the one specified in the alignment, in order to rewrite the query (David et al., 2011).

Some existing works, although they do not explicitly integrate generalization/specialization (subsumption) relationships into the rewriting process, nevertheless exploit complex correspondences, mainly of the (s: c) type (Thiéblin et al., 2016, 2017). These approaches use transformation rules to adapt SPARQL queries, particularly SELECT queries, to different target vocabularies. However, a deeper integration of subsumption relations and complex correspondences (c: c) may be essential for handling highly structured ontologies.

Correndo et al. (2010) propose a SPARQL query rewriting algorithm based on vocabulary alignments to query heterogeneous RDF sources. While relevant for data integration, support for complex (c: c) correspondences and exploration of concept hierarchies could improve rewriting quality. Similarly, Fujino et al. (2012) translated queries between ontologies by focusing on (s: c) and (c: s) types, without considering complex correspondences or hierarchical relationships. Thiéblin et al. (2021) explore two rewriting strategies based on simple alignments or instance-based approaches, but highlight the potential of more expressive correspondences, such as (c: c).

Thakker et al. (2018) also propose a SPARQL reformulation framework using vocabulary mappings, though without specifying the types used.

Ondo et al. (2025) propose an innovative approach for the automatic rewriting of SPARQL queries across aligned ontologies, taking into account different types of alignments, both simple (s: s) and complex. This method places particular emphasis on complex correspondences (c: c), which constitutes a notable advancement. However, although subsumption relations are recognized as relevant in this context, they are not directly addressed in this work. Their potential is nevertheless highlighted as a relevant direction for future research, opening the way to new avenues of exploration in the field of semantic interoperability.

This work builds upon the foundation laid in Ondo et al. (2025), aiming to strengthen the semantic interoperability of knowledge across aligned ontologies.

3 ADOPTED METHOD

Our method consists in generating a set V and a set $T_{triplet}$, representing respectively the variables and the triples present in the SPARQL query of the source ontology O. Then, taking into account the correspondences based on subsumption between the ontologies, we obtain a set V' and a set $T'_{triplet}$, corresponding to the rewritten query in the target ontology O'.
To formalize this, we define:

$$T_{triplet} = (T_1, T_2, \dots, T_n) \quad (1)$$

Where $T_{triplet}$ is a set of n triples, each triple $T_i = \langle s_i, p_i, o_i \rangle$ for $i \in \{1, 2, \dots, n\}$, representing respectively the subject, predicate, and object in the ontology.

The proposed method combines logical inference, hierarchical extraction, and semantic analysis to identify and exploit relevant concepts from two aligned ontologies, following a structured and concise six-step process.

3.1 Loading and Inference on the Ontology

The process starts by importing the aligned ontology via the Owlready2 library (Lamy, 2020), enabling OWL model manipulation. The ontology is then enriched using the Pellet reasoner, which infers implicit knowledge such as subclass, subproperty, equivalence, and complex axioms, ensuring full reasoning for subsequent semantic processing and query rewriting tasks.

3.2 Hierarchical Identification and Extraction of Source and Target Entities

This step involves exploring the ontology to extract two distinct sets of entities (classes and properties), differentiated by their membership in the source or target ontology, as indicated by IRI prefixes. This separation structures the analysis of correspondences. For each source entity, one or more target correspondences are identified based on hierarchical generalization and specialization relations. When an equivalence relation connects a complex entity pair, dedicated functions are applied to derive logical expressions comprising operators such as conjunction, disjunction, existential and universal quantification, and negation. This analytical phase uncovers explicit complex correspondences, enabling a fine-grained and formally interpretable modeling of the semantic relationships between the aligned ontologies.

3.3 Building a Cross-Knowledge Base

A correspondence table is generated for each entity of the source ontology, relying on the subsumption and equivalence relations established between the two ontologies. This table contains the source super-entity, the semantic sub-entities associated with it in the source ontology, whether atomic elements or complex expressions, as well as the corresponding or inferred sub-entities in the target ontology. All of these correspondences are structured within a formal knowledge base, intended to be reused in our SPARQL query rewriting process.

