

基于优先级的超协调本体推理研究^{*})

王治纲 孙小林 文坤梅 李玉华

(华中科技大学计算机学院 武汉 430074)

摘要 随着语义 Web 研究的逐步深入,本体推理问题业已受到众多相关人员的重视。而超协调本体的推理问题,作为本体维护以及本体集成的研究基础,更是迫切地有待解决。本文对本体知识库作出了一定的扩充,定义了基于优先级的本体知识库的一系列相关概念,并且在此基础上,给出了两种在超协调的本体中寻求一致性推理的途径,希望能为相关的研究带来一定的参考与借鉴。

关键词 本体,推理,非一致性,优先级

The Study of Reasoning about Inconsistent Ontology with Priorities

WANG Zhi-Gang SUN Xiao-Lin WEN Kun-Mei LI Yu-Hua

(College of Computer Science & Technology, Huazhong University of Science & Technology, Wuhan 430074)

Abstract Reasoning with inconsistency is a well-known topic in logics and AI. Several approaches have been proposed to deal with inconsistent ontology in current Semantic Web researches. The objective of this paper is to propose solutions for handling conflicts of inconsistency with an assumption of prioritized knowledge base. We have introduced the formal definitions of the prioritized knowledge base of ontology, and investigated the strategies of inconsistency reasoning processing based on the priority.

Keywords Ontology, Reasoning, Inconsistency, Priority

1 概述

本体是对语义网中各种分布式共享概念的形式化描述。由于描述上或者分类上的含混、语义上的冲突等多种因素,都会造成本体的不一致性,这种不一致又会为推理问题带来新的困扰。在描述逻辑的推理中,对于一个本体知识库 K 和一个公式 ψ ,推理过程可以归约到判断 $K \Box \psi$ 是否为可满足的。当 $K \Box \psi$ 成立时,即对查询回答“是”;当 $K \Box \neg \psi$ 时,则回答“否”。在一致性本体库中, $K \Box \psi$ 和 $K \Box \neg \psi$ 是无法同时得到满足的。然而由于集成或本体描述本身而带来的矛盾性,可能会导致某些概念及其否定在本体库中同时存在,体现为整个本体知识库的不一致性,或者称之为超协调性,形式化表示为 $K \Box \perp$,这里我们用符号 $\perp$ 表示描述逻辑中的空概念。由于经典逻辑具有平凡性,知识库的这个矛盾的前提会导致推理中可能推出任意的结论。

目前,在超协调本体的推理研究中已经有了一些探索,主要有两类方法来处理本体的非一致性。第一类方法是对发生冲突的本体进行诊断和修改^[1],而在大型的、分布式的环境中,对本体进行修改是不太现实和可行的;第二种方法是通过使用一些非标准的推理方法,来回避本体的非一致性。董明楷等将缺省逻辑引入到了描述逻辑中,以描述逻辑为主要框架,对单调逻辑和非单调逻辑进行了整合^[2]。文^[3]提出了一个超协调本体的推理的基本框架,并借用了“概念的相关性”(concept relevance)来实现非一致本体库中的推理。

本文借鉴了文^[2]中的一些思想,并没有直接引入非单调推理的方法,而是试图在超协调的本体库中寻找一个尽可能大的一致性子集,在这个子集上来完成单调性的逻辑推理。我们在本体知识库中引入优先级的概念,按照本体的优先级,

为超协调本体定义了一种偏序关系。而在进行推理时,依照这种优先级的偏序性,来依次装载相应的本体知识库,在保持知识库的一致性同时,确保优先级高的本体先得到应用。

2 基于优先级的本体知识库的基本概念

为了提供一种通用性的超协调本体推理方法,需要首先摆脱特定语言的角度来看待本体,通常,可以将一个本体语言看作是一组语法规则的集合,而本体知识库可以看作是一阶谓词公式的集合。

定义 1 一个本体知识库可定义为一个二元组 $K = \langle T, A \rangle$,其中 T 为术语库 $Tbox$, A 为断言库 Aox 。概念 C 在本体库 K 中是可满足的,当且仅当 C 在 $\langle T, A \rangle$ 中并且 C 不包含于 $\perp$,否则称概念 C 是不可满足的。如果 C 是可满足的,则 C 是 K 蕴含的,记为 $K \Box C$ 。

