一种基于本体的异构数据源模式集成

于 琦,周 勇
(大连理工大学 软件学院,辽宁 大连 116620)

摘 要:本体是概念模型的明确的规范说明,能够精确地描述概念体系和领域知识。为了将异构数据源中的数据识别出来并进行语义相关的集成,提出了一种基于本体集成异构数据源的方法。首先将各个数据源中的数据以XML文档形式进行描述,然后将各个XML文档的文档类型定义(DTD)转化为DIM数据模型表示,最后通过语义聚类、全局模式生成等步骤,实现XML文档的基于本体的语义集成。文中提出的方法以普林斯顿大学的心理学家、语言学家和计算机工程师联合设计的一种基于认知语言学的英语词典为本体库,可有效地识别出异构数据源中的具有等价语义或相近语义的数据,从而更准确地对异构数据源中的数据进行集成。

关键词:本体;异构数据源;模式集成;WordNet
中图分类号:TP311.13 **文献标识码**:A **文章编号**:1673-629X(2008)02-0034-04

An Ontology-Based Schema Integration of Heterogeneous Data Sources

YU Qi, ZHOU Yong
(Software School, Dalian University of Technology, Dalian 116620, China)

Abstract: An ontology is an explicit specification of a conceptualization, which could represent the conceptualization and domain knowledge more clearly. In this paper an approach of ontology-based integration of heterogeneous data source is proposed to recognize and integrate the data in the heterogeneous data source. Firstly, every data source is described as XML documents, and then each document type definition (DTD) of the XML documents is converted into a data model called DIM, finally the integrated XML document could be got through several steps, such as semantic clustering, global schema generate and so on. The method proposed in this paper is based on the electric English dictionary designed by the psychologists, linguists, computer engineers of Princeton University, which could recognize the data that contains the same or similar semantic and integrate the heterogeneous data source more accurate.

Key words: ontology; heterogeneous data sources; schema generate; wordNet

0 引 言

随着计算机及互联网的快速发展,网上的数据每天都在以指数级的速度增长,从而形成了一个巨大的异构数据库,同时随着各企业及其相关部门信息化工作的进一步深入,数据的存放也日益分散,这就造成了成千上万个异构数据源的存在。这些异构的数据源,就像是相互独立的"信息孤岛",使数据源之间的互操作变得复杂、困难。企业内部各个部门的"信息孤岛"虽然表面上相互分离,但是,各个部门分散的数据之间也存在着各种各样的联系。作为企业的主管部门要从全局角度了解分析数据,从而为企业的发展制定方针政策,就必须要面对企业内部各部门数据分散的问题。同时,如果各个部门相互间能够做到数据共享,就可以更好地提高工作效率,因此异构数据源数据集成的工作迫在眉睫。异构数据源集成通过分析企业各部门中的"信息孤岛",从而找出它们之间潜在的内部联系,并对这些"信息孤岛"进行集成,为数据的管理与共享提供了方便。

文中介绍数据集成中几种常见的方法;描述了文中数据集成模型所用到的一些概念和技术;阐述了数据集成模型的创建过程及执行算法。

1 常见数据集成方法

目前,在开发针对异构数据集成系统时所采用的体系结构虽各不相同,但其基本机构可分为四类:数据仓库法、中介器法、联邦数据库法和基于语义的方法。

1.1 数据仓库法

将来自几个数据源的数据副本,按照一个集中、统一的视图要求,进行预处理、转换,以符合数据仓库的

收稿日期:2007-05-09
作者简介:于 琦(1982-),男,辽宁阜新人,硕士研究生,主要研究方向为数据挖掘、数据集成;周 勇,副教授,研究方向智能计算与数据挖掘、软件质量与测试技术。

模式,并存储到数据仓库中[1]。这种方法既可以用于数据集成,又便于进行联机分析和数据挖掘。缺点是当数据源中的数据发生变化时,数据仓库中的数据也要做相应的修改,因此会造成数据更新不及时、数据的重复存储。

1.2 中介器法

该方法使用与数据仓库方法完全不同的结构。数据仍然保存在各异构数据源上,集成系统仅提供一个虚拟的集成视图和对该视图的查询的处理机制。系统自动将用户对集成视图的查询请求转换成对各异构数据源的查询并将结果返回给用户[2]。

