用户子本体的提取与简化

李 旸,李启炎
LI Yang,LI Qi-yan

同济大学 CAD 研究中心,上海 200092
CAD Research Center,Tongji University,Shanghai 200092,China
E-mail:ly.china@gmail.com

LI Yang,LI Qi-yan.Extraction and simplification of user sub-ontology.Computer Engineering and Applications,2008,44(36):171-173.

Abstract: With wider and deeper application of the ontologies,ontologies tend to grow bigger and bigger.For specific users or agents,they may be interested in some subsets of an ontology.It is unnecessary for them to operator the huge ontology.So,there is high requirement that user can extract sub-ontologies from an ontology base.According to the shortages of previous research,this paper proposes a user sub-ontology extraction and simplification method that can protect hierarchical relationship,output sub-ontologies without any extra user interference,and reduce the workload of users.

Key words: ontology;sub-ontology;ontology extraction

摘 要:随着本体的应用日益广泛和深入,本体的规模也会变得越来越大。特定用户往往只对本体的某个部分子集感兴趣,没有必要操作巨大的本体,因此,从本体库中提取出用户感兴趣的子本体的需求非常迫切。针对已有研究的不足,提出了一种能够保存本体层次关系的用户的子本体提取和简化方法,方法只需要用户给出的兴趣概念集即可自动输出子本体,不需要更多的人工干预,节省了用户工作量。

关键词:本体;子本体;本体提取
DOI:10.3778/j.issn.1002-8331.2008.36.048 **文章编号:**1002-8331(2008)36-0171-03 **文献标识码:**A **中图分类号:**TP311

1 引言

近年来,在计算机科学中关于本体的研究和应用越来越多。所谓本体,最著名并被广泛引用的定义是由 Gruber 提出的"本体是概念模型的明确的规范说明"[1]。通俗地讲,本体是用来描述某个领域以至更广范围内的概念以及概念之间的关系,使得这些概念和关系在共享的范围内具有大家共同认可的、明确的、唯一的定义,这样,人与机器之间以及机器与机器之间就可以进行一种可以自动处理的信息或知识的交流。本体为特定领域内应用系统的设计提供可共享的概念体系,为异构的数据源提供互操作机制。目前,本体已经被广泛应用于语义 Web、智能信息检索、信息集成、数字图书馆、自然语言处理等领域[2]。

随着本体应用的深入以及知识时代知识信息的快速膨胀,人们面对的本体也会变得越来越大、越来越复杂。使用这样庞大的本体,对很多用户来说,将会有比较多的困难。事实上,在大部分的时候,用户关心的只是大型本体中的一小部分本体概念,因此,如果可以得到一个更小的、包含了用户所需要信息的本体,就可以更好地帮助用户来使用本体。

2 相关研究

对子本体的提取已经有不少的研究。Spyn 等人在文献[3]中说明了子本题提取的必要性,同时 Bhatt 等人在文献[4-5]中描述了一种分布式本体提取的方法。但是文献[4-5]使用的方法,依赖于一个领域内的本体工程师,这个专家必须完全了解领域本体。首先,本体工程师根据用户的需求,把本体内的概念和属性标记为必选概念和必选属性,然后使用一个自动化机制,挑选出这些概念间的所有可能路径。最后根据选择的概念和属性,生成一个子本体。这个方法不仅需要人的介入,还需要一个完全了解领域本体的专家。李炜等[6]提出一个子本体提取的方法,该方法只需要用户给出感兴趣的概念,就可以自动提取子本体,但是,该方法并没有考虑也不能很好处理本体中很重要的关系 IS_A 关系,因而丢失了重要的层次关系。本文对以上方法进行了改进,既不需要用户太多的干预,同时,很好地保留了概念间的层次关系,并利用传递性关系性质,对子本体视图进行了进一步的简化。

3 用户子本体的提取方法

定义 1[6] 本体是一个二元组 $O<C,R>$。其中,C 表示概念的集合,R 表示概念之间关系的集合。若概念 c_1 到概念 c_2 具有关系(Relation),则记为:$r:c_1\rightarrow c_2$,其中称 c_1 为 r 的源概念,c_2 为 r 的目标概念。

