Preserving the Original Query Semantics in Routing Processes

Crishane Freire[1], Nicolle Cysneiros[2], Damires Souza[1] and Ana Carolina Salgado[2]
[1]*Federal Institute of Education, Science and Technology of Paraiba, João Pessoa, PB, Brazil*
[2]*Federal University of Pernambuco, Recife, PE, Brazil*

Keywords: Query Semantics, Query Reformulation, Query Routing, Query Semantics Preservation.

Abstract: In distributed data environments, peers (data sources) are connected with each other through a set of semantic correspondences in such a way that peers directly connected are called semantic neighbours. Queries are submitted considering partial information provided by a peer schema and may be answered by other neighbour peers. From the query submission peer, the original query is successively rewritten into queries over the peers, according to the correspondences between the original peer and the target ones. In this process, some of the original query terms may be lost while other ones may be added, leading to a semantic loss of the original query. In this work, we argue that it is essential to try preserving the original query semantics if we wish to hold what the users defined as important at query submission time. With this in mind, we propose an approach to preserve the original query semantics in query routing processes. Furthermore, we present a metric for assessing the relevance of neighbour peers according to an estimated query semantic value obtained at each query reformulation. In this paper, we present the developed approach and some experimental results we have accomplished.

1 INTRODUCTION

Query answering in distributed data environments faces some challenges related mainly to the large number of data sources, their autonomous nature, and the heterogeneity of their data. These environments have a diversity of perspectives and are composed by data sources (*peers*), which are linked by means of semantic mappings (here called *correspondences*). Peers directly connected by correspondences are called semantic neighbours.

One special problem concerning these architectures is how to exploit the correspondences between neighbour peers in order to answer queries. This problem, named as *query routing process*, regards identifying relevant neighbour peers for answering particular user queries, in such a way that answers may fit better the user needs. Particularly, one key issue is how to preserve the semantics of a submitted query as long as it is routed through the set of neighbour peers (Delveroudis et al. 2009; Kantere et al. 2009).

In fact, query routing processes and query reformulation strategies have a great influence on each other. We argue that, in query routing processes, the original query semantics should be preserved as far as possible. By the "query semantics", we mean the set of terms (i.e., concepts and properties), which are required at query formulation time. To this end, two aspects should be considered, namely: (i) query reformulation along the peers should take into account what users defined as important in the original submitted query; and (ii) the query routing process should avoid forwarding a reformulated query which has been identified with a high degree of semantic loss (Delveroudis and Lekeas, 2007).

With this in mind, in this work, we propose an approach to preserve, as far as possible, the original query semantics in query routing processes. To this end, we define a query semantic reference, which is used to prune the query semantic loss. Furthermore, we present a metric for assessing the relevance of neighbour peers according to an estimated query semantic value obtained at each query reformulation.

Our contributions are summarized as follows:

- We propose an approach to preserve the original query semantics in query routing processes taking into account a semantic reference which is defined on the fly.
- We create a query semantic reference according to a domain ontology composed by a set of expanded terms.
- We define a metric to assess the degree of query

Freire C., Cysneiros N., Souza D. and Salgado A..
Preserving the Original Query Semantics in Routing Processes.
DOI: 10.5220/0004893100700080
In *Proceedings of the 16th International Conference on Enterprise Information Systems* (ICEIS-2014), pages 70-80
ISBN: 978-989-758-027-7

semantics preservation at query reformulation time.
- We describe experiments regarding the effectiveness of our approach.

This paper is organized as follows: Section 2 defines our setting; Section 3 introduces a motivating example. In Section 4, we propose our approach to preserve the original query semantics in query routing processes. Section 5 shows some experiments we have accomplished. Section 6 discusses some related work. Finally, Section 7 draws our conclusions and points out some future work.

2 OUR SETTING

We have instantiated our approach in a Peer Data Management System (PDMS) (Roth and Skritek, 2013), although it can be instantiated in any dynamic distributed data environment. PDMS consist of a set of peers, each one with an associated schema that represents the data to be shared with other peers. In such systems, schema matching techniques are used to establish correspondences between peer schemas, which form the basis for query reformulation.

Our system is called SPEED, a PDMS that adopts an ontology-based approach to assist relevant issues in peer data management (Pires, 2009). In SPEED, the peers are clustered according to the same knowledge domain (e.g., *Education*, *Health*), and an ontology describing the domain is available to be used as background knowledge. Ontologies are also used as a uniform conceptual representation of peer schemas. Peers are semantically related by means of correspondences. We consider that correspondences are defined between pairs of semantic neighbour peers, i.e., peers that are semantically related as identified by a clustering process. In this scenario, in a query routing process, queries are submitted at one peer and reformulated to other peers, and the final query result is the union of the query answers (considering that data conversions have already been done) returned by each accessed peer.

In this work, particularly, we consider two aspects: (i) the ontology matcher that generates the semantic correspondences between peer schemas; and (ii) the query reformulation process itself. In the following subsections, we describe a running scenario, the semantic matcher (Pires et al. 2009) and the query reformulation process (Souza et al. 2011) used in our work.[1]

2.1 Running Scenario

In this scenario, we use ontologies belonging to the Education's knowledge domain. The peers have complementary data about academic people and their work (e.g., publications). We assume that peer ontologies have been normalized into a uniform representation format according to some background knowledge (in our work, we use a domain ontology). Throughout the paper, we will use a domain ontology concerning Education, named UnivCSCMO.owl[1] (Figure 1).