1.3 联邦数据库法

数据源相互独立,但通过数据源之间的数据交换格式进行一一映射,一个数据源可以访问任何其他数据源提供的信息[2]。这种方法的优点是容易实现,而缺点则是工作量极大,扩展性差。

1.4 基于语义的方法

该方法将数据中具有相同概念或者含义的数据进行归类,从而将数据表示成若干个类及其相互关系的集合。文中提出的数据集成模型就是基于这种方法。

2 本体与 XML

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"[3],通俗地讲,本体就是用来描述某个领域甚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,人机之间以及机器之间就可以进行交流。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、数据图书馆等领域[4]。

WordNet[5]是 Princeton 大学的心理学家、语言学家和计算机工程师联合设计的一种基于认知语言学的英语词典。它不是光把单词以字母顺序排列,而且按照单词的意义组成一个"单词的网络"。鉴于 WordNet 中词汇量的丰富,可以将其作为一个参考的本体集合。

XML 是由 SGML (Standard Generalized Markup ,标准化通用标记语言)发展而来,是 SGML 的一个简化子集,它以一种开放的自我描述方式定义数据结构,在描述数据内容的同时能突出对结构的描述,从而体现出数据之间的关系,因此使用 XML 作为数据的统一表示,并采用文献[6]中提到的文本文件形式进行存储。

文中提出的数据集成模型,以普林斯顿大学开发的电子辞典 WordNet 作为本体,通过数据转换模块完成异构数据源到 XML 文档的转换,并对数据转换模块输出的 XML 文档进行集成。由于数据转化不是文中讨论重点,详细内容请参阅文献[7]。

3 数据集成模型实现

首先将代表 XML 文档模式信息的 DTD 转化成系统定义的 DIM 数据模型,然后通过语义聚类、全局模式生成、XML 集成文档生成三个步骤,完成数据集成。图 1 是该数据集成模型的结构图。

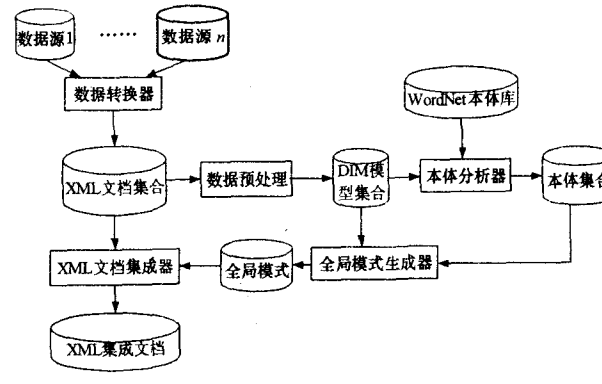

图 1　数据集成模型

3.1 DIM 数据模型

一个 DTD 文档由元素、属性组成。元素可以有子元素和属性,子元素可以以列表的方式或者选择的方式组织。将 DTD 转化成 DIM(Data Integration Model)模型,每个 DIM 模型对应一个 DTD,DIM 是一个三元组 $< E, A, R >$,其中,E 代表元素集合,A 代表属性集合,R 代表元素关系集合。这里把 DTD 看成由元素组成的一个树,树中的每个节点都是 DTD 中的元素,而属性作为节点内部构成部分,树中的每两个节点之间的关系存储在 R 中。

每个 $e \in E$ 包括五个属性<label, nodeStructure, contentType, cadinality, attlist>,其中,label 表示该元素名称;nodeStructure∈{AND, OR, LEAF}表示该元素的子元素的组织方式;contentType∈{EMPTY, #PCDATA, ANY, COMPLEX(表示该元素含有嵌套的子元素)}表示该元素的内容类型;cadinality∈{?、+、*、Null}表示该元素的基数约束;attlist 表示该元素的属性集合。

每个 $a \in A$ 包括三个属性{label, contentType, cadinality}, label 表示该属性名称; contentType ∈{CDATA, Emumerated, ID, IDREF, IDREFS, NMTOKEN, NMTOKENS, ENTITY, ENTITIES, NOTATION}表示该属性的内容类型; cadinality ∈{IMPLIED, REQUIRED, FIXED, Default}表示该属性的基数约束。