Figure 1 below illustrates an excerpt from the correspondence table generated by our methodology, which is based on the transitive propagation of subsumption relations across multiple levels. This excerpt highlights all the sub-entities of *Event* in the source ontology along with their possible counterparts (sub-entities) in the target ontology. More precisely, the entities in the target ontology identified as subclasses of the *Event* class in the source ontology are explicitly defined as subclasses of the *Working_event* class in the source ontology due to the hierarchical structure of subclasses. Since *Working_event* is itself a subclass of *Event*, these entities are automatically inferred to be subclasses of *Event*. This inference relies on the transitivity of subsumption, which states that:

Let *A*, *B*, and *C* be three sets def*A* includes the entities from the target ontology that are identified as direct subclasses of *Working_event*. Set *B* represents the *Working_event* entity in the source ontology, while *C* corresponds to the *Event* entity within the same source ontology.

$$if\ :\ \forall x\big(A(x) \rightarrow B(x)\big)\ and\ \forall x\big(B(x) \rightarrow C(x)\big)$$

$$\text{Then, by logical transitivity, } \forall x\big(A(x) \rightarrow C(x)\big)$$

```
Source entity : Event
├── sub-entities (within the same ontology) :
│   - Administrative_event
├──────── - Camera_Ready_event
├──────── - Registration_of_participants_event
├──────── - Reviewing_event
├──────── - Reviewing_results_event
├──────── - Submission_event
│   - Social_event
├──────── - Banquet
├──────── - Reception
├──────── - Trip
│   - Working_event
├──────── - Conference == Conference
├──────── - Tutorial
├──────── - Workshop
├── sub-entities in the target ontology (inferred
│   - Conference = Conference
│   - pc_meeting
│   - session
```

Figure 1: Example of a correspondence table.

3.4 Interaction with the User via Natural Language and AI Model

When a question is posed, a structured prompt is automatically generated, incorporating the previously established semantic correspondence base between entities of the source and target ontologies. This prompt is then submitted to the OpenAI's GPT-4 model, which analyzes the query, identifies the relevant part of the source graph, and extracts two sets of corresponding entities (source side and target side), grouped as a structured paired list. The elements of these two lists represent the components of a hierarchy of entities from the source ontology, as well as their equivalents or subsuming concepts identified in the target ontology.

3.5 An Automatic SPARQL Query Generator

Based on the entities extracted respectively from the source and target ontologies, are automatically generated SPARQL queries. Each entity is analyzed to produce, for each scenario type, a query adapted to the source ontology as well as its translated version for the target ontology. Structural analysis enables the extraction of relevant RDF triples, which then serve as the basis for the dynamic construction of queries. Thus, the generated SPARQL query adapts to the semantic and structural characteristics of the analyzed entities.

3.6 Validation of the Generated SPARQL Queries

To validate the SPARQL queries automatically generated from the semantic correspondences between entities in the source and target ontologies, our method relied on a rigorous process combining multiple levels of verification.

We began by analyzing the structure of the data sources using Protégé 2000. Our method then automatically generated a correspondence table linking equivalent source and target subgraphs. This table was compared against the aligned ontologies to ensure consistency.

When a user submitted a natural language question, our method retrieved the relevant source and target subgraphs from the correspondence table, followed by a compliance check to ensure logical soundness.

Validation continued with a structural and visual comparison of source and target queries, referencing the subgraph table. This was reinforced by a syntactic analysis of the automatically generated SPARQL queries to verify compatibility of the results and confirm the accuracy of the rewriting. This process served as a qualitative evaluation of the fidelity of the generated queries.

