Supporting Semantic Search on Heterogeneous Semi-structured Documents

Yassine Mrabet[1,2], Nacéra Bennacer[2], Nathalie Pernelle[1], and Mouhamadou Thiam[1,2]

[1] LRI, Université Paris-Sud 11, INRIA Saclay, F-91893 Orsay cedex, France
{Yassine.Mrabet,Nathalie.Pernelle,Mouhamadou.Thiam}@lri.fr
[2] SUPELEC Systems Sciences (E3S), F-91192 Gif-sur-Yvette cedex, France
Nacera.Bennacer@supelec.fr

Abstract. This paper presents *SHIRI-Querying*[1], an approach for semantic search on semi-structured documents. We propose a solution to tackle incompleteness and imprecision of semantic annotations of semistructured documents at querying time. We particularly introduce three elementary reformulations that rely on the notion of aggregation and on the document structure. We present the Dynamic Reformulation and Execution of Queries algorithm (DREQ) which combines these elementary transformations to construct reformulated queries w.r.t. a defined order relation. Experiments on two real datasets show that these reformulations greatly increase the recall and that returned answers are effectively ranked according to their precision.

1 Introduction

The research advances on automating ontology population and document annotation are promising. But even for named entity-based approaches [1, 2] or pattern-based approaches [5] it remains difficult to locate precisely instances since some of them may be blended in heterogeneous semi-structured documents. The granularity of the annotation could be precise, at the term level, or imprecise, at the node level, in a semi-structured document [5]. In the worst case, the annotated unit is the whole document. Semantic imprecision may also appear when associated annotations are not accurate enough (e.g. using *Event* metadata instead of $Conference$ metadata). From another hand, annotations are often incomplete since automatic annotators do not find all instances and relations. To alleviate these problems, some semantic search systems try to gather answers satisfying the user query by going beyond the simple use of available metadata. Some approaches [4, 6] deal with semantic imprecision by approximating the concepts and the relations expressed in user queries using an ontology (e.g. exploiting subsomption, contextual closeness, path of semantic relations). Other works combine ontology-based search and classical keyword search [3, 7] in order to deal with incomplete annotations. The use of keywords increases the

[1] SHIRI : Digiteo labs project (LRI, SUPELEC).

B. Pernici (Ed.): CAiSE 2010, LNCS 6051, pp. 224–229, 2010.

recall by retrieving instances that are not reachable using semantic annotations, but some semantic constraints of the query are relaxed.

In this paper, we propose an ontology-based search approach called *SHIRI-Querying*. Our contributions are: (i) a reformulation method to query incomplete and imprecise semantic annotations of semi-structured documents (ii) an order relation that ranks the constructed queries according to their relevance and (iii) a dynamic algorithm which builds and executes reformulated queries w.r.t. the defined order. The *SHIRI-Querying* system uses the standard W3C languages RDF/OWL for representing resources and SPARQL for their querying. It has two main components. The *adapter* is designed to conform the provided annotations to the $SHIRI$ annotation model. It uses a set of logical rules and generates automatically the annotations base to be queried. The *Query Engine* processes ontology-based queries and reformulates them using the *SHIRI* annotation model. In this generic model [5] the granularity of the annotation is the document node (e.g. XML and HTML tags). Each node is annotated as containing one or several instances of different concepts of a given domain ontology. This allows bypassing the imprecise localisation of instances at the term level. The annotation model also allows representing structural links between document nodes, which enables dealing with the incompleteness of semantic relations in the provided annotations. We define three elementary query reformulations: the *SetOfConcept* and *PartOfSpeech* reformulations which allow retrieving instances that are aggregated in the same node and the *neighborhood-based* reformulation which allows retrieving instances located in close nodes that may be related by the required semantic relations. Reformulations of the user query are then obtained by combining these elementary transformations. The Dynamic Reformulation and Execution of Queries algorithm (*DREQ*) constructs these combinations and executes them w.r.t an order relation. This order relation gives priority to answers where nodes contain homogeneous instances and answers where nodes are linked by the required semantic relations. In contrast to most approaches which work on answers and/or whole annotated datasets, the answers are ranked as the reformulated queries are constructed. Experiments on two real datasets show that these reformulations greatly increase the recall and that the answers are effectively ranked according to their precision.