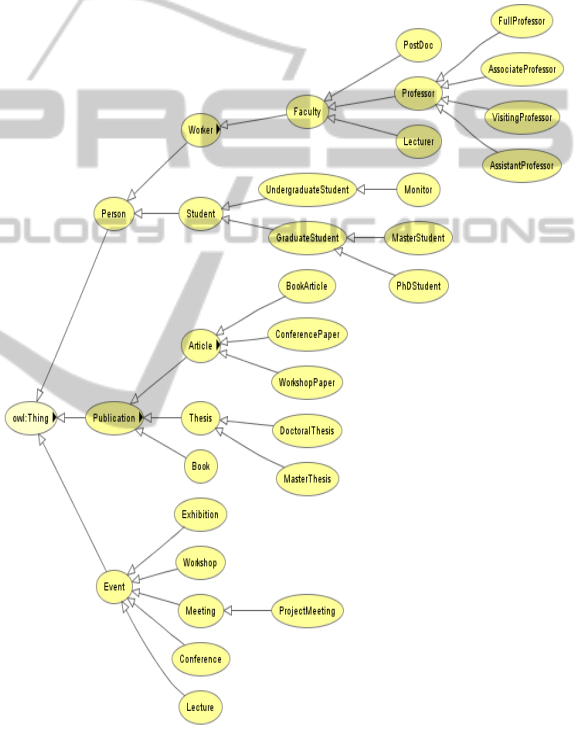

Figure 1: An excerpt from the domain ontology.

2.2 Semantic Matcher

The output of a matching process is called an alignment, which contains a set of correspondences indicating which terms (concepts or properties) of the two ontologies logically correspond to each other (Euzenat and Shvaiko 2007).

The ontology semantic matcher (*SemMatcher*) (Pires et al. 2009) considers domain ontologies (DO) as reliable references, which are usually made available on the Web. It uses them in order to bridge

[1]http://www.cin.ufpe.br/~speed/SemMatch/UnivCsCMO.owl

the conceptual differences or similarities between two peer schemas. In this process, a linguistic-structural matcher and a semantic matcher are executed in parallel. The obtained similarity values of both matchers are combined through a weighted average.

The *SemMatcher* approach identifies seven kinds of semantic correspondences (Souza et al. 2009), each one associated with a particular weight which corresponds to the level of confidence on such correspondence as follows: *isEquivalentTo (1.0), isSubConceptOf (0.8), isSuperConceptOf (0.8), isCloseTo (0.7), isPartOf (0.5), isWholeOf (0.5), and isDisjointWith (0.0)*. The weights reflect the degree of similarity between the correspondent elements, from the strongest relationship (*equivalence*) to the weakest one (*disjointness*).

Example 1: To better illustrate, consider a scenario composed by two peers P_1 and P_2. Each peer is described by an ontology: O_1 and O_2, respectively. In order to identify the semantic correspondences between O_1 and O_2, we used the *SemMatcher* tool. As a result, a set of correspondences (i.e., an alignment) between O_1 and O_2 and their respective weights were identified. Figure 2 depicts a fragment of the alignment file. In this illustrative set, we can see the *isEquivalentTo* correspondence between the term FullProfessor in O_1 and O_2. By using the semantics underlying the DO, the *SemMatcher* also identifies other types of correspondences such as the *isCloseTo* correspondence (i.e, terms with a semantic similarity degree, which is achieved from the identification of sibling concepts in the domain ontology) between *Lecturer* in O_1 and *Professor* in O_2.

Ontology 1	Correspondence	Ontology 2	Measure
Faculty.worksAtProject	isEquivalentTo	Faculty.worksAtProj...	1
FullProfessor	isEquivalentTo	FullProfessor	1
FullProfessor	isSubConceptOf	Professor	0,8
FullProfessor	isPartOf	Course	0,3
FullProfessor	isDisjointWith	AssociateProfessor	0
FullProfessor	isDisjointWith	AssistantProfessor	0
FullProfessor.partOf	isEquivalentTo	ObjectUnionOf(Assis...	1
GraduateCourse	isEquivalentTo	GraduateCourse	1
GraduateCourse	isSubConceptOf	Course	0,8
Lecturer	isSubConceptOf	Faculty	0,8
Lecturer	isCloseTo	Professor	0,7
Manual	isEquivalentTo	Manual	1

Figure 2: Fragment of an alignment.

2.3 Query Reformulation

In a query reformulation process, terms from a source peer do not always have exact corresponding terms in a target one, what may result in an empty set of reformulations and, possibly, no answer to users. Query reformulation strategies by terms expansion (Carpineto and Romano, 2012; Souza et al., 2009) aim to provide users with not only exact answers but also approximate, additional (i.e., expanded) answers.

The *SemRef* approach (Souza et al., 2009) uses the set of semantic correspondences to produce two kinds of query reformulations: (i) an *exact* one, considering only equivalence correspondences and (ii) an *enriched* one, resulting from the set of the other correspondences identified by the *SemMatcher*. To define the query enriching mode, the user may set four variables, which specify what should be considered when a query Q is to be enriched. The variables are defined as follows:

- *Approximate*: includes terms that are close to the ones of Q;
- *Specialize*: includes terms that are sub-concepts of some terms of Q;
- *Generalize*: includes terms that are super-concepts of some terms of Q;
- *Compose*: includes terms that are part-of or whole-of some terms of Q.

The queries can be formulated by considering the terms provided by the peer ontology, using SPARQL[2] or ALC/DL (Baader et al., 2003). For the sake of simplicity, in this paper, we assume that Q is a query expressed over a peer ontology, which has the following form: $Q = \{T_i\}$, $i = 1 \ldots n$, where T_i is a term (concept or property) that belongs to the peer ontology.

Considering the scenario described in Example 1, suppose that a user submits a query *Q = {FullProfessor}* in P_1 and set all four enriching variables to TRUE. The *SemRef* tool produces two query reformulations: an exact reformulation, denoted by *Qexact = {FullProfessor}*, and another one, enriched, denoted by *Qenriched={Professor, Course}*.

In the next section, we present a motivating example to justify the proposal of our approach.