因此可以用有向图来表示本体。其中有向图的顶点集由本体的概念集构成,边集由关系构成,边的方向从源概念指向目

作者简介:李旸(1972-),男,讲师,主要研究领域为智能 CAD,企业信息化;李启炎,男,教授,博导。
收稿日期:2008-08-27 **修回日期:**2008-11-03

标概念。

定义 2 对本体 $O<C,R>$,如果对概念 $c,c'\in C$, $\exists r_1:c\to c_1$, $r_2:c_1\to c_2,r_3:c_2\to c_3,\cdots,r_n:c_n\to c'\in R$,则概念 c 到 c'是可达的,其中 $r_1\ r_2\cdots r_n$ 称为概念 c 与 c'具有的虚关系。

定义 3 本体 $O<C,R>$和本体 $O'<C',R'>$, 若满足:$C'\subset C$, $R'\subset R$,

$\forall r\in R',r:c_1\to c_2$,有 $c_1\in C',c_2\in C'$;

$\forall r\in R,r:c_1\to c_2$,若有 $c_1\in C',c_2\in C'$,那么 $r\in R'$。

则称本体 O'是本体 O 的子本体。

以图 1 所示的本体 O 为例,假设用户感兴趣的概念集为 $S\{A,E\}$,那么其他的概念(如 $H,I,J,L\cdots$)就相对不再重要。这时,希望可以得到一个尽量小的本体 O',使得这个 O'包含了概念集 S,并且 O'所包含的语义信息也应该在 O 中。即:根据本体 $O<C,R>$和用户给定的概念集 S,生成一个本体 $O'<C',R'>$,其中 $S\subset C'\subset C,R'\subset R$。

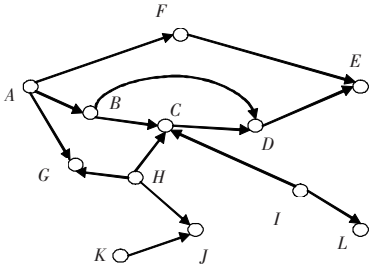

图 1 一个假想的本体 O

定义 4 给出概念集 S 和本体 $O(C,R)$及其子本体 $O'(C',R')$,若 $S\subset C'$,对于任意 $c_1\in C-S,c_2\in S$,满足:

$\forall r_1:c_2\to c_1\in R'$, $\exists c_3\in S$,使得 $r_2:c_1\to c_3\in R'$;

$\forall r_1:c_1\to c_2\in R'$, $\exists c_3\in S$,使得 $r_2:c_3\to c_1\in R'$;

$\forall c_1\in C,c_3\in S$, 若 c_3 到 c_1 及 c_1 到 c_2 在本体 O 上是可达的,则有概念 $c_1\in C'$;

$\forall r:c_4\to c_5\in R$,若 $c_4,c_5\in C'$,则有关系 $r\in R'$。

则称本体 O'是由概念集 S 在本体 O 上生成的子本体, 其中概念集 S 称为源概念记为 Co,$C-S$ 称为隐含概念,记为 Ch。在本体图中(图 2),源概念用黑点表示,隐含概念用圆圈表示。

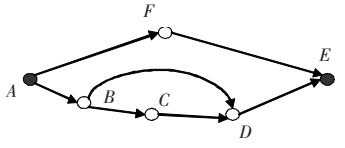

图 2 子本体的例子

上述的分析并未专门考虑特殊的 IS_A 关系。事实上,本体中最常使用的也是最重要的关系 IS_A 关系, 由于具有特殊的性质,可以专门进行处理。一般来说,子本体需要保留相对完整的层次关系,这样用户看上去比较清晰。下面先定义共同祖先概念:

如果概念 c_1、c_2 之间存在 is_a 关系:$Is\_a:c_1\to c_2$,则称 c_2 为 c_1 的父概念,c_1 为 c_2 的子概念。

若概念 c_1、c_2 之间存在导出关系 $r_1r_2\cdots r_n:c_1\to c_2,n\geqslant 1$,且 $r_1r_2\cdots r_n$ 均为 Is_a 关系,则称 c_2 为 c_1 的祖先概念。c_1 到 c_2 之间的 is_a 的关系数称为 c_1 与 c_2 的层次距离,记为 $Ld(c_1,c_2)$。若有一组概念 $c_i\in S,\forall c_i\in S,ac$ 是所有 c_i 的祖先概念,称 ac 为这组概念的共同祖先概念。对本体 $O<C,R>,S\subset C,A$ 为 S 的共同祖先概念集合,若 $ac\in A$,使 $\sum_{c_i\in S,ac\in A}Ld(c_i,ac)$为最小,则称 ac 为 S 的最近共同祖先概念。

很明显,概念集 S 生成子本体,如果需要获得一个最小的完整的层次关系,则必定要包含 S 的最近共同祖先概念 ac。所有 S 集合的元素存在到 S 的最近共同祖先概念 ac 节点的 is_a 的各种长度的虚关系,但是,反过来,则不一定存在虚关系,通过对上面的生成的子本体定义分析,ac 不符合生成的子本体的定义。因此,获得一个完整的层次关系,需要对最近共同祖先概念 ac 进行特殊处理。

对于概念集 S,首先求出其最近共同祖先概念 ac,并将 ac 作为一个特殊节点加入 S 得到 $S'=S\cup\{ac\}$,用 S'作为输入概念集,则可以得到带有层次结构的子本体。

下面根据上面的讨论,给出了根据概念集 S 生成层次结构的子本体的算法。

算法 1 层次结构的子本体的生成算法。

输入:用户感兴趣的概念集 S,本体 $O(C,R)$。

输出:由概念集 S 在本体 $O(C,R)$上生成的层次结构的子本体 $O'(C',R')$。

步骤:设 ROOT 为概念层次的根。

任取 $s_1\in S$,将 s_1 到 ROOT 的元素一次加到表 pl 中,pl 长度为 n,$pl[1]=s_1$,$pl[n]=$ROOT

$\forall Si\in S-\{s_1\}$,$p=Si$,反复执行检查 p 在 pl 的出现,并将 p 的父概念赋予 p,直至 p 在 pl 中出现,并记住其在 pl 中的标号 t_{max}

当 S 中元素循环一遍后,最大的 t_{max} 所代表的元素 $pl[t_{max}]$赋予 ac

$S=S\cup\{ac\}$

对 $O(C,R)$进行深度优先遍历,取 $\forall c_i\in S$,进行深度优先遍历,记录所有遍历到的点为 G。遍历完成后,看是否有 S 中的点没有遍历到,若存在 $c_i\in S,c_i\notin G$,则以 c_i 为起点,进行深度优先遍历,把遍历到的点添加到 G 中,直到任意 $c_i\in S$,都有 $c_i\in G$。

对 $O(C,R)$的反图 $O''(C,R'')$重复上面的过程,把过程中遍历到的所有点集记为 G'。

$G\cap G'$作为生成子本体的概念集 C'

$\forall r\in R$,若 r 的源概念和目标概念都属于 C',则把 r 加入到 R'中

输出子本体 $O'(C',R')$

4 用户子本体简化方法

在上例中,虽然得到了本体 O,但是对于用户,不一定需要 O 上的所有关系。例如:从他对领域内知识的理解看来,对于虚关系 $r_{AB}\ r_{BC}\ r_{CD}$ 所表示的语义可以由关系 r_{AD} 代替(如图 3 所示,虚线表示导出关系)。

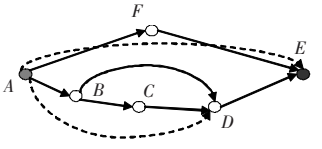

图 3 由图 1 所示本体和概念集 $S\{A,E\}$ 得到的导出本体

定义 5 导出关系:对于 $r\in R,r:c_1\to c_{n+1}$ $n>1$,若在概念 c_1 和 c_{n+1} 之间存在虚关系$(r_1\cdot r_2\cdots r_n):c_1\to c_{n+1}$,且虚关系和关系 r在语义上相同, 则 r 称为在本体 O 上概念 c_1 到 c_{n+1} 的导出关系,记为 $r:c_1\to c_{n+1}$,其中 n 称为导出关系的长度。