4 EXPERIMENT AND RESULTS

This section presents the results obtained by applying our method to several ontology alignment scenarios, illustrating its capability to automatically detect, interpret, and rewrite complex expressions across aligned ontologies. It begins by introducing the dataset used in the experiments, followed by a description of typical cases involving generalization and specialization. For each scenario, we provide the corresponding SPARQL query rewritings, emphasizing their semantic relevance and structural consistency with respect to the detected alignments.

4.1 The Dataset

In the context of querying complex aligned ontologies, it is essential to consider the various correspondence structures that may exist between them. These configurations constitute a fundamental basis for developing effective query answering methods. Consequently, the selection of a suitable dataset is critical to the success of such experiments. However, identifying a fully aligned ontology that enables the evaluation of all possible correspondence scenarios is often challenging and, in some cases, infeasible.

To address these challenges, we enriched two existing datasets (cmt-ConOf.owl) by leveraging a collection of sixteen ontologies derived from the well-known conference dataset, commonly used in research on ontology alignment and querying. This dataset was prioritized due to its widespread adoption within the ontology alignment and query rewriting research community (Thiéblin et al., 2018, 2021 ; Shvaiko et al., 2023 ; Trojahn et al., 2021), and it serves as a well-established reference in these applications.

Furthermore, to assess the robustness and adaptability of the method across different alignment configurations, an additional pair of alignments (cmt–edas.owl), derived from the same dataset but without enrichment, was also considered.

4.2 Scenarios and Results

Our dataset analysis identified several ontology alignment scenarios involving subsumption relations. These include generalization and specialization cases, where source entities relate to target entities with partial correspondences, as well as direct equivalences. These scenarios illustrate the variety and complexity of relationships present in ontology alignments.

4.2.1 Scenario 1: Combinations of SubClassOf Relations with Complex (c: c) Correspondences

This scenario illustrates a typical case of equivalence between two classes from aligned ontologies, each featuring its own internal hierarchy. Defining an equivalence axiom between the central concepts enables the propagation of subsumption relations across both ontologies. Consequently, specializations defined in one ontology can be inferred in the other, thus facilitating semantic integration and enabling coherent query rewriting. Such configurations are crucial in systems that rely on alignments to query heterogeneous data sources effectively.

Let us consider the example shown in Figure 2, inferred from our dataset.

- In the source ontology, the concepts are defined as follows:
  - The class *PeerReviewedJournal*, denoted as c_1, is defined by an equivalence to the following

conjunction: $c_1 \equiv Journal \sqcap usesProcess.value('PeerReview')$

- The class *EngineeringPeerReviewedJournal*, denoted as c_2, is a subclass of $c_1: c_2 \sqsubseteq c_1$

▪ In the target ontology:

- The *class RefereedJournal*, denoted as c_1', is defined as follows: $c_1' \equiv$ *Publication ⊓ governedBy.some(ReviewPolicy).*
- The class *MedicalRefereedJournal*, denoted as c_2', is a subclass of $c_1': c_2' \sqsubseteq c_1'$

An ontological alignment is established between the concepts c_1 (in the source ontology) and c_1'(in the target ontology), in the form of an equivalence axiom:$c_1 \equiv c_1'$.

According to description logic, an axiom of the form *EquivalentClasses (A, B)* implies that:

- $A \sqsubseteq B$ and $B \sqsubseteq A$
- The definitions associated with *A* and *B* are considered interchangeable
- Any entity subsumed by *A* is also subsumed by *B*, and vice versa (Baader et al.,2003).

On this basis, several logical inferences can be drawn:

If $c_1 \equiv c_1'$, $c_2 \sqsubseteq c_1, c_2' \sqsubseteq c_1'$, Then, by transitivity and equivalence:

$$c_2' \sqsubseteq c_1;\ c_2 \sqsubseteq c_1';\ c_1 \sqsubseteq c_1'\ ;\ c_1' \sqsubseteq c_1$$

After performing a test for the same information need expressed through different formulations, Figure 3 illustrates the processing within this category generated by our approach.