3 MOTIVATING EXAMPLE

Our motivating example is based on the running scenario described in Section 2.1. It is composed by 3 peers (Figure 3), which are linked to other peers by means of correspondences, as defined in Section 2.1.

As depicted in Figure 3, each box represents a peer ontology. In this scenario, suppose that a user submits a query *Q={GraduateStudent}* at peer P1

[2]http://www.w3.org/TR/rdf-sparql-query/

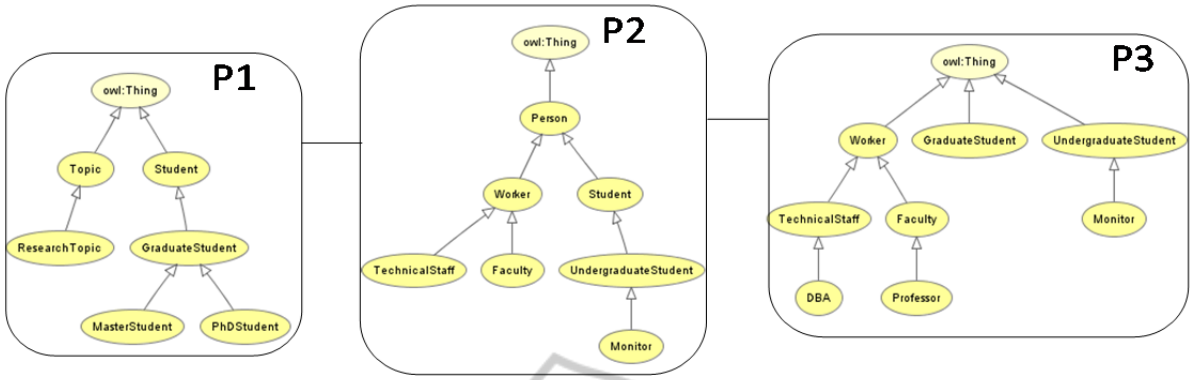

Figure 3: Motivating Example.

and, in order to acquire more related results, s/he sets the *Specialize* and *Generalize* variables to TRUE. At P1, query Q is expanded to *Q={GraduateStudent, Student, PhDStudent, MasterStudent}* taking into account the enriching correspondences. Then Q is executed.

In order to be forwarded to P2, Q is reformulated into Q_{12}*={UndergraduateStudent, Student, Person}* according to the local ontology of P2. To clarify matters, Figure 4 presents a fragment of the alignment between P1 (O1) and P2 (O2).

Ontology 1	Correspondence	Ontology 2	Measure
GraduateStudent	isDisjointWith	UndergraduateStudent	0
GraduateStudent	isSubConceptOf	Student	0,8
Student	isCloseTo	Worker	0,7
Student	isSuperConceptOf	UndergraduateStudent	0,8
Student	isEquivalentTo	Student	1
Student	isSubConceptOf	Person	0,8
Topic	isDisjointWith	Person	0

Figure 4: Fragment of the Alignment between P1 and P2.

When reformulating Q_{11} to Q_{12}, some terms *(GraduateStudent, PhDStudent, MasterStudent)* are lost. On the other hand, Student is preserved and the terms *Person* and *UndergraduateStudent* are included. Despite the term *UndergraduateStudent* is related to the term *Student*, it doesn't have a semantic association with the term *GraduateStudent* of the original query. Indeed, they are disjoint terms as we can verify in the alignment (Figure 4).

After Q_{12} has been executed locally, it is reformulated according to the correspondences between P2 and P3 (Figure 5). This results in query Q_{13} *= {GraduateStudent, UndergraduateStudent, Monitor, Worker}*. Although the original term *GraduateStudent* is present in Q_{13}, the other included terms are producing a semantic loss with respect to the **original query** that was looking for *GraduateStudent*. In this sense, it is likely that the query results will not be suitable to the user's interests.

It is important to emphasize that there is no semantic loss with respect to the **current query** (the one that reaches the peer). The reformulated query terms *UndergraduateStudent*, *Monitor* and *Worker* have a semantic correspondence with the current query terms *Student* and *Person* (Figure 5).

Ontology 1	Correspondence	Ontology 2	Measure
Monitor	isSubConceptOf	UndergraduateStudent	0,8
Person	isSuperConceptOf	Worker	0,8
Student	isSuperConceptOf	GraduateStudent	0,8
Student	isSuperConceptOf	UndergraduateStudent	0,8
Student	isCloseTo	Worker	0,7
TechnicalStaff	isEquivalentTo	TechnicalStaff	1
TechnicalStaff	isSubConceptOf	Worker	0,8
TechnicalStaff	isSuperConceptOf	DBA	0,8
TechnicalStaff	isCloseTo	Faculty	0,7
UndergraduateStudent	isEquivalentTo	UndergraduateStudent	1
UndergraduateStudent	isSuperConceptOf	Monitor	0,8
UndergraduateStudent	isDisjointWith	GraduateStudent	0
UndergraduateStudent	isDisjointWith	Worker	0
Worker	isEquivalentTo	Worker	1

Figure 5: Fragment of the Alignment between P2 and P3.

4 OUR APPROACH

Although query reformulation strategies by means of query expansion have been used in order to improve queries recall, it sometimes results in a precision loss (Campos et al., 2013). Close terms which are added to reformulated queries may produce a *gap* between the original query and their expanded forms.

Considering that we are dealing with a large-scaled dynamic network, composed by thousands of peers, the successive processes of query reformulation may lead to a high degree of query semantics loss. In our setting, this may occur due to two possible reformulation actions: (i) by *losing* (i.e., removing) query original terms, when there is

no equivalence correspondence between these terms and the target peer ontology ones; and/or (ii) by *including* new close terms acquired at the query enrichment step.

In order to preserve the original query semantics, we propose the creation of a semantic reference. This semantic reference is defined with respect to the original submitted query according to a background knowledge provided by a domain ontology. Besides, we define some metrics to assess the query semantic value.