如前所述,如果将本体知识库看作是一个有关概念以及概念之间各种关系的规则的集合,那么可以根据每条规则在本体库中的重要程度给该规则拟订一个优先级。现在可以定义一个映射每条规则与优先级别之间的一个映射。

定义 2 优先级映射 $PR: K \times N \rightarrow \{0\}$,其中 N 代表自然数集,它将知识库 K 中每条规则 φ 都映射到了一个非负的整数优先级上。

对优先级语义上的解释可以定义为:若 $PR(\varphi_1) < PR(\varphi_2)$,则规则 φ_2 在知识库 K 中享有比 φ_1 更高的优先度,显然,优先级为 0 的规则,在知识库中的优先等级是最低的。根据这个映射,就可以将原有的知识库 K 扩展为基于优先级的分级知识库 ΠK 。

定义 3 优先级知识库 ΠK ,是包括了对知识进行描述的规则以及该规则的优先级所组成的集合,形式化表示为 ΠK

^{*})本课题获国家自然科学基金项目(60403027)、湖北省自然科学基金项目(2005ABA258)、软件工程国家重点实验室开放基金项目(SKLSE05-07)资助。

$=\{(\varphi_i, p_i): i=1, 2, \dots\}$, 其中 φ_i 代表了规则, p_i 表示了该规则所对应的优先级。

根据规则优先级的不同, 又可以将知识库做一个划分 $\Pi K=(\Pi K_1, \Pi K_2, \dots)$, 其中 ΠK_i 代表了拥有相同优先等级 p_i 的规则集合, 即 $\Pi K_i=\{(\varphi_j, p_i): j=1, 2, \dots\}$, 相应的有 $PR(\Pi K_i)=p_i$, 值得注意的是, 这里的 i 和 p_i 并不一定是相等的。同样地, 若 $PR(\Pi K_i)<PR(\Pi K_j)$, 则 ΠK_j 比 ΠK_i 有更高的优先级。在定义好了基于优先级的知识库以后, 下面就可以进一步讨论本体知识库中的一致性问題。

定义 4 非一致性等级, 对于一个优先级知识库而言, 它的非一致性等级等于它的最小非一致性知识库的优先级, 记为 $\text{Inc}(\Pi K)$ 。

如果 ΠK 是一致的, 那么 $\text{Inc}(\Pi K)=0$, 否则 $\text{Inc}(\Pi K)=\max\{p_i: (\varphi_i, p_i) \square \Pi K \square U_{\geq p_i} \square \perp\}$, 其中 $U_{\geq p_i}$ 表示所有优先级大于或等于 p_i 的规则所构成的知识库, 它也可以表示为 $\text{Inc}(\Pi K)=\max\{p_i: (\bigcup \Pi K_i: (PR(\Pi K_i) \geq p_i) \square \perp)\}$

定义 5 冲突单元, 给定一个优先级知识库 K , K 中的一个冲突单元 C 为 K 中规则的一个集合, 并且这个集合必须满足: ① C 是不一致的; 且 ② C 必须是最小的, 即:

$$C \square \perp \square (\square \varphi: \varphi \square C, C - \{\varphi\} \square \perp)$$

定义 6 冲突规则集: 至少涉及到了一个冲突 C 的所有规则的集合, 记为 X 。

对于一个知识库而言, 冲突规则集 $X(K)$ 是 K 中所有冲突的并集, 即:

$$C(K) = \bigcup \{C: C \subseteq K\}$$

3 优先级本体知识库中的超协调推理算法

非单调推理是知识表示和推理的一个重要分支。在一般形式下, 其形式化通常是不可判定的。基于这个原因, 用于可判定的一些子情形的过程基本上都限制在命题逻辑, 从而使命题逻辑和一阶逻辑之间有了更大差距。

