Updating Direct Graph for Incremental Reasoning in OWL 2 QL Ontology

Changlong Wang[1,2], Zhiyong Feng[1] , Xin Wang[1],Guozheng Rao[1],Xiaowang Zhang[1]

[1]School of Computer Science and Technology, Tianjin University,Tianjin 300072, China,
[2]School of Computer Science and Engineer Technology, NWNU,Lanzhou 730070, China,

Abstract. *We propose an incremental reasoning approach to QL ontologies by mapping an evolving ontology to an updatable digraph and maintaining dynamic transitive closure to obtain incremental classification. We describe the procedure of updating ontology digraph and present an approach to identify the affected paths for incremental classification. We implement the proposed method in a prototype incR and evaluated it on widely-used ontologies. The experiments show that our approach can lead to performance gain.*

Keywords: Ontology; Incremental Reasoning; Direct Digraph

1 Introduction

Existing incremental reasoning techniques mainly focused on ontologies in OWL 2 and its profiles OWL 2 EL and OWL 2 RL [1–3]. In recent years, OWL 2 QL ontologies have become increasingly important in Ontology-Based Data Access (OBDA) [4]. However, to the best of our knowledge, there exists no incremental classification approach or implementation tailored to QL ontologies. In this paper, we close this gap by extending the graph-based approach for static QL ontolgoies in [5].

2 Preliminaries

Graph Theory Notions. We use traditional notions in graph theory. A digraph D consists of a non-empty finite vertex set *V(D)* and edge set *E(D)*. The degree of a vertex v is denoted by $d(v)$. A path p is a sequence p=$v_1, v_2, ..., v_k$ of distinct vertices in $V(D)$, such that for every $v_i, v_{i+1} \in p$, $(v_i, v_{i+1}) \in E(D)$. A path from u to v is denoted with $u \rightsquigarrow v$. The size of path p is represented by the number of edges contained in p. The number of distinct paths $u \rightsquigarrow v$ is denoted by $Paths(u, v)$. We use $TC(D)$ to denote the transitive closure of a digraph D.

Digraph Representation of QL 2 Ontology. The idea of digraph representation of a QL ontology is that, given an OWL 2 QL ontology O and its digraph D_O, each vertex $v \in V(D_O)$ represents a basic concept(role). Let S_1 and S_2 be two atomic concepts(roles), $O \models S_1 \sqsubseteq S_2$ iff $\exists(S_1, S_2) \in E(TC(D_O))$. Please refer to [5] for details. We use $L(\alpha)$ (resp. $R(\alpha)$) to denote the basic concept or basic role on the left (resp. right) sides of α. we also use $d(L(\alpha))$ or $d(R(\alpha))$ to represent degree of the corresponding vertex. For an axiom of the form $Q_1 \sqsubseteq Q_2$, we use $d(Q_1)$ and $d(Q_2)$ to represent the degrees of Q_1 and Q_2, respectively.

Table 1. Test ontologies

Ontology	**DL**	$\sharp A.C.$	$\sharp A.R.$	$\sharp Ont$	T_{QUONT}
Mouse	ALE	2753	1	3463	0.104
Gene	SH	26225	4	42655	1.236
EL-Galen	$ALCH(D)$	23136	950	97811	3.528
FMA-OBO	ALE	75139	2	119558	4.723

3 Updating Digraph

When an axiom is added to an ontology, it is easy to update the corresponding digraph. Let O be an ontology and α^+ be an added axiom, if the basic concepts and basic roles in α^+ are already contained in O, we only insert edge(s) into D_O; if α^+ contains new basic concepts and basic roles, we insert vertex and edge(s) according to Definition 2 in [5]. Axiom deletions are complicated and discussed in details as follows:

From Definition 2 in [5], there exist some differences between concept axiom additions and role axiom additions. Let α^- be a deleted axiom, we consider two cases :
— Case 1: α^- is a concept inclusion.
(1) If $d(L(\alpha^-)) > 1$ and $d(R(\alpha^-)) > 1$, then, only the edge $(L(\alpha^-), R(\alpha^-))$ is removed.
(2) If $d(L(\alpha^-)) > 1$ and $d(R(\alpha^-)) = 1$, then, the edge $(L(\alpha^-), R(\alpha^-))$ and its head are removed.
(3) If $d(L(\alpha^-)) = 1$ and $d(R(\alpha^-)) > 1$, then, the edge $(L(\alpha^-), R(\alpha^-))$ and its tail are removed.
(4) If $d(L(\alpha^-))) = 1$ and $d(R(\alpha^-)) = 1$, then, the edge $(L(\alpha^-), R(\alpha^-))$ is removed, both its tail and head are removed.
— Case 2: α^- is a role inclusion axiom of the form $Q_1 \sqsubseteq Q_2$.
(1) If $d(Q_1) > 1$ and $d(Q_2) > 1$, then, the 4 edges (Q_1, Q_2), (Q_1^-, Q_2^-), $(\exists Q_1, \exists Q_2)$, $(\exists Q_1^-, \exists Q_2^-)$ need to be removed.
(2) If $d(Q_1) > 1$ and $d(Q_2) = 1$, then, the 4 edges (Q_1, Q_2), (Q_1^-, Q_2^-), $(\exists Q_1, \exists Q_2)$, $(\exists Q_1^-, \exists Q_2^-)$, and the 4 vertices Q_2, Q_2^-, $\exists Q_2$, $\exists Q_2^-$ need to be removed.
(3) If $d(Q_1) = 1$ and $d(Q_2) > 1$, then, the 4 edges (Q_1, Q_2),(Q_1^-, Q_2^-), $(\exists Q_1, \exists Q_2)$,$(\exists Q_1^-, \exists Q_2^-)$, and 4 vertices Q_1, Q_1^-, $\exists Q_1$,$\exists Q_1^-$ need to be removed.
(4) If $d(Q_1) = 1$ and $d(Q_2) = 1$, then, the 4 edges (Q_1, Q_2),(Q_1^-, Q_2^-), $(\exists Q_1, \exists Q_2)$,$(\exists Q_1^-, \exists Q_2^-)$, and the 8 vertices Q_1, Q_1^-, $\exists Q_1$, $\exists Q_1^-$,Q_2,Q_2^-, $\exists Q_2$,$\exists Q_2^-$ are removed.

4 Identifying the Affected Path for Incremental Classification

An edge addition (deletion) into (from) digraph may change the number of path $i \rightsquigarrow j$. It is necessary to identify those affected paths and recompute the transitive closure of such affected subdigraph.

Definition 1 (affected path). Given a digraph D_O containing vertices i, j, u, v, and path $i \rightsquigarrow j$, if the number of $i \rightsquigarrow j$ changes after an edge (u, v) is inserted (removed) into (from) D_O, the path $i \rightsquigarrow j$ is called *affected path*.

When we add (delete) axioms, no cycles is created in the corresponding digraph D_O. From Lemma 3.1 in [6], the following proposition holds:

Table 2. IncR VS.Pellet.Time in seconds

Ontology	n	$Number\ of\ A.P.(Av/Mx)$	$Size\ of\ A.P.(Av/Mx)$	$T_{IncR(Av/Mx)}$	$T_{Pellet}(Av/Mx)$
mouse	1	96/321	148/431	0.002/0.032	0.021/0.039
mouse	2	164/404	170/518	0.003/0.043	0.044/0.063
mouse	4	225/664	274/673	0.006/0.061	0.068/0.072
mouse	8	452/811	471/831	0.013/0.084	0.082/0.095
Gene	1	23/71	42/79	0.006/0.182	0.029/0.191
Gene	2	58/104	63/119	0.112/0.243	0.357/0.489
Gene	4	76/164	98/294	0.128/0.302	0.432/0.764
Gene	8	149/394	219/438	0.363/0.414	0.927/1.014
EL-Galen	1	213/871	432/968	0.018/0.211	0.698/1.164
EL-Galen	2	486/1004	586/1484	0.024/0.298	0.835/2.718
EL-Galen	4	869/1864	1076/2038	0.217/0.393	1.107/2.973
EL-Galen	8	327/714	1482/3752	0.572/0.819	1.328/3. 219
FMA-OBO	1	45/89	83/174	0.007/0.131	0.012/0.275
FMA-OBO	2	69/127	192/387	0.148/0.573	0.726/0.915
FMA-OBO	4	136/259	293/653	0.152/0.512	0.759/1.416
FMA-OBO	8	283/894	461/934	0.263/0.748	1.663/1.943

Proposition 1. Let $i \rightsquigarrow j$ be an affected path after the edge (u, v) is inserted (or deleted), the number of the paths $i \rightsquigarrow j$ increases (or decreases) by $Paths(i, u) * Paths(v, j)$.