2 Annotation Model

Let $\mathcal{O}(\mathcal{C}_\mathcal{O}, \mathcal{R}_\mathcal{O}, \preceq, \mathcal{D}_\mathcal{O})$ be the domain ontology where $\mathcal{C}_\mathcal{O}$ is the set of concepts, $\mathcal{R}_\mathcal{O}$ is the set of relations between concepts ($\mathcal{R}_O^f$, $\mathcal{R}_O^{if}$ are resp. functional and inverse functional relations), $\preceq$ denotes the subsumption relation between concepts or relations and $\mathcal{D}_\mathcal{O}$ defines the domain and the range for each relation. The annotation model, denoted $\mathcal{A}(\mathcal{C}_\mathcal{A}, \mathcal{R}_\mathcal{A}, \preceq, \mathcal{D}_\mathcal{A})$, is generated automatically from the domain ontology. $\mathcal{C}_\mathcal{A} = \mathcal{C}_\mathcal{O} \cup \mathcal{C}_\mathcal{S}$, $\mathcal{R}_\mathcal{A} = \mathcal{R}_\mathcal{O} \cup \mathcal{R}_\mathcal{S}$. $\mathcal{C}_\mathcal{S}$ and $\mathcal{R}_\mathcal{S}$ are the concepts and the relations defined for the annotation task. In this model, concept instances are identified by URIs of document nodes and the literals associated by the *hasValue* attribute are the textual contents of annotated nodes.

We define the following aggregate metadata in C_S and R_S:

- The *PartOfSpeech* concept is used to annotate document nodes containing several instances of different concepts.

- The *SetOfConcepts* metadata is used to annotate document nodes containing several instances of the same concept. A concept $SetOfc_i$ is defined as a subclass of *SetOfConcepts* for each concept $c_i \in \mathcal{O}$. Moreover, we define relations denoted $rSet$ and $rSet^{-1}$ in $\mathcal{R}_\mathcal{S}$ derived from (inverse) functional relations r in $\mathcal{R}_\mathcal{O}$ in order to represent relations between an instance and a set of instances.

- The *neighborOf* relation expresses a path in a XML/HTML document tree.

Instances of these metadata are generated by the *adapter* using a set of logical rules. If a document node contains only one instance of a domain concept c, it is annotated by c. The datatype properties of this instance become properties of the node. Else, it is annotated either by $SetOfc_i$ metadata or *PartOfSpeech* metadata. The property *isIndexedBy* is instantiated for *PartOfSpeech* nodes. The provided annotations of domain relations $r \in \mathcal{R}_\mathcal{O}$ are instantiated between nodes whose types are in $\mathcal{C}_\mathcal{O}$ (domain concepts). In the case where r links a node of type $c_j \in \mathcal{C}_\mathcal{O}$ with a node of type $SetOfc_i$, r is substituted by $rSet$ or $rSet^{-1}$.

3 Query Reformulations

Preliminary Definitions: Consider the pairwise disjoint infinite sets I, B, L and V (IRIs, Blank nodes, Literals and Variables). A triple pattern is a triple $(s, p, o) \in (I \cup V) \times (I \cup V) \times (I \cup V \cup L)$. A **basic graph pattern** P is a set of triple patterns. $?v$ in a triple indicates that v is a variable. An RDF query is a basic graph pattern or a constructed graph pattern (using constructors such as union or intersection). To facilitate the reading of this paper, we consider only queries described by basic graph patterns. The filters that we consider use equality and inclusion operators between variables and literal values.