出于这些因素的考虑, 本文并没有直接在超协调本体推理中引入非单调推理的方法, 而是试图在不一致的优先级本体库中, 通过本体之间的优先级顺序来解决冲突问题, 寻找一个尽可能大的一致性的本体知识库子集, 然后在这个子集上来完成单调性的逻辑推理。

3.1 冲突即终止算法

冲突即终止算法的基本思想, 直感上可以理解为, 系统从具有最高优先级的知识库开始进行推理, 逐级加入优先级较低的知识库层, 当新加入的知识库引发了冲突时即停止, 则上次所获得的就是最大的一致性知识库, 然后再可以按照这个知识库的内容对查询作出应答。

在进行算法描述时, 我们首先假定这个优先级本体知识库 ΠK 由 n 个按照优先级的大小来分层的本体知识库组成, 即

$$\Pi K=(\Pi K_1, \Pi K_2, \dots, \Pi K_n), \text{ 且 } PR(\Pi K_1)>PR(\Pi K_2)>\dots>PR(\Pi K_n)$$

则我们可以定义冲突即终止算法如下:

算法 1 冲突即终止算法

input: 分层的优先级知识库 $\Pi K=(\Pi K_1, \Pi K_2, \dots, \Pi K_n)$
output: 一致性本体知识库 KB
begin
 $KB \leftarrow \emptyset$;
 $i \leftarrow 1$;
 while ($i \leq n$ and $KB \square \Pi K_i \square \perp$)
 $KB \leftarrow KB \square \Pi K_i$;
 $i \leftarrow i+1$;

return KB
end

当这个算法结束时, 最后的 ΠK_i 所对应的优先级就是所得到的 KB 的非一致性等级, 即 $\text{Inc}(KB)=PR(\Pi K_i)$, 也就是说 $\Pi K_1 \cup \Pi K_2 \cup \dots \cup \Pi K_i$ 是非一致性。当算法结束时, $i=n+1$ 时, 这意味着整个 ΠK 都是一致性的。这个算法的复杂度是 $O(n)$ 的, 这里的 n 代表了优先级本体知识库 ΠK 中的层数, 而运用二分查找法的话, 可以将算法的复杂度提高到 $O(\log_2 n)$ 。算法以一种直感有效的方法摒弃了知识库中优先级较低的非一致性本体, 从而选取出本体库中的一致性子集, 保证了查询的正确性。

但是这种方法有一个很大的缺陷, 它直接将引起非一致的优先级较低的本体层不作任何区分而予以忽略。但是在实际的应用中, 往往引起冲突的本体规则只是这些低优先级本体层中的很小一部分, 而其他更多的本体知识, 即使不会引起知识库的冲突, 也因为其较低的优先级而同时遭到了抛弃, 这样就大大减少了本体库中的本体数量, 导致本体知识库的描述和推理范围受到人为的削弱。因此, 考虑将算法进行改进, 在发生冲突时并不马上终止, 而是将剩下未冲突的本体知识仍旧纳入到最后的一致性本体知识库中。

3.2 最大一致性算法

冲突即终止算法在本体库发生冲突时就停止了对剩余本体的考察, 因此它导致了很大一部分低优先级的一致性本体无法加入到知识库中。而下面要讨论的最大一致性构造算法与冲突即终止算法的最大差异就在于当不一致发生时, 并不马上将当前以及后续的本体层予以抛弃, 而是按照优先级逐层对本体库进行探究, 将没有涉入冲突的所有本体都一一加入到最后的本体知识库中。显然它比前面的算法保留了更多的本体知识信息, 而且也有效地避免了一致的产生。

算法 2 最大一致性优先级本体知识库构造算法

input: 分层的优先级知识库 $\Pi K=(\Pi K_1, \Pi K_2, \dots, \Pi K_n)$
output: 最大一致性本体知识库 KB
begin
 $KB \leftarrow \emptyset$;
 for $i \leftarrow 1$ to n do
 if ($KB \square \Pi K_i \square \perp$) then
 $KB \leftarrow KB \square \Pi K_i$;
 else
 $CS \leftarrow \Pi K_i \cap X(KB \square \Pi K_i)$;
 $KB \leftarrow KB \square (\Pi K_i - CS)$;
 $i \leftarrow i+1$;
 return KB
end