In ontology development, a domain engineer often adds (or deletes) a set of axioms related to a certain concept (role). Such modification will lead to a changed set of edges incident to a common vertex and more affected paths in the corresponding digraph.
Proposition 2 [6] Let D_O be a digraph and E_v be a updating set of edges incident to a common vertex v, let $\Delta Paths(i, j)$ be the change to $Paths(i, j)$,

$\Delta Paths(i, j) \leftarrow Paths(i, v) * \Delta Paths(v, j) + \Delta Paths(i, v) * Paths(v, j) + \Delta Paths(i, v) * \Delta Paths(v, j)$.

From above discussion, it is feasible to design an algorithm for incremental classification in QL ontologies. Give a digraph D and any two vertices $i, j \in D$, the number of the path $i \rightsquigarrow j$ can be represented by an $n \times n$ adjacency matrix Q such that $Q(i, j) = 0$ if there exists no path from i to j. The transitive closure of a digraph D is presented by an $n \times n$ adjacency matrix M_D^* such that $M_D^*(i, j)$=1 if $Q(i, j) \neq 0$, else $M_D^*(i, j)$=0. While updating the digraph, if $Q(i, j)$ increases from 0, $M_D^*(i, j)$=1, if $Q(i, j)$ decreases to 0, $M_D^*(i, j)$=0.

5 Implementation and Evaluation

We implement our approach in IncR. We perform experiments to compare IncR with QUONTO [4] and Pellet [1]—QUONT supports the graph-based approach in [5] and Pellet supports module-based incremental classification [1].

We select 4 widely-used ontologies from Bioportal [1], of which the EL-Galen falls out of OWL 2 QL and need to be approximated by a QL ontology. Table 1 provides

[1] http://bioportal.bioontology.org/

basic information: the DL language , the number of atomic concept and role, the size of ontology. The last column of Table 1 presents the time in seconds taken by QUONTO.

In order to compare IncR with Pellet, we use similar experimental methodology in [1]: for various values of n, 1) remove n random axioms; 2) classify the resulting ontology using IncR and Pellet ; then, we repeat the following steps 30 times: 3) randomly remove an additional n axioms, add back the previously removed n axioms, and reclassify the ontology using IncR and Pellet. All the results are gathered during step 3) of the experiment. All tests are performed on a server with Intel Xeon E5620 2.40GHz CPU, running Windows Server 2008 R2 Enterprise and Java 1.7 with 16GB of RAM available to JVM.

Table 2 summarizes the results of our experiments for n=1,2,4 and 8, where Av means average value and Mx represents maximum value. Column 3 $Number of A.P.(Av/Mx)$ and 4 $Size of A.P.(Av/Mx)$ indicates the number of affected paths and their total size, respectively. Colomn 5 shows the time spent in maintaining the transitive closure of the updated digraph, the time value includes time spent in identifying affected paths and time in re-computing the transitive closure of all the affected paths. The last column provides the time taken to classify those updated ontologies by Pellet.

6 Conclusion

We have proposed an approach to incremental classification in QL ontology by exploiting digraph algorithm, which allows us to efficiently identify the small affected paths and to reuse previous computation. We demonstrate the performance gain indicated by the significant reduction of classification time. As a big challenge in the future, we will develop a parallel version of our method to exploit many cores/CPUs available in modern systems.

Acknowledgments. This work is supported by the NSFC (61100049,61373165) and the 863 Program of China (2013AA013204).

References

1. Grau B. C., Wiener C. H., Kazakov Y.,et al.: Incremental classification of description logics ontologies. J. of Automated Reasoning. 44(4), 337-369, 2010.
2. Kazakov Y., Klinov P.: Incremental reasoning in OWL EL without bookkeeping. In ISWC, 232-247, 2013.
3. Motik B., Nenov Y., Piro R., Horrocks I.: Incremental update of datalog materialisation : the backward/forward algorithm. In AAAI, 1560-1568, 2015.
4. Calvanese D., Giacomo G. D., Lembo D., Lenzerini M.,et al.: The mastro system for ontology-based data access. Semantic Web, 2(1), 43-53, 2011.
5. Domenico L., Santarelli V., Savo D. F.: A graph-based approach for classifying owl 2 QL ontologies. In Description logic, 747-759, 2013.
6. King, Valerie, Garry Sagert: A fully dynamic algorithm for maintaining the transitive closure. In STOC, 492-498, 1999.