We define a **model-based query** q as a quadruplet (P, S, F, D) where :
- P is a basic graph pattern which complies with a model (i.e. $\mathcal{O}$ or $\mathcal{A}$). $V(P)$ denotes the set of variables of P and $C(P)$ denotes the set of concepts of P.
-F is a constraint defined by a logical combination of boolean-valued expressions.
-S is the set of variables that appear in the $SELECT$ clause of the query.
-D is an $\mathcal{A}$-compliant RDF dataset to be matched with P and F.

Example: The $\mathcal{O}$-based query q_0 is defined by (P_0,F_0,S_0,D) where :
P_0 ={ ($?art$, rdf:type, Article), ($?aut$, rdf:type, Person), ($?aut$, hasName, $?aName$), ($?conf$, rdf:type, $Conference$), ($?art$, publishedIn, $?conf$) ($?art$, authoredBy, $?aut$), ($?conf$, hasName, $?cName$)}
F_0 : {$?cName =$ "$WWW2008$"} and S_0 : { $?art$, $?aut$, $?aName$ }

Neighborhood-based Reformulation: The aim of the neighborhood-based reformulation is to exploit the structural neighborhood of document nodes in order to find nodes that may be related by the semantic relations expressed in the user query. The neighborhood-based reformulation, denoted f_{nr}, substitutes the ontological relation of a given triple by a $neighborOf$ relation.

Example: $f_{nr}(q_0, (?art, authoredBy, ?aut)) = q'_1(P'_1, F_0, S_0, D)$ where $P'_1 = \{(?art, rdf{:}type, Article),\ ...(?art, neighborOf, ?aut),\ ...\}$. Applying f_{nr} may generate semantically-independent subgraph patterns.

Definition 1. *p is a **semantically-independent subgraph pattern** of P if :*
$-\forall v_1, v_2 \in V(p),\ (?v_1, r, ?v_2) \in P \rightarrow (?v_1, r, ?v_2) \in p$
$-\forall (?v_1, r, ?v_2) \in P$ *s.t.* v_1*(resp.* v_2*)* $\in V(p)$, v_2*(resp.* v_1*)* $\notin V(p) \rightarrow r{=}neighborOf$
Splitting a query into semantically-independent subgraph patterns allows applying distinct aggregative reformulations on distinct sub-parts of a query.

***PartOfSpeech* Reformulation:** The *PartOfSpeech* reformulation denoted f_{pr} assumes that the required semantic relations can be found between instances aggregated in the same node. It is applied on semantically-independent subgraph patterns. The target subgraph must be $\mathcal{O}$-based, i.e. it does not contain metadata from C_S or R_S (this constraint is part of the reformulations construction plan). f_{pr} substitutes all filter constraints of the subgraph pattern by filter constraints on the textual contents of $PartOfSpeech$ nodes. Equality constraints are relaxed into inclusion constraints.

Example: for $p \in P'_1$ s.t. $p = \{(?art, rdf{:}type, Article),\ (?conf, rdf{:}type, Conference),\ (?conf, hasName, ?cName),\ (?art, publishedIn,\ ?conf)\}$,
$f_{pr}(q'_1, p) = q'_2(P'_2, F'_2, S'_2, D)$ s.t.
$P'_2 = \{(?pos,$ rdf:type, $PartOfSpeech),\ (?pos,$ isIndexedBy, $Conference),\ (?pos,$ isIndexedBy, $Article),\ (?pos,$ hasValue, $?lPos),\ (?pos,$ neighborOf, $?aut),\ (?aut,$ rdf:type, $Person),\ (?aut,$ hasName, $?aName)\ \}$
$F'_2 : \{(?lPos\ contains\ "WWW2008")\}$, $S'_2 : \{$?aut, ?aName, $?pos\}$

***SetOfConcept* Reformulation:** The *SetOfConcept* reformulation, denoted f_{sr}, substitutes the ontological type c of a given variable v by the $setOfc$ type if for all triples of P: (1) if v is the subject, the relation r is not inverse functional and (2) if v is the object, the relation is not functional. The relation r is then substituted by $rSet^{-1}$ (case 1) or $rSet$ (case 2).