These metrics are based on preservation and enrichment measures, which are calculated at query reformulation time. In the following sections, we define the semantic reference and the metrics used to assess the query semantic value.

4.1 Defining a Semantic Reference

The semantic reference is used for pruning (i.e. removing) terms of the reformulated query that are producing a semantic loss with respect to the original query. In order to define the semantic reference, we state, at first, the direct semantic relation, as follows.

Definition 1 (Direct Semantic Relation): Let O_1 and O_2 be two neighbour peer ontologies. We state that a term (concept or property) T_1 in O_1 has a *Direct Semantic Relation (DSR)* with T_2 in O_2, if there is a defined relationship between them, by means of a correspondence *Co*, where *Co* ∈ {*isEquivalentTo, isSubConceptOf, isSuperConceptOf, isCloseTo, isPartOf, isWholeOf, isDisjointWith*}.

Definition 2 (Semantic Reference): Let *Q* be a query submitted at peer P_i, in accordance with its local ontology O_i, and *DO* a corresponding domain ontology. A *Semantic Reference* (SR) is defined as a triple SR = {<t_i, t_j, *w*>}, where t_i is a query term ϵ O_i, t_j is a term ϵ DO and t_i has a DSR with t_j; *w* is the weight associated to each existing correspondence between t_i and t_j.

To create the SR, firstly, an alignment between the submission peer ontology and the domain ontology is generated. Then, by considering the enriching variables, the original terms are expanded and the weights associated to the semantic correspondences between the original term and its expanded terms are identified. Thus, the SR is a set of triples composed by the original term, the expanded term and its correspondence weight.

By maintaining the correspondence weights in the SR allows choosing the semantically closest correspondences (according to their weights) from the existing ones. This is especially important when there are reformulated query terms belonging to the SR that are not expanded by some original correspondent term.

Example 2: To clarify matters, consider the query *Q* = *{FullProfessor}*, shown in Example 1. Also, consider a fragment of an alignment between O_1 and the DO, as depicted in Figure 6.

Ontology 1	Correspondence	Ontology 2	Measure
Faculty.teacherOf	isEquivalentTo	Faculty.teacherOf	1
Faculty.worksAtP...	isEquivalentTo	Faculty.worksAtP...	1
FullProfessor	isEquivalentTo	FullProfessor	1
FullProfessor	isSubConceptOf	Professor	0,8
FullProfessor	isPartOf	ResearchProject	0,3
FullProfessor	isPartOf	Course	0,3
FullProfessor	isCloseTo	VisitingProfessor	0,7
FullProfessor	isDisjointWith	AssociateProfessor	0
FullProfessor	isDisjointWith	AssistantProfessor	0
FullProfessor.par...	isEquivalentTo	ObjectUnionOf(A...	1
GraduateCourse	isEquivalentTo	GraduateCourse	1
GraduateCourse	isSubConceptOf	Course	0,8

Figure 6: Fragment of the alignment between O_1 and the DO.

Considering that all four enriching variables have been set to TRUE, a semantic reference SR of Q was produced by taking into account the existing semantic correspondences, as follows:

SR=*{<FullProfessor,FullProfessor,1,0>,<FullProfessor,Professor,0,8>,<FullProfessor,ResearchProject,0,3>,<FullProfessor,Course,0,3>,<FullProfessor,VisitingProfessor,0,7>}*.

Both terms *AssociateProfessor* and *AssistantProfessor* are disjoint with *FullProfessor*, thus they are not included in the query reformulation. This is due to the fact that the disjoint correspondence is only used when there is a negation in the original query. In this way, SR is called a *semantic reference* of Q and will be used for pruning some terms in Q reformulations along a query routing process.

Thus, enriched terms which do not belong to the SR will indeed be pruned from the reformulated queries. The goal is to preserve the original query semantics and avoid possible semantic loss that may occur at query reformulation time.

4.2 Assessing the Query Semantic Value

When a query is reformulated to another peer schema, some of the original query terms may be lost due to the differences between peer schemas. On the other hand, the user may choose whether the query reformulation Q_r should consider more

semantic extra terms than the equivalent ones. In this way, during an overall query routing process, whenever Q_r reaches a target peer, it can be enriched with more terms and, consequently, may retrieve more results.

These loss and enrichment processes should be assessed in order to decide if the reformulated query is still useful for users (i.e., fits the original query semantics) and may be forwarded to the other peers.

In this light, suppose Q_{origin} is the original query submitted by the user, Q is a forwarded query from P_i to P_j, Qr is its reformulated query and SR a semantic reference of Q_{origin}. After pruning Qr, we may assess the semantic value of Qr by means of two metrics, defined as follows:

Query Preservation (QPreserv): this metric calculates the number of terms belonging to Q_{origin}, which have been preserved after query reformulation. The *QPreserv* measure is stated as follows:

$$\mathrm{QPreserv}_{\mathrm{Qr}} = \frac{\#\mathrm{t}_{\mathrm{eq}}}{\#\mathrm{t}_{\mathrm{origin}}} \quad (1)$$

where

t_{eq} is the number of terms in Q_r, which are equivalent to the original terms of Q_{origin};
t_{origin} is the number of terms in Q_{origin}.

Query Enrichment (QEnrich): this metric calculates the query enrichment value considering the terms that have been expanded according to the original terms. Thus, the *QEnrich* metric is stated as follows:

$$\mathrm{QEnrich}_{\mathrm{Qr}} = \frac{\sum_{k=1}^{nto}(\sum_{l=1}^{nte_k} w_{kl}/nte_k)}{nto} \quad (2)$$

where

nto, number of terms in Q_{origin};
nte_k, number of enriched terms for each term of Q_{origin};
w_{kl}, weight associated to the enriched term in the semantic reference.