Examples of questions used for evaluation include:

- *Which journals make use of the peer-review process?*
- *Can you list all journals that are associated with the peer-review process?*

In this category, the rewriting proceeds as follows, with:

- $\boldsymbol{T(x,C)} \wedge \boldsymbol{T(y,C)}$**:** x and y are of type C (a class of the ontology), with $C = c_1, c_2, \ldots, c_n$
- $\boldsymbol{P(x,p,v)}$: x has the data property p with the value v
- $\boldsymbol{R(x,y,r)}$: there exists a relation r (object property) between x and y, with x and y ∈ C

Based on the results illustrated in Figure 3, we observe that the *WHERE* clause applied to the source ontology satisfies the following conditions:

$T_{triplet} = \{\boldsymbol{A} \cup \boldsymbol{B} \cup \boldsymbol{C}\}$, with:

$$\begin{aligned} A &= \{\, x \mid T(x,c_1) \,\} \\ B &= \{\, x \mid T(x,c_2) \,\} \\ C &= \{\, x \mid T(x,c_3) \wedge P(x,p,v) \,\} \end{aligned} \tag{2}$$

$$\Downarrow$$

$$\begin{aligned} A &= \{\, x \mid T(x, EngineeringPeerReviewedJournal\} \\ B &= \{\, x \mid T(x, PeerReviewedJournal) \,\} \\ C &= \{\, x \mid T(x, Journal) \wedge P(x, usesProcess, PeerReview) \,\} \end{aligned}$$

For the target ontology, the *WHERE* clause satisfies the following conditions: $T'_{triplet} = \{D \cup E \cup F\}$, with:

$$\begin{aligned} D &= \{\, x \mid T(x,c_1') \,\} \\ E &= \{\, x \mid T(x,c_2') \,\} \\ F &= \{\, x \mid T(x,c_3') \wedge \exists y(T(y,c_4') \wedge R(x,y,r') \,\} \end{aligned} \tag{3}$$

$$\Downarrow$$

$$\begin{aligned} D &= \{\, x \mid T(x, RefereedJournal \,\} \\ E &= \{\, x \mid T(x, MedicalRefereedJournal) \,\} \\ F &= \{\, x \mid T(x, Publication) \wedge \exists y(T(y, ReviewPolicy) \wedge R(x,y,governedBy) \end{aligned}$$

4.2.2 Scenario 2: Combinations of SubclassOf, SubPropertyOf, and Complex (c: c) Correspondences

In this category, we apply a rewriting strategy based on the combination of the subClassOf and subPropertyOf relations, leveraging the semantic hierarchies defined between the entities of the source and target ontologies.

```
Source entity : PeerReviewedJournal == RefereedJournal ; Journal & usesProcess.value('PeerReview')
├── sub-entities (within the same ontology) :
│   - EngineeringPeerReviewedJournal
├── sub-entities in the target ontology (inferred as sub-entities of this source entity) :
│   - RefereedJournal = Publication & governedBy.some(ReviewPolicy) ; PeerReviewedJournal
│   - MedicalRefereedJournal
```

Figure 2: Combinations of subClassOf relations with complex (c: c) correspondences.

```
Please formulate your request  : Can you list all journals that are associated with the peer-review process?
Onto source = ['PeerReviewedJournal', "Journal,usesProcess,value'PeerReview'", 'EngineeringPeerReviewedJournal']
Onto cible = ['MedicalRefereedJournal', 'RefereedJournal', 'Publication,governedBy,ReviewPolicy']
Appropriate SPARQL query on the source side                  | Automatically generated query on the target side
------------------------------------------------------------ | ------------------------------------------------
SELECT ?variable                                             | SELECT ?variable
WHERE {                                                      | WHERE {
  {                                                          |   {
  ?variable rdf:type source :PeerReviewedJournal .           |   ?variable rdf:type target :MedicalRefereedJournal .
  }                                                          |   }
  UNION                                                      |   UNION
  {                                                          |   {
  ?variable rdf:type source :EngineeringPeerReviewedJournal . |   ?variable rdf:type target :RefereedJournal .
  }                                                          |   }
  UNION                                                      |   UNION
  {                                                          |   {
  ?variable rdf:type source :Journal .                       |   ?variable rdf:type target :Publication .
  ?variable source:usesProcess value'PeerReview' .           |   ?variable target:governedBy ?ReviewPolicy .
  }                                                          |   }
}                                                            | }
```