Example: $f_{sr}(q_0, ?aut) = q'_3(P'_3, F_0, S_0, D)$ s.t.
$P'_3 = \{(?art,$ rdf:type, $Article),\ (?aut,$ rdf:type, $SetOfPersons),\ (?conf,$ rdf:type, $Conference),\ (?art,$ publishedIn, $?conf),\ (?art,$ authoredBySet, $?aut),\ (?aut,$ hasValue, $?aName),\ (?conf,$ hasName, $?cName)\}$

Reformulations Construction Plan: The reformulation of a query $q_0(P_0, F_0, S_0, D)$ is a query $q_i(P_i, F_i, S_i, D)$ obtained by the composition of elementary *PartOfSpeech*, *SetOfConcept* and *neighborhood-based* reformulations. We consider that a set of document nodes is more relevant if its nodes do not contain aggregated instances and if they are related by the expected semantic relations.

Definition 2. *Let $N(q)$, $Pos(q)$ and $Sets(q)$ be resp. the number of $neighborOf$, $PartOfSpeech$ and $SetOfc$ metadata in a query q. The (well) order $\preccurlyeq$ is defined s.t.* $q_i \preccurlyeq q_j \leftrightarrow ((N(q_i) > N(q_j)) \vee ((N(q_i) = N(q_j)) \wedge ((Pos(q_i) > Pos(q_j)) \vee ((Pos(q_i) = Pos(q_j)) \wedge (Sets(q_i) \geq Sets(q_j))))$

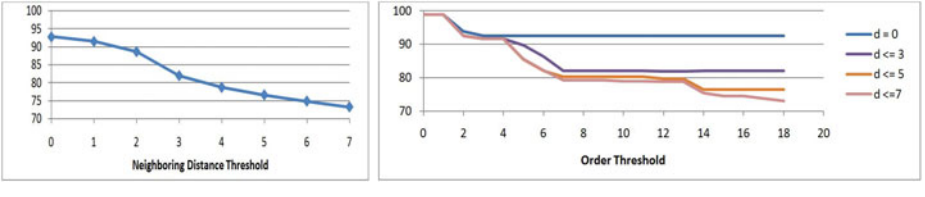
Dynamic Reformulation and Execution of Queries Algorithm (DREQ) DREQ allows constructing and executing the reformulated queries with respect to $\preccurlyeq$. When $DREQ$ is stopped at a given order, the answers are those retrieved by the best constructed queries. $DREQ$ computes all reformulations in EXPTIME w.r.t the number of variables of the user query, but, as the algorithm is dynamic, we obtain a new set of equally-ordered reformulations in PTIME.

4 Experimental Results

SHIRI-Querying has been implemented and experimented to study how the precision and the recall measures vary according to the order relation. The $neighborOf$ relation is defined as an undirected path of length d in the HTML /XML tree. We also study how d influences the results. The reformulations proposed in our approach can introduce wrong answers which may appear when the query is reformulated using f_{sr} and f_{pr} which relax filters in $PartOfSpeech$ or $SetOfConcepts$ nodes or using f_{pr} and f_{nr} which relax semantic relations. The two experimented datasets belong to the scientific conferences and publications domain.

The first dataset is composed of annotated publication references provided by the DBLP XML repository, the INRIA HTML server and the HAL XML repository. It consists of more than 10.000 RDF triples describing 1000 publications. We submitted a set of queries looking for conferences, their dates, locations, papers and authors. A precision of 100% and a recall of 100% were reached with an order threshold of 9 and $d \leq 3$. A smaller order threshold leads to a smaller recall and a higher distance d leads to almost 0% precision. In this case ($d > 3$), in two data sources, each paper is associated to all conferences. The 100% values for the recall and the precision measures are due to the regular structure of the data sources. However, each data source has a different and specific structuring and the $DREQ$ reformulations were able to integrate answers from all sources.