The query semantic value (QSV), after the query reformulation, is given by the sum of the measures obtained from formulas (1) and (2). The QSV is properly stated in formula (3), as follows.

$$\mathrm{QSV}_{\mathrm{Qr}} = \mathrm{QPreserv}_{\mathrm{Qr}} + \mathrm{QEnrich}_{\mathrm{Qr}} \quad (3)$$

The QSV is established with values between 0 and 1.8, where 0 indicates no kind of relevancy, and 1.8 states the maximum relevancy for query reformulation. In this latter value, all the original terms have been preserved and the maximum enrichment has been acquired as well.

Since the query is reformulated by considering the target peer schema, we argue that the QSV may be used to assess the peer relevance with respect to that query. In this sense, such measure may be used as a selection criterion to query forwarding in a routing process. The next section presents some experimental results we have obtained.

5 EXPERIMENTS

Our experimental scenario is the one described in Section 2.1. In this setting, queries are submitted and reformulated along the network of neighbour peers.

In order to achieve more meaningful results, the query Q should be executed in the enriched mode, i.e., by considering all semantic correspondences (*isEquivalentTo, isSubConceptOf, isSuperConceptOf, isCloseTo, isPartOf, isWholeOf, isDisjointWith*). Taking into account the query semantics, Q should be routed only to the most relevant peers that can contribute with it.

The goals of our experiments are twofold: (i) to show how the original query semantics may be preserved along routing processes; and (ii) in which degree the query semantics is maintained. The former is measured by using the Jaccard coefficient (Han and Kamber, 2006). The latter is evaluated by considering the precision measure (Rijsbergen, 1979). To this end, we consider a set of peers, which are interconnected in a sequence, as follows: P2178 to P2278; P2278 to P2378; P2378 to P2478 and P2478 to P2578.

We have defined a QSV threshold to be used as a selection criterion to query forwarding. As the threshold value depends on the weights associated to the semantic correspondences in the matching process, we have carried out some tests. We have found out that the reformulated queries with QSV above 0.4 were closer to the original query semantics. Also, they have produced better query results in the majority of the tested settings. As a result, we have stated our QSV threshold as 0.4.

To evaluate our experiments we have used the PDMS SPEED. The SPEED interface provides the user with an ontology view that represents the query submission peer schema. Queries may be submitted by using SPARQL or DL languages and the enriching variables are defined by the user (Souza et al. 2009).

Given this scenario, we submitted a query *Q* asking for *GraduateStudent* and *Workshop* at P2178 and all the enriching variables were selected. Figure 7 depicts the query submission interface. After the query submission, a semantic reference SR of query

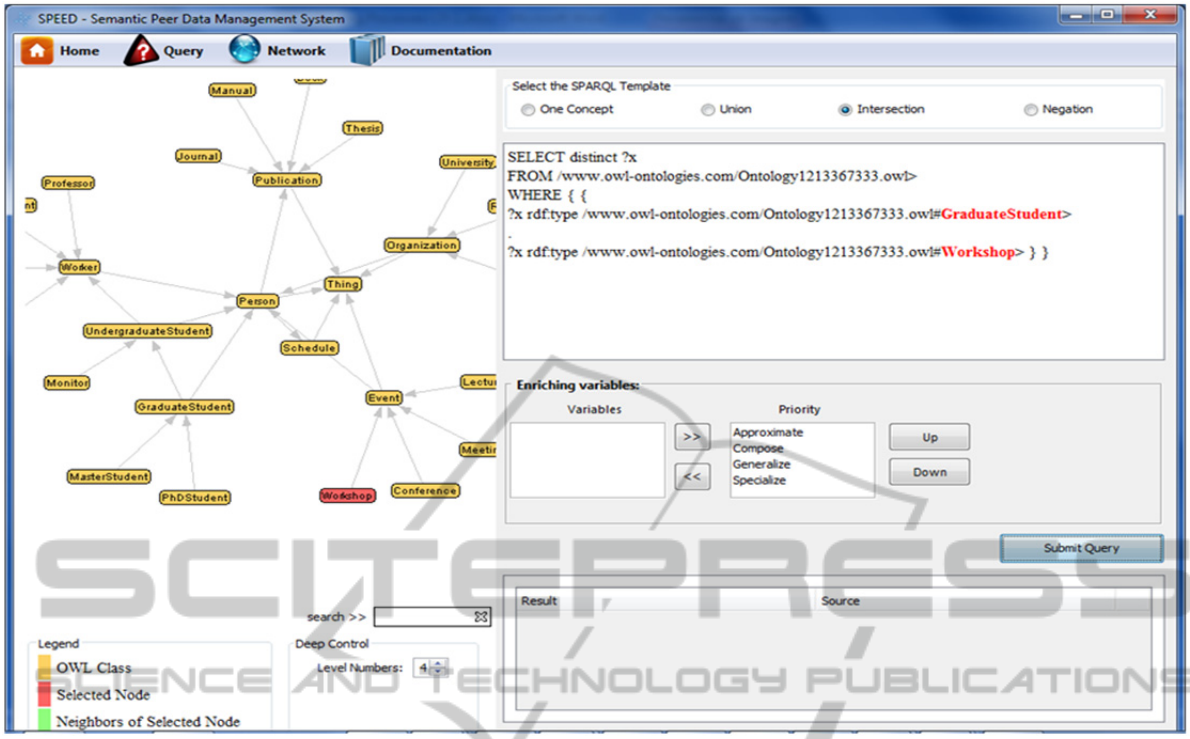

Figure 7: Query Submission Interface.