定义 6 本体 $O<C,R>$和在 O 上的导出关系集 R_p 一起,构成了一个新的本体 $O'<C,R'>$,其中 $R'=R_p\cup R$,称本体 O'为 O

的一个导出本体。

可以看出,通常用户所习惯使用的概念,往往是由这些导出关系所定义的。而根据这些导出关系,就可以对由概念 S 生成的子本体进行简化。例如图 3 所示的本体。

导出关系 $r_{AE}(r_{AF}\,r_{FE})$可以在语义上代替虚关系 $r_{AF}\,r_{FE}$,并且概念 F 不是用户所给定的概念,因此,就可以在本体中把概念 F 去掉,在概念 A 到概念 E 中添加一条边 r_{AE}。

导出关系 $r_{AD}(r_{AB}\,r_{BC}\,r_{CD})$虽可代替虚关系 $r_{AB}\,r_{BC}\,r_{CD}$,但是,在 A 到 E 中还存在一条虚关系 $r_{AB}\,r_{BD}\,r_{DE}$,其中,根据图 2 所示,r_{BD} 不是 $r_{BC}\,r_{CD}$ 的导出关系,因此 r_{BD} 不能代替 $r_{BC}\,r_{CD}$。这时,就必须保留关系 r_{AB}。因此,得到如图 4 所示的本体。

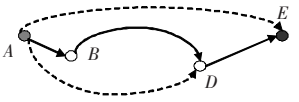

图 4　图 3 所示本体的化简步骤

定义 7　在导出本体 $O<C,R\cup P>$上对某两个概念 $c_1,c_2\in C$,若 c_1 到 c_2 上存在虚关系 $r_1\,r_2\cdots r_i\cdots r_m\cdots r_n$,如果存在导出关系 $r':r_i\cdots r_m$,那么称虚关系可以被导出关系 r'简化。

定义 8　本体 $O<C,R>$、导出关系集 P 和由概念集 S 生成的本体 $O'<C',R'>$,如果满足下面条件:

$S\subset C$;O 是本体 $O'<C',R'\cup P>$的子本体;

$\forall c_1,c_2\in C$,如果在 c_1、c_2 中存在虚关系 $r_1\ r_2\cdots r_i\cdots r_m\cdots r_n$ 可以被导出关系 $r':r_i\cdots r_m$ 所代替,那么对 $r_k(i\leqslant k\leqslant m)$,$\exists c_3\in c-\{c_1,c_2\}$,在概念 c_1 到 c_3,或 c_3 到 c_2 中存在不能被简化的虚关系包含 r_k;

$\forall c_3\in C$,$\exists c_1,c_2\in C$,使得 c_1 到 c_3 是可达的,c_3 到 c_2 是可达的,那么称本体 O 是根据概念集 S 在导出本体 O'上的最简子本体。

根据导出关系进行的子本体的一般化简算法可参见文献[6]。导出关系由于中间间隔了很多个概念,很多情况下不是很有意义,同时要化简导出关系,首先要定义导出关系,其实用性不是很高。从另一个角度,本体中最常使用的也是最重要的关系 IS_A 关系,由于具有特殊的性质,可以专门进行处理。一般通过 IS_A 关系可以将本体概念组成一个层次结构,且满足传递性、属性继承、性质继承等特性。