Figure 3: Query rewriting with combinations of subClassOf relations.

Let there be two aligned ontologies, with:

- Source ontology: Let c_1 *and* c_2 be two classes, ***r*** an object property, and ***p*** a data property.
- Target ontology: Let c_1' *and* c_2' be two classes, $\boldsymbol{r'}$ an object property, and p' a data property.

And the following relationships between the entities of the two ontologies:

- $c_1 \equiv c_1'$, $c_2' \sqsubseteq c_2$
- $r' \sqsubseteq r$, with $domain(r) = c_1$ *et* $range(r) = c_2$; $domain(r') = c_1'$ *et* $range(r') = c_2'$
- $p' \sqsubseteq p$, $domain(p) = c_1$; $domain(p') = c_1'$

Thus, by leveraging subsumption relationships over data properties, object properties, and classes, our approach enabled us to infer the following correspondences:

- For relationships between classes and object properties, our approach allowed us to identify two main categories:

  - Subsumption with existential quantifiers ($\exists$)

$$\{ x \mid T(x, c_1') \land \exists y (T(y, c_2') \land R(x, y, r'))\} \sqsubseteq \{ x \mid T(x, c_1) \land \exists y (T(y, c_2) \land R(x, y, r)) \} \quad (4)$$

  - Subsumption with universal quantifiers ($\forall$)

$$\{ x \mid T(x, c_1') \land \forall y (T(y, c_2') \Rightarrow R(x, y, r'))\} \sqsubseteq \{ x \mid T(x, c_1) \land \forall y (T(y, c_2) \Rightarrow R(x, y, r))\} \quad (5)$$

- For relationships between classes and data properties, the following category was identified through our approach:

$$\{ x \mid T(x, c_1') \land P(x, p', v)\} \sqsubseteq \{ x \mid T(x, c_1) \land P(x, p, v)\} \quad (6)$$

These relations show that, if the subsumption relationships are correctly exploited, a query formulated on the target ontology can be considered a valid specialization of an equivalent query on the source ontology. Figure 4 illustrates the result deduced from the logic formalized in Equation (4).

The evaluation of this category was based on the following natural language questions:

- *I require the set of individuals of type Person who are associated with at least one postal address.*
- *I would like to obtain the list of people who have at least one postal address.*

4.2.3 Scenario 3: Transitive Propagation of Subsumptions across Multiple Levels

The idea developed in this category is illustrated in Figure 1, which presents the complete ontological hierarchy of the superclass *Event*, including entities inferred within the target ontology. Although entities such as *Conference*, *pc_meeting,* and *session*, which are specific to the target ontology, are explicitly defined as direct subclasses of *Working_event* in the source ontology, they are inferred as specializations of Event through transitive propagation of subsumption relations, in accordance with ontological inference mechanisms.