Q was dynamically produced. In this experiment, SR={*<GraduateStudent, GraduateStudent, 1.0>, <GraduateStudent, Student, 0.8>, <GraduateStudent, PhdStudent, 0.8>, <GraduateStudent, MasterStudent, 0.8>, <GraduateStudent, ResearchProject, 0.3>, <GraduateStudent, Course, 0.3>, <Workshop, Workshop, 1.0>, <Workshop, Event, 0.8>, <Workshop, Exhibtion, 0.7>, <Workshop, Conference, 0.7>, <Workshop, Meeting, 0.7>, <Workshop, Lecture, 0.7>}* was generated as the semantic reference of *Q*.

At each routing step, the obtained reformulated query was compared with SR. Thus, terms that may produce semantic loss in the query reformulation at hand were pruned.

At first, the query was expanded locally to *Q={GraduateStudent, MasterStudent, PhdStudent, Workshop, Event, Lecture, Meeting, Conference}* in order to find out extra similar terms at the submission peer P2178. Then, it was executed. Now, considering the peer neighbours, Q was reformulated in accordance with P2278 ontology.

Figure 8 presents the reformulation log between the peers P2178 and P2278. The semantic correspondence weight of each term is described in parentheses. The query that will be forwarded to the next peer is presented at the field *Reformulated Query After Pruning*. The terms without DSR with the query original terms are depicted at the field *Pruned Terms*. The fields *QPreserv, QEnrich* and *QSV* regard the measures that were assessed by using the metrics defined in Section 4.2.

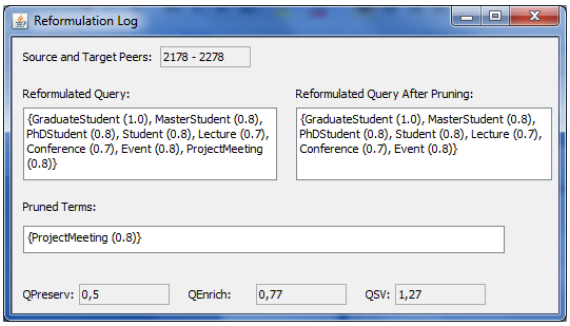

Figure 8: Reformulation log from P2178 to P2278.

All the expanded terms which were added to the reformulated query are in accordance with the original query semantics, i.e., only terms belonging to the SR were included. In addition, to assess the *QPreserv, QEnrich* and *QSV* values of the reformulated query after pruning, it is necessary to use the right weight of the semantic correspondence

associated to each term present in the SR.

Thus, using the metrics defined in Section 4.2, we assess the *QPreserv, QEnrich* and *QSV* as follows:

- Query Preservation:

$$QPreserv = \frac{1}{2} = 0{,}5$$

- Query Enrichment:
  Since the query enrichment measure is assessed by considering the original query terms, it is rather important to know which expanded terms have DSR (i.e., direct semantic relation) with these original query terms. Taking into account the SR, it is possible to identify that *MasterStudent*, *PhDStudent*, and *Student* have DSR with the original term *GraduateStudent*, and; *Lecture*, *Conference* and *Event* have DSR with the original term *Workshop*.

$$QEnrich = \frac{((0.8 + 0.8 + 0.8)/3) + ((0.7 + 0.7 + 0.8)/3)}{2}$$

$$QEnrich = 0.77$$

- Query Semantic Value:

$$QSV = 0.5 + 0.77 = 1.27$$

The *QSV* value is used to point out if the reformulated query is a suitable query that should be forwarded. After the reformulation step, the query is only routed to the next neighbour peer if the QSV is above the semantic threshold of 0.4.

From P2278 to P2378, the original term *GraduateStudent* was preserved and other terms were added by the query enrichment step (Figure 9).

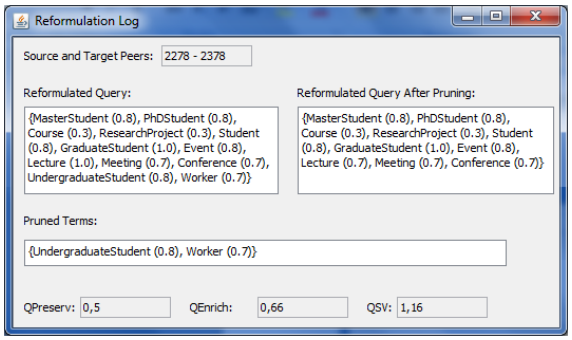

Figure 9: Reformulation log from P2278 to P2378.

In this reformulation example, the term *Lecture* present in the field *Reformulated Query* has a semantic correspondence weight equal to 1.0, because it was obtained from an equivalence correspondence. However this weight is not in accordance with the weight associated to *Lecture* in the Semantic Reference (SR). In SR, *Lecture* is a close term to the original term *Workshop,* and, for this reason, its value is 0.7.

To ensure the right assessment of the *QSV*, *QPreserv* and *QEnrich* values, the semantic correspondences weights are adjusted in accordance with the SR. In the field *Reformulated Query After Pruning*, the term *Lecture* appears with the new value 0.7 instead of 1.0. In addition, from P2378 to P2478 the original term *Workshop* was restored and *Lecture* was lost (Figure 10).

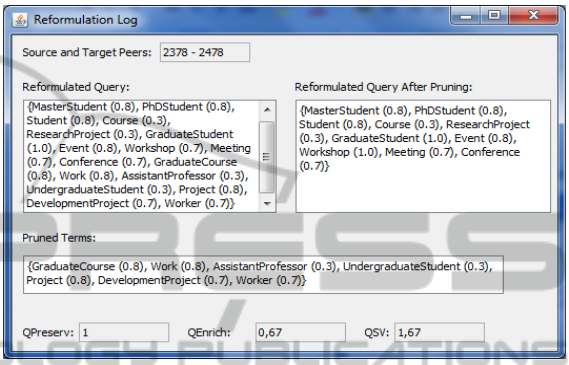

Figure 10: Reformulation log from P2378 to P2478.