每个 $r \in R$ 是一个二元组 $< e1, e2 >$,表示元素

$e1$ 是元素 $e2$ 的父节点。

因此,根据上面的转换规则,可将 DTD 转换为 DIM 数据模型,例如下面的例子:

```
<! ELEMENT Employee( name,department,contact)
<! ATTLIST Employee no #CDATA #REQUIED>
<! ATTLIST Employee manager #CDATA #IMPLIED>
<! ELEMENT contact ( home|mobile) >
<! ELEMENT department(department_name,address) >
<! ELEMENT name( #PCDATA) >
<! ELEMENT home( #PCDATA) >
<! ELEMENT mobile( #PCDATA) >
<! ELEMENT department_name( #PCDATA) >
<! ELEMENT address( #PCDATA) >
```

DIM 模型见表 1。

表 1 DIM 模型

E	Employee	And	COMPLEX	Null	no,manager
	contact	Or	COMPLEX	Null	
	department	And	COMPLEX	Null	
	name	LEAF	#PCDATA	Null	
	home	LEAF	#PCDATA	Null	
	mobile	LEAF	#PCDATA	Null	
	department - name	LEAF	#PCDATA	Null	
	address	LEAF	#PCDATA	Null	
A	no	CDATA	REQUIRED		
	manager	CDATA	REQUIRED		
R	r1	<Employee,name>			
	r2	<Employee,department>			
	r3	<Employee,contact>			
	r4	<contact,home>			
	r5	<contact,mobile>			
	r6	<department,department - name>			
	r7	<department,address>			

3.2 语义聚类

将每个 DIM 数据模型中的 $e \in E$ 取出,以 WordNet 本体为参考模型,将 E 中具有相同概念或含义的元素进行聚类,并生成新的 E,名称为 new$E$。每个 $e' \in$ new$E$ 包含两个属性{name、list},其中,name 表示该类元素的一个统一的名称,list 表示与该类具有相同含义的元素集合。算法 1 描述了语义聚类过程,首先取出一个 DIM 数据模型,将其中的所有元素放入到图 1 中系统的本体集合中,然后继续遍历其他 DIM 数据模型,取出其中的元素并与系统的本体集合中的元素比较,判断这两个元素是否为语义等价的,如果等价,则将新的元素加入到系统已有的本体元素的 list 中去;如果没有,则向系统中添加新的本体元素。当所有 DIM 遍历后,就可以生成系统需要的本体集合。这个过程与文献[8]中的提到的本体映射过程非常相似,但又不完全相同,因为这里省略了相似度计算及优化等过程。

算法 1:生成系统本体集合。

输入:DIM 集合

输出:系统本体集合

①遍历 DIM 集合,取出一个 DIM 模型,分析该模型中的集合 E。

② 分析集合 E,取出其中的一个元素 e,利用本体分析器进行分析:如果 new$E$ 本体集合为空,将 e 添加到 new$E$ 中,并令 e.name = e.label,e.list[0] = e.label。如果 new$E$ 本体集合非空,若 new$E$ 本体集合中的某个元素e' 与 e 是语义等价的(即 e,e' 具有相同的概念或含义),则 e'.list[n+1] = e.label。若 new$E$ 本体集合中不包含与 e 语义等价的元素,则将 e 添加到 new$E$ 中,并令 e.name = e.label,e.list[0] = e.label。