Figure 5 illustrates the results generated from the subsumption relations identified between the source and target ontologies. In the absence of complex triples, the results correspond to the union of instances of atomic classes from both ontologies, aggregated in the following generalized form.

```
Please formulate your request  : I require the set of individuals of type Person who are associated with at least one postal address.
Onto source = ['person,HasPostalAdress,PostalAdress']
Onto cible = ['Person,hasCity,City']
Appropriate SPARQL query on the source side              | Automatically generated query on the target side
-------------------------------------------------------- | -------------------------------------------------------
SELECT ?variable                                         | SELECT ?variable
WHERE {                                                  | WHERE {
  {                                                      |   {
  ?variable rdf:type source :person .                    |   ?variable rdf:type target :Person .
  ?variable source:HasPostalAdress ?PostalAdress .       |   ?variable target:hasCity ?City .
  }                                                      |   }
}                                                        | }
```

Figure 4: Rewriting results involving combinations of subClassOf, subPropertyOf.

```
Onto source = ['Reviewing_results_event', 'Tutorial', 'Working_event', 'Reviewing_event', 'Conference', 'Trip
Onto cible = ['pc_meeting', 'Conference', 'session']
Appropriate SPARQL query on the source side              | Automatically generated query on the target side
-------------------------------------------------------- | ----------------------------------------------
SELECT ?variable                                         | SELECT ?variable
WHERE {                                                  | WHERE {
  {                                                      |   {
  ?variable rdf:type source :Reviewing_results_event .   |   ?variable rdf:type target :pc_meeting .
  }                                                      |   }
  UNION                                                  |   UNION
  {                                                      |   {
  ?variable rdf:type source :Tutorial .                  |   ?variable rdf:type target :Conference .
  }                                                      |   }
  UNION                                                  |   UNION
  {                                                      |   {
  ?variable rdf:type source :Working_event .             |   ?variable rdf:type target :session .
  }                                                      |   }
  UNION                                                  | }
  {                                                      |
  ?variable rdf:type source :Reviewing_event .           |
  }                                                      |
  UNION                                                  |
  {                                                      |
```


Figure 5: Rewriting result involving transitive propagation of subsumption relations over multiple levels.

- **Source Ontology:**

$$T_{triplet} = \{ x \mid T(x, c_1) \vee T(x, c_2) \vee T(x, c_3) \vee \ldots \vee T(x, c_n)\} \quad (7)$$

- **Target Ontology:**

$$T'_{triplet} = \{ x \mid T(x, c'_1) \vee T(x, c'_2) \vee T(x, c'_3) \vee \ldots \vee T(x, c'_n)\} \quad (8)$$

Here, n denotes the number of entities (Classes) involved in the rewriting process.

4.2.4 Scenario 4: Specific Sublevel

Relying on the logic represented in Figure 1, a query issued to the entity *Working_event* automatically triggers the retrieval of its explicitly defined subclasses in the target ontology, through ontological inference mechanisms consistent with the established subsumption relations. Figure 6 presents the class *Working_Event* from the source ontology along with its subclasses explicitly defined in the target ontology.

```
Source entity : Working_event
├── sub-entities (within the same ontology) :
│   - Conference
│   - Tutorial
│   - Workshop
├── sub-entities in the target ontology (infe
│   - Conference = Conference
│   - pc_meeting
│   - session
```

Figure 6: Rewriting of a Specific Sublevel.

```
Please formulate your request  : What are the different types of working events ?
Onto source = ['Working_event', 'Tutorial', 'Workshop', 'Conference']
Onto cible = ['pc_meeting', 'session', 'Conference']
Appropriate SPARQL query on the source side           | Automatically generated query on the target side
------------------------------------------------------ | ----------------------------------------------
SELECT ?variable                                       | SELECT ?variable
WHERE {                                                | WHERE {
  {                                                    |   {
  ?variable rdf:type source :Working_event .           |   ?variable rdf:type target :pc_meeting .
  }                                                    |   }
  UNION                                                |   UNION
  {                                                    |   {
  ?variable rdf:type source :Tutorial .                |   ?variable rdf:type target :session .
  }                                                    |   }
  UNION                                                |   UNION
  {                                                    |   {
  ?variable rdf:type source :Workshop .                |   ?variable rdf:type target :Conference .
  }                                                    |   }
  UNION                                                | }
  {                                                    |
  ?variable rdf:type source :Conference .              |
  }                                                    |
}                                                      |
```

Figure 7: result of the subsumption relations of a specific sublevel.