After query pruning, some terms were eliminated and the semantic weights were adjusted. The term *Workshop* was restored by means of a closeness semantic correspondence, but in our experiments it is an equivalent term to an original one of Q. So, the weight associated to the term *Workshop* is 1.0 as we can see in the field *Reformulated Query After Pruning* (Figure 10).

Finally, from P2478 to P2578 the query is reformulated and forwarded to P2578 (Figure 11). Figure 12 summarizes the reformulated queries and their respective obtained measures in this query routing process illustration.

In this reformulation example, if no query original term had been found, the reformulated query would not be forwarded to P2578. *QPreserv, QEnrich* and *QSV* would have the values of 0, 0.3 and 0.3, respectively. In this case, the *QSV* value of 0.3 would be below the semantic threshold of 0.4. So, Q would not be forwarded to P2578.

5.1 Results Evaluation

In order to evaluate the experimental results, the query routing process was carried out in two different ways. In the former, $QR_{without\_Pruning}$, regards Q reformulations without taking into account if the expanded terms are semantically related to the terms of the original query, i.e., *without pruning*. In the

Source Peer - Target Peer	...	Reformulated Query	...	QPreserv	QEnrich	QVS
2178 - 2178	...	{PhDStudent (0.8), MasterStudent (0.8), GraduateStudent (1.0), Workshop (1.0), Event (0.8), Conference (0.7), Meeting (0.7), Lecture (0.7)}		1	0,76	1,76
2178 - 2278	...	{MasterStudent (0.8), PhDStudent (0.8), GraduateStudent (1.0), Student (0.8), Event (0.8), Lecture (0.7), Conference (0.7)}	...	0,5	0,77	1,27
2278 - 2378	...	{GraduateStudent (1.0), Student (0.8), Course (0.3), ResearchProject (0.3), MasterStudent (0.8), PhDStudent (0.8), Lecture (0.7), Event (0.8), Meeting (0.7), Conference (0.7)}	...	0,5	0,66	1,16
2378 - 2478	...	{GraduateStudent (1.0), MasterStudent (0.8), PhDStudent (0.8), Course (0.3), ResearchProject (0.3), Student (0.8), Workshop (1.0), Meeting (0.7), Conference (0.7), Event (0.8)}	...	1	0,67	1,67
2478 - 2578	...	{Course (0.3), ResearchProject (0.3), GraduateStudent (1.0), MasterStudent (0.8), PhDStudent (0.8), Student (0.8)}	...	0,5	0,3	0,8

Figure 12: Query reformulations from P2178 to P2578.

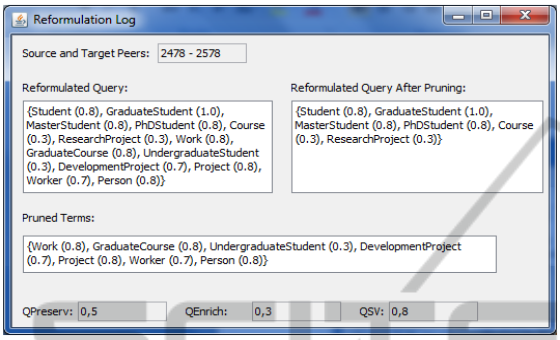

Figure 11: Reformulation log from P2478 to P2578.

latter, $QR_{pruning}$, regards Q reformulations using our pruning approach, i.e., *with pruning*.

We define that the SR is our "golden standard query", which would be capable of producing the best results for a particular user query. We use the Jaccard coefficient to evaluate the reformulated queries similarity with respect to SR in the query routing process ($QR_{without\_Pruning}$ and $QR_{pruning}$). Experimental results show the effectiveness of our approach when comparing query reformulations *without prunning* and *with pruning*. Figure 13 presents a summary of these similarity evaluation results. We observe that $QR_{pruning}$ remains in a better level of similarity with the SR. Moreover, in $QR_{without\_Pruning}$, the reformulated query tends to decrease the level of similarity with respect to the SR as long as it is forwarded to the next peers.

Besides, we consider the precision measure (Rijsbergen, 1979) as the ratio between the number of recovered relevant terms in the reformulated query and the total number of relevant terms in the SR. This measure is computed by each peer which participates in the query routing process.

As already mentioned, in our setting, peers are semantically related. Thus, when a query is forwarded to neighbour peers that are semantically distant from the original one, it is likely that it may occur a query semantic loss at each reformulation step. Nevertheless, the experimental results show that $QR_{pruning}$ has achieved a better level of precision. This is due to the fact that during the query routing process, the mechanism of pruning has eliminated the reformulated query terms that could produce a semantic loss of the original query. Figure 14 presents these results (i.e., the precision gain), by comparing $QR_{pruning}$ with $QR_{without\_pruning}$. In other words, the results indicate a precision increase when the query routing process take the semantic pruning into account.

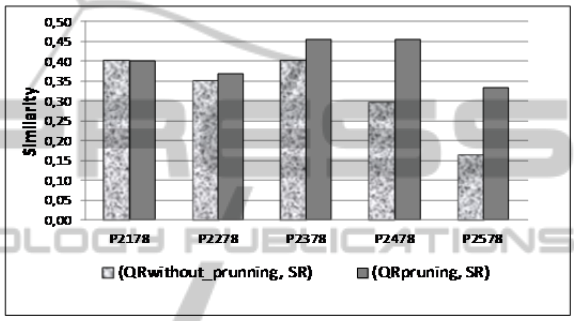

Figure 13: Similarity evaluation.

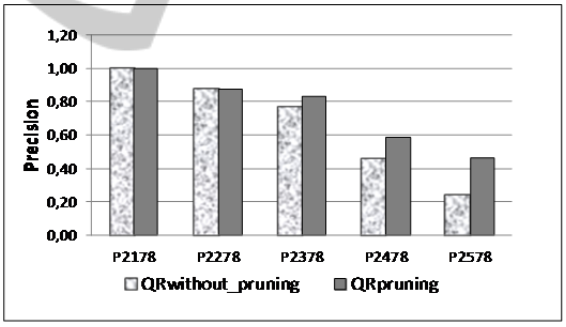

Figure 14: Precision measure.