其中具有传递性的关系明显容易简化,传递性的一般描述为:如果一个关系 R 具有传递性,那么对于任意的 x、y 和 z:

$$R(x,y)\wedge R(y,z)\rightarrow R(x,z)$$

同时,很多关系都满足传递性,如 part_of(部分关系)等。

对于特殊的有一组单一的满足传递性的关系组成的导出关系:对于若在概念 c_1 和 c_{n+1} 之间存在虚关系$(r_1\cdot r_2\cdots r_n):c_1\rightarrow c_{n+1}$,由于 $r_1\cdot r_2\cdots r_n$ 都是关系 r,且 r 满足传递性,即 $r:x\rightarrow y\wedge r:y\rightarrow z$ 蕴含 $r:x\rightarrow z$,容易得出存在关系 $r\in R$,$r:c_1\rightarrow c_{n+1}$,且虚关系和关系 r 在语义上相同。根据这个原理,构造子本体简化算法。

算法 2　由概念集 S 生成的子本体的简化算法。

输入:概念集 S,由概念集 S 在本体 $O(C,R)$上生成的子本体 $O_1(C_1,R_1)$。

输出:简化的子本体 $O_2(C_2,R_2)$。

步骤:将 C_1 复制到 C_2,R_1 复制到 R_2

遍历 C_1 所有概念节点,对于所有隐含节点 tc,即属于 C_1 但不属于 S 的概念节点,若 tc 相关的关系满足关系对:$r:ca\rightarrow tc$,$r:tc\rightarrow cb$,r 为传递性关系,且 tc 没有其他相关关系,则依次在 R_2 中用 $r:ca\rightarrow cb$ 替代关系对 $r:ca\rightarrow tc$,$r:tc\rightarrow cb$,并在 C_2 中删除节点 tc。

输出子本体 $O_2(C_2,R_2)$。

5　总结

本文在分析已有的关于子本体的提取的研究分析的基础上,对已有的方法进行了改进,提出了一种能够保存本体层次关系的用户的子本体提取和简化方法,该方法只需要用户给出的用户感兴趣的概念集,就可以自动输出相应的子本体,不需要更多的人工干预,节省了用户工作量,同时,子本体中最大地保存了原有的本体层次结构。同样,根据本体概念间 IS_A 关系的传递性对子本体进行了简化。然而,由于本体应用的复杂性,关于子本体的提取和简化,用户将有更多的个性化的需求需要考虑和研究,这也是下一步研究工作的方向。

参考文献:

[1] Gruber T R.A translation approach to portable ontology specifications,KSL 92-71[R].Knowledge System Laboratory,1993.

[2] Deng Z H,Tang S W,Zhang M,et al.Overview of ontology[J].Acta Scientiarum Naturalium Universitatis Pekinensis,2002,38(5):730-738.

[3] Spyns P,Meersman R,Mustafa J.Data modelling versus ontology engineering[C]//SIGMOD,2002.

[4] Bhatt M,Flahive A,Wouters C,et al.A distributed approach to sub-ontology extraction[C]//AINA'04,2004-03.

[5] Wouters C,Dillon T,Rahayu W,et al.A practical walkthrough of the ontology derivation rules[C]//DEXA2002,2002-08.

[6] 李炜,马俊,吴宇进,等.基于用户概念视图的本体约减[J].计算机工程,2005,31(24).

(上接 170 页)

[2] Bayardo R.Efficiently mining long patterns from databases[C]//Haas L M,Tiwary A.Proc of the ACM SIGMOD Int'l Conf Management of Data.New York:ACM Press,1998:85-93.

[3] Lin D,Kedem Z M.Pincer-Search:A new algorithm for discovering the maximum frequent set[C]//Schek H J,Saltor F,Ramos I,et al.Proc of the 6th European Conf Extending Database Technology.Berlin:Springer-Veriag,1998:105-119.

[4] 宋余庆,朱玉全,孙志挥,等.基于 FP-tree 的最大频繁项目集挖掘及更新算法[J].软件学报,2003,14(9):1586-1592.

[5] 吉根林,杨明,宋余庆,等.最大频繁项目集的快速更新[J].计算机学报,2005,28(1):128-135.

[6] 陈耿,朱玉全,杨鹤标.关联规则挖掘中若干关键技术的研究[J].计算机研究与发展,2005,42(10):1785-1789.

[7] 王丽珍,周丽华.生成频繁项目集的一种贪心算法[J].计算机工程与应用,2001,37(13):86-88.