The second corpus consists of RDF annotations of 32 call-for-papers web sites and is consequently very heterogeneous. These annotations (consisting of 30.000 RDF triples) were generated automatically using $SHIRI - Extract$ [5]. We then submitted a set of 15 queries. Without reformulation, all queries have no answers (0% recall), while we obtained a 56% recall by using the $DREQ$ algorithm for $d \leq 7$. At the same distance threshold ($d \leq 7$) the precision is still 72%. The results



(a) Precision according to d (b) Precision according to order

Fig. 1. Answers' Precision

show that domain relations can often be retrieved between instances located in close document nodes. Figure 1(b) presents the average precision value for the same set of user queries, for several values of d, by varying the order threshold from 1 to 18. The precision variations show that the order relation is relevant to rank the answers.

5 Conclusion and Future Work

In this paper, we presented the *SHIRI-Querying* approach to support semantic search on heterogeneous semi-structured documents. Ontology-based user queries are reformulated to gather document nodes from documents that were annotated in an imprecise and incomplete manner by semantic annotation tools. These reformulations allow retrieving instances that are related by the requested semantic relations even if these relations are not available in the knowledge base. We defined an order relation between reformulated queries to give priority to queries that preserve most the semantics of the user query. All reformulations are constructed dynamically w.r.t this order relation in the *DREQ* algorithm. Experimental results show that the recall greatly increases and that the precision decreases reasonably as the ordered reformulated queries are performed. In the near future we plan to combine keyword-based search with our reformulation approach to increase the recall without losing the semantics of the query. We also plan to use semantic-based heuristics exploiting functional properties of relations in order to avoid some wrong answer cases.

References

1. Borislav, P., Atanas, K., Angel, K., Dimitar, M., Damyan, O., Miroslav, G.: KIM - Semantic Annotation Platform. J. of Nat. Lang. Engineering 10(3-4), 375–392 (2004)
2. Etzioni, O., Cafarella, M., Downey, D., Kok, S., Popescu, A., Shaked, T., Soderland, S., Weld, D., Yates, A.: Unsupervised named-entity extraction from the web: An experimental study. Artificial Intelligence 165(1), 91–134 (2005)
3. Bhagdev, R., Chapman, S., Ciravegna, F., Lanfranchi, V., Petrelli, D.: Hybrid Serach: Effectively Combining Keywords and Semantic Searches. In: Bechhofer, S., Hauswirth, M., Hoffmann, J., Koubarakis, M. (eds.) ESWC 2008. LNCS, vol. 5021, pp. 554–568. Springer, Heidelberg (2008)
4. Corby, O., Dieng-Kuntz, R., Gandon, F., Faron-Zucker, C.: Searching the semantic web: Approximate query processing based on ontologies. IEEE Intelligent Systems Journal, Computer Society 21(1), 20–27 (2006)
5. Thiam, M., Bennacer, N., Pernelle, N., Lo, M.: Incremental Ontology-Based Extraction and Alignment in Semi-Structured Documents. In: Bhowmick, S.S., Küng, J., Wagner, R. (eds.) DEXA 2009. LNCS, vol. 5690, pp. 611–618. Springer, Heidelberg (2009)
6. Hurtado, C.-A., Poulovassilis, A., Wood, P.-T.: A Relaxed Approach to RDF Querying. In: Cruz, I., Decker, S., Allemang, D., Preist, C., Schwabe, D., Mika, P., Uschold, M., Aroyo, L.M. (eds.) ISWC 2006. LNCS, vol. 4273, pp. 314–328. Springer, Heidelberg (2006)
7. Castells, P., Fernàndez, M., Vallet, D.: An adaptation of the vector-space model for ontology-based information retreival. IEEE T. on Know. and Data Eng. 19(2) (2007)