6 RELATED WORK

Although there is some research concerning information loss analysis (Arenas et al., 2010, Roth 2011), few works (Kantere et al., 2009; Delveroudis et al., 2009) discuss the issue of query semantics preservation in query routing processes.

With respect to the first issue, Arenas and his group (Arenas et al., 2010) address the problem of providing foundations for metadata management by analysing schema mappings. In this work, a query is “target rewritable” if the mapping used to transfer source information is able to answer the query by

using the target data. The work of Roth (2011) provides a completeness-driven query planning. It forwards queries by considering peers and mappings that promise large result sets and mappings with low information loss. Histograms are used to estimate the potential data contribution of mappings.

In terms of query semantics preservation, Kantere *et al*. (2009) present GrouPeer, an adaptive, automated approach to clustering peers based on their common interests. This work allows peers to individually decide whether to answer the successively rewritten query or to automatically rewrite its original version. The work of Delvedouris *et al*. (2009) discusses the query semantic loss in query reformulation process. It proposes an algorithm that estimates the semantic loss of the rewritten queries by means of syntactic differences between the original query and the reformulated queries.

Differently from these related works, our approach enhances query routing processes by assuring the semantics preservation of the original query, as closely as possible. To this end, we use a semantic reference to avoid the query semantic loss and some metrics to assess the query semantic value. This value is used to avoid forwarding queries with a high semantic loss.

7 CONCLUSION AND FUTURE WORKS

In this work, we address the problem of preserving the original query semantics in query routing processes. We argue that the reformulated queries along the set of peers should be analyzed according to the original required query semantics. Query semantics evaluation may contribute not only to minimize the query routing time, but also to reduce the search space by considering only peers that can indeed contribute with relevant answers.

To help matters, we have proposed a semantic reference, which is built by considering a domain ontology (available as a background knowledge) according to a given submitted query. We use the semantic reference to avoid semantic loss at query reformulation time. Furthermore, we have specified three metrics in the light of a query routing process, namely: *query preservation*, *query enrichment* and *query semantic value*. These metrics have been used in some accomplished experiments, which have shown some benefits. Particularly, we verify that we can minimize the semantic loss and avoid forwarding queries with a high semantic loss. As a result, our approach helps to produce query results which best meet the users' needs.

As further work, we are integrating our approach with an information quality management service.The idea is joining the three defined metrics with other ones in order to select the most relevant neighbour peers to route queries.

REFERENCES

Arenas, M., Perez, J., Reutter, J. L., Riveros, C., 2010. Foundations of schema mapping management. In Proc. of *PODS*, Indianapolis, USA, pages 227–238.

Baader, F., Calvanese, D., McGuinness, D., Nardi D., Patel-Schneider P. Editors, 2003. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press.

Campos, L. M., Fernández-Luna, J. M., Huete, J. F., Vicente-López, E. 2013. XML search personalization strategies using query expansion, reranking and a search engine modification. In Proc. of *28th Annual ACM Symposium on Applied Computing - SAC '13,* Coimbra, Portugal, pages 872-877.

Carpineto, C., Romano, G. 2012. A Survey of Automatic Query Expansion in Information Retrieval. In ACM Computing Surveys, v.44, n.1, pages 1-50.

Delveroudis, Y., Lekeas, P. V. 2007. Managing Semantic Loss during Query Reformulation in PDMS. In *SWOD IEEE*, pages 51-53.

Delveroudis, Y., Lekeas, P. V., Souliou, D., 2009. On Estimating Semantic Loss in Peer Data Management Systems. In Proc. of *First International Conference on Advances in P2P Systems*, Sliema, Malta, pages 51-53.

Euzenat, J., Shvaiko, P., 2007. *Ontology Matching*. Springer-Verlag.

Han, J. W., Kamber, M., 2006. *Data Mining Concepts and Techniques: Elsevier Inc*, 2nd edition.

Kantere, V., Tsoumakos D., Sellis T., Roussopoulos N., 2009. GrouPeer: Dynamic Clustering of P2P Databases. In *Information Systems Journal*, v. 34, n. 1, pages 62–86.

Pires, C. E., 2009. *Ontology-based Clustering in a Peer Data Management System*. Ph.D. thesis, CIn/UFPE, Recife, Brazil.

Pires, C. E., Souza, D., Pachêco, T., Salgado, A. C., 2009. A Semantic-based Ontology Matching Process for PDMS. In Proc. of *2nd International Conference on Data Management in Grid and P2P Systems (Globe'09*), Linz, Austria, pages 124-135.

Rijsbergen, C. J. 1979. *Information Retrieval*, 2nd Ed. Stoneham, MA: Butterworths.

Roth, A. 2011. *Efficient Query Answering in Peer Data Management Systems*. PhD thesis, Humboldt Universität zu Berlin, Germany.

Roth A., Skritek, S., 2013. Peer Data Management. In *Data Exchange, Information, and Streams*, v. 5, pages

185-215.

Souza D., Arruda T., Salgado A. C., Tedesco P., Kedad, Z., 2009. Using Semantics to Enhance Query Reformulation in Dynamic Environments. In Proc. of *13th East European Conference on Advances in Databases and Information Systems* (ADBIS'09), Riga, Latvia, pages 78-92.

Souza, D., Pires, C. E., Kedad, Z., Tedesco, P., Salgado, A. C., 2011. A Semantic-based Approach for Data Management in a P2P System. In *Journal on Transactions on Large-Scale Data- and Knowledge-Centered Systems* (TLDKS) Lecture Notes in Computer Science, v. 6790, pages 56-86.