In this category, the SPARQL query on the source ontology consists of selecting triples that satisfy the conditions specified in the *WHERE* clause, if any conditions are present. As illustrated in Figure 7, the *WHERE* clause of the source query must evaluate a set of criteria:

$$T_{triplet} = \{\, x \mid T(x, c_1) \vee T(x, c_2) \vee T(x, c_3) \vee T(x, c_4)\} \quad (9)$$
$$\Downarrow$$
$$T_{triplet} = \{\, x \mid T(x, Working\_event) \vee T(x, Conference) \vee T(x, Tutorial) \vee T(x, Workshop)\}$$

While the target query is required to satisfy the constraints imposed by the target ontology.

$$T'_{triplet} = \{\, x \mid T(x, c'_1) \vee T(x, c'_2) \vee T(x, c'_3)\} \quad (10)$$
$$\Downarrow$$
$$T'_{triplet} = \{\, x \mid T(x, Conference) \vee T(x, pc_{meeting}) \vee T(x, session)\}$$

The application of our approach to the example questions below yields the results shown in Figure 7. Tested questions were :

- *What events are part of the working event group?*
- *What are the different types of working events?*

5 DISCUSSION

As previously mentioned, most existing approaches focus on simple or partially complex correspondences, while those addressing complex correspondences rely primarily on equivalence relations. By taking into account the majority of subsumption relations, our method represents a significant advancement over these previous methods by broadening the range of relations utilized during rewriting our method is characterized by strong modularity, enabling easy adaptation to different types of ontologies and alignment configurations. It thus aims to enrich the state of the art, responding to the limited support for subsumption relations in automatic rewriting processes, particularly when involving complex correspondences (c: c) in SPARQL query rewriting.

In this study, we consider that we have significantly achieved the objectives we initially set, by demonstrating the feasibility and effectiveness of our method to SPARQL query rewriting in the context of complex ontology alignment involving subsumption relations. Nevertheless, certain challenges remain to be addressed, notably the systematization and full automation of handling these relations, especially in contexts of very large-scale aligned ontologies. Another promising avenue for future research lies in the systematic exploration of various forms of correspondences involving cardinality restrictions, particularly those that leverage minimum, maximum, and exact cardinalities.

REFERENCES

Amini, R., Norouzi, S.S., Hitzler, P., & Amini, R. (2024). Towards Complex Ontology Alignment using Large Language Models. *Iberoamerican Conference on Knowledge Graphs and Semantic Web.*

Baader, F., Calvanese, D., McGuinness, D., Nardi, D., et Patel-Schneider, PF (dir.). (2003). *Manuel de logique de description: théorie, implémentation et applications*. Cambridge University Press.

Correndo, G., Salvadores, M., Millard, I., Glaser, H., & Shadbolt, N. (2010, March). SPARQL query rewriting for implementing data integration over linked data. In Proceedings of the 2010 EDBT/ICDT Workshops (pp. 1-11).

David, J., Euzenat, J., Scharffe, F., & Trojahn dos Santos, C. (2011). The alignment API 4.0. *Semantic web*, *2*(1), 3-10.

Fujino, T., & Fukuta, N. (2012, September). A SPARQL query rewriting approach on heterogeneous ontologies with mapping reliability. In *2012 IIAI International Conference on Advanced Applied Informatics* (pp. 230-235). IEEE.