可以看到, 算法中首先还是以优先级为基本的判定准则, 按照优先级的高低来一一对本体知识库的各个层次来进行考察。对于与前面已有知识库相一致的本体库层, 可以直接将它们放入新的知识库中。而对于那些与已有知识库存在冲突的本体库层, 也不是一味地予以抛弃。由于这个新的知识库层比已有的知识库的优先级要低, 于是舍弃了新层次中涉入冲突的那些本体, 而将剩下的不会引起不一致的本体同样放入新的知识库内。一方面保证了每次加入到本体库中的本体比知识库中已有的本体优先级更低, 而另一方面, 又保证了新加入的本体不会给知识库引入不一致性。换句话说, 算法在舍弃一定的本体时, 既保证了所舍弃的本体信息必定会带来冲突, 又保证了所舍弃的本体知识优先级较低。这样, 尽可能确保了所得到的本体知识库的一致性和完备性。

结论 本文在本体知识库中引入了优先级这个概念, 通过优先级来解决本体的不一致性冲突, 从而找到一条解决超协调本体推理的有效途径。在今后的工作中, 我们将进一步

考察以上算法在实际使用中的效率性,以及探讨超协调本体推理在非单调推理的推广。

参考文献

- 1 Schlobach S, Cornet R. Non-standard reasoning services for the debugging of description logic terminologies. In: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI'03), Acapulco, Mexico, August 2003. 355~362
- 2 董明楷,蒋运承,史忠植. 一种带缺省推理的描述逻辑. 计算机学报, 2003, 26(6): 729~736

- 3 Huang Z, Harmelen F, Teije A. Reasoning with Inconsistent Ontologies. In: Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence (IJCAI'05), Edinburgh, Scotland, August 2005. 454~459
- 4 Benferhat S, Garcia L. Handling locally stratified inconsistent knowledge bases. *Studia Logica*, 2002. 77~104
- 5 Lang J, Marquis P. Removing inconsistencies in assumption-based theories through knowledge-gathering actions. *Studia Logica*, 2001. 179~214
- 6 Marquis P, Porquet N. Resource-bounded paraconsistent inference. *Annals of Mathematics and Artificial Intelligence*, 2003. 349~384

(上接第 105 页)

结果也是推理控制的关键。将节点封装为对象后,只需在历史记录中记载已访问对象的标识,这样可以减少存储负担。

3.3 推理控制

推理控制可以在设计和查询两个阶段实施。设计阶段可以根据敏感对象的反向推理链,确定引发推理泄露的对象,进而通过授权调整来消除泄露。但是,授权调整是以牺牲数据可用性为代价的,对于聚合敏感问题,基于信息最大可用的原因,授权调整并不能有效解决,只能在查询阶段实施控制,这样才能在保证文档安全同时,提高数据可用性。

过程:查询阶段的推理控制

输入:用户 u 的历史记录 O ,显式拒绝集 O' ,当前查询结果 O'' ,对象关系集 R

输出:允许或拒绝查询,新的历史记录

方法:

1. $O_1 = O \cup O'$, 根据算法 1, 根据 R 计算 O_1^+
2. 检查推理泄露: 对 $\forall o \in O'$, 若 $o \in O_1^+$, 则拒绝查询并返回 O

3. 允许查询并返回 O_1

例 6 仅在授权 AU02 下针对用户 US01 的安全推理控制。

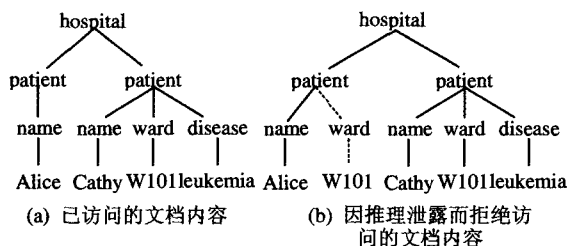


图 3 XML 文档安全推理控制示例

图 3(a)是 US01 当前符合授权策略的已访问内容,对象 SO01 包含其中;当其试图访问(b)中虚线指示的节点时(即对象 SO04; (SO04, locate: ward/W101, type: SimpleObject)),使用上述过程根据关系 R01,可以推理出与 SO01 具有公共根节点的对象 SO02,从而得到敏感的组合对象 AO01。故控制系统将拒绝 US01 对 Alice 所属病房信息的查询,保证敏感信息不被推理泄露。