Lamy, J.-B. (2020). Ontologies with Python: Programming OWL 2.0 Ontologies with Python and Owlready2. Publisher Apress Berkeley, CA. ISBN 978-1- 4842-6551-2. https://doi.org/10.1007/978-1-4842-6552-9

Ondo, A.L., Capus, L., & Bousso, M**.** (2025). Enhancing SPARQL Query Rewriting for Complex Ontology Alignments. *International journal of Web & Semantic Technology, Vol.16 (2)*. DOI: 10.5121/ijwest. 2025.16201

Shvaiko P., M. Abd Nikooie Pour, A. Algergawy, P. Buche, L. J. Castro, J. Chen, A. Coulet, J. Cu , H. Dong, O. Fallatah, D. Faria, I. Fundulaki, S. Hertling, Y. He, I. Horrocks, M. Huschka, L. Ibanescu, S. Jain, E. Jim´enez-Ruiz, N. Karam, P. Lambrix, H. Li, Y. Li, P. Monnin, E. Nasr, H. Paulheim, C. Pesquita, T. Saveta, G. Sousa, C. Trojahn, J. Vatascinova, M. Wu, B. Yaman, O. Zamazal, L. Zhou,(2023). Results of the ontology alignment evaluation initiative 2023, in: P. Shvaiko, J. Euzenat, E. Jim´ enezRuiz, O. Hassanzadeh, C. Trojahn (Eds.), Proceedings of the 18th International Workshop on Ontology Matching (OM 2023) co-located with the 22nd International Semantic Web Conference (ISWC 2023), Athens, Greece, November 7, 2023, volume 3591 of CEUR Workshop Proceedings, CEUR-WS.org, 2023, pp. 97–139. URL: https://ceur-ws.org/Vol-3591/oaei23 paper0.pdf

Thakker D, Schwabe D, Kozaki K, et al. Estimating query rewriting quality over LOD. *Semantic Web*. 2018;10(3):529-554. doi:10.3233/SW-180311

Thiéblin, É., Amarger, F., Haemmerlé, O., Hernandez, N., & Trojahn, C. (2016). Rewriting SELECT SPARQL queries from 1: n complex correspondences. CEUR-WS: Workshop proceedings.

Thiéblin, E., Amarger, F., Hernandez, N., Roussey, C., & Trojahn Dos Santos, C. (2017). Cross-querying lod datasets using complex alignments: an application to agronomic taxa. In *Metadata and Semantic Research: 11th International Conference, MTSR 2017, Tallinn, Estonia, November 28– December 1, 2017, Proceedings 11* (pp. 25-37). Springer International Publishing.

Thiéblin, E., Haemmerlé, O., & Trojahn, C. (2021). Automatic evaluation of complex alignments: An instance-based approach. *Semantic Web*, *12*(5), 767-787.

Thiéblin, Haemmerlé, Jane Hernandez, Trojahn dos Santos (2018). Task-Oriented Complex Ontology Alignment: Two Alignment Evaluation Sets. European Semantic Web Conference (ESWC 2018), Jun 2018, Heraklion, Greece. pp.655–670, ff10.1007/978-3-319-93417-4_42ff.ffhal-02102383f

Trojahn C., M. Abd Nikooie Pour, A. Algergawy, F. Amardeilh, R. Amini, O. Fallatah, D. Faria, I. Fundulaki, I. Harrow, S. Hertling, P. Hitzler, M. Huschka, L. Ibanescu, E. Jim´ enez-Ruiz, N. Karam, A. Laadhar, P. Lambrix, H. Li, Y. Li, F. Michel, E. Nasr, H. Paulheim, C. Pesquita, J. Portisch, C. Roussey, T. Saveta, P. Shvaiko, A. Splendiani, J. Vatascinov´a, B. Yaman, O. Zamazal, L. Zhou, (2021). Results of the ontology alignment evaluation initiative 2021, in: P. Shvaiko, J. Euzenat, E. Jim´enez-Ruiz, O. Hassanzadeh, C. Trojahn (Eds.), Proceedings of the 16th International Workshop on Ontology Matching co-located with the 20th International Semantic Web Conference (ISWC 2021), Virtual conference, October 25, 2021, volume 3063 of CEUR Workshop Proceedings, CEUR-WS.org, 2021, pp. 62–108. URL: http://ceur-ws.org/Vol-3063/oaei21 paper0.pdf.