4 系统框架及实现

基于 RDF 的 XML 文档推理控制的体系结构如图 4(1~9 为运行阶段控制流程;I~II 为安全设计)所示,主要由三大模块组成,分别为查询引擎、访问控制和推理控制。

我们在达梦 SDM4 数据库管理系统的 XML 数据库上实现上述系统。该数据库采用面向节点授权的访问控制机制,支持满足 XPath 规范的路径查询。推理控制系统在此基础

上实施。在设计阶段,完成节点封装、对象关系指定和授权,最终形成给定 XML 文档的安全策略。在查询阶段,用户提交的查询经由查询引擎得到查询结果,先由访问控制模块根据安全策略检查授权合法性,若违反则拒绝响应,否则将结果提交给推理控制模块。后者先将结果与历史记录进行临时归并,然后利用对象关系对归并结果进行推理扩展,检查是否存在推理泄露以决定对用户的响应。如果不存在推理泄露,在响应用户之前还需更新原有历史记录。

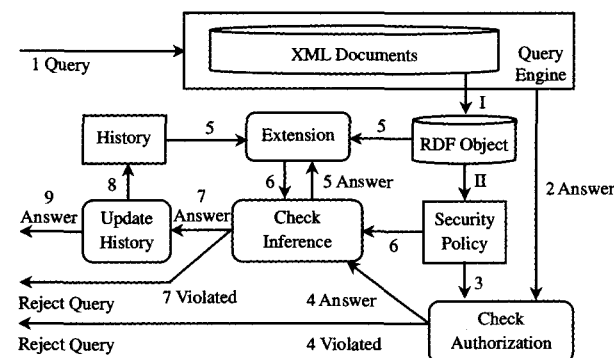


图 4 XML 推理控制系统体系结构

总结 本文在 RXACL 的基础上进一步通过 RDF 刻画 XML 文档中节点间的语义关系,提出了基于 RDF 的 XML 安全推理控制方法。通过 XML 类型的引入扩展了 RXACL 模型,简化了复杂的授权过程;以对象标识形式存储的历史记录可以降低存储负担,同时也避免了复杂的文档归并过程。后续的研究将致力于解决文档间的关联推理问题。

参考文献

- 1 Yang XiaoChun, Li Chen. Secure XML Publishing without Information Leakage in the Presence of Data Inference. In: Proc. of the 30th VLDB Conference, 2004. 96~107
- 2 Tan Z J, Pang Y M, Shi B L. Reasoning about functional dependency for XML. *Journal of Software*, 2003, 14(9): 1564~1570
- 3 Farkas C, Stoica A. Correlated data inference in ontology guided XML security engine. In: Proc. of the 17th WG 11.3 working conference on Data and Application Security, 2003
- 4 Gowadia V, Farkas C. RDF metadata for XML access control. In: Proc. of the 2003 ACM workshop on XML security, 2003. 39~48
- 5 Consortium W W W. Primer RDF. W3C Recommendation. Available at: <http://www.w3.org/TR/2004/REC-rdf-primer-20040210>, 2004
- 6 Damiani E, Vimercati S D C, Paraboschi S, et al. A fine-grained access control system for XML documents. *ACM TISSEC*, 2002, 5(2): 169~202
- 7 Bouganim L, Ngoc F D, Pucheral P. Client-Based Access Control Management for XML documents. In: Proc. of the 30th VLDB Conference, 2004. 84~95
- 8 Fan Wenfei, Chan Chee-Yong, Garofalakis M. Secure XML Querying with Security Views. In: Proc. of the 2004 ACM SIGMOD international conference on Management of data, 2004. 587~598
- 9 Denning D E. A Preliminary Note on the Inference Problem in Multilevel Database Management Systems. In: Proc. of the National Computer Security Center Invitational Workshop on Database Security, 1986