③ 继续遍历集合 E,若还有元素节点,则返回 ②。若已经为空,则执行 ④。

④ 继续遍历 DIM 集合,若还有 DIM 模型,则返回 ①;若已经为空,则执行 ⑤。

⑤ 返回集合 new$E$。

3.3 全局模式生成

完成上面的所有工作后,进入到最为关键的全局模式生成阶段。首先要创建一个空的 XML 文档,并将该文档的 DTD 根节点命名为 root,该 DTD 即为全局模式的最初形式。然后取出一个 DIM 数据模型,依次遍历其中的元素节点,比较当前元素节点是否与全局模式中某一元素节点语义等价,若存在语义等价的节点,则要进一步比较当前元素节点的父节点与全局模式中与其具有语义等价性的元素的父节点是否同样是语义等价的,只有在各自父节点同样是语义等价的节点时,才能证明当前节点与全局模式中的元素节点是语义等价的。这是因为语义等价的两个节点可能在不同的语境下表示不同的含义,例如同样是 ID 这个元素,但是在 department 和 student 这两个元素下,却表示不同的含义。若当前节点在全局模式中具有语义等价节点,则将节点相关信息添加到与其等价的节点的相应信息中;若当前节点在全局模式中不具有语义等价节点,则在全局模式中添加一个新节点。当所有的 DIM 遍历完毕后,全局模式也随即生成。

算法 2:生成全局模式。

输入:new$E$,各个数据源对应的 DIM

输出:全局模式

① 创建名为 intXML 的新 XML 文档,并设其 DTD 只包含一个节点且为根节点 root。

② 遍历 DIM 集合,取出一个 DIM 模型 dim。

③ 从 dim 中取出一个元素 e,并在 intXML 文档 DTD 中查询是否具有与该元素具有语义等价的元素

节点,若有则执行④;否则,执行⑤。

④比较这两个元素的父节点是否也是语义等价,若等价,则将 e 的子元素、属性添加到该元素的对应位置;否则,执行⑤。

⑤生成对应的元素,并将 e.label 改成 new$E$ 中对应的 name;将 e 具有的属性取出,并在 new$A$ 中找到对应的名字;将 e 在 R 中的对应关系添加到 intXML 中。

⑥继续遍历 dim,若集合 E 已经为空,则执行⑦;否则,执行③。

⑦继续遍历 DIM 集合,若已经为空,则执行⑧;否则,执行②。

⑧返回 intXML 文档。

3.4 XML 集成文档生成

利用 JDOM API 接口分析每个数据源对应的 XML 文档,并根据元素、属性名称与全局模式中的元素和属性的对应关系进行名称转化,然后将对应的数据添加到 intXML 文档中。

4 结　论

数据集成是解决"信息孤岛"问题的根本方法。文中提出了一种基于本体的异构数据源集成方法,将不同数据源转换成对应的 XML 数据,然后通过语义聚类、全局模式生成等步骤实现异构数据源的集成。当异构数据源中具有语义等价的信息时,文中提出的方法可以进行集成,若各个异构数据源没有语义等价的信息时,又可以对这些数据进行简单的重组,使它们都存放到一个 XML 文档中,这是因为在全局模式生成的第一步设置了与各个数据源都无关的 root 根节点,因此该方法还是可行的。但是,由于该方法中使用的 WordNet 本体库只能识别拼写正确的英文单词,所以在数据的命名方面还需要改进。

参考文献:

[1] Xu Xiangzhong. Knowledge-based Intelligent Query Processing System[C]//In: ICYCS 2001. [s.l.]:[s.n.],2001:1048-1051.

[2] Molina H G. Database System Implementation[M]. Englewood Cliffs,NJ:Prentice Hall, Inc,2000:420-452.

[3] Gruber T R. A translation approach to portable ontology specifications[R]. Stanford: Knowledge System Laboratory, Stanford University,1993.

[4] 邓志鸿,唐世渭,张　铭,等 .Ontology 研究综述[J].北京大学学报:自然科学版,2002,38(5):730-738.

[5] Fellbaum C, Miller G. WordNet: an electronic laxical database [M]. [s.l.]: The MIT Press, 1998.

[6] 吴永春.XML 数据存储方法研究及应用[J].计算机技术与发展,2006,16(2):139-141.

[7] 丁月华,杨　敏.基于 xml 的异构数据源集成与交换的实现[J].计算机应用与软件,2006,23(10):134-136.

[8] 李选如,何洁月.语义集成:本体映射方法研究[J].计算机技术与发展,2007,17(2):121-124.

(上接第 33 页)

4 结束语

全局 Pfair 公平调度是目前分布式实时系统中较理想的调度算法之一,PD^2 调度算法因为其卓越的性能在越来越多的实时任务调度中被采用。文中基于 Linux 内核实现 PD^2 调度算法是一种有益的尝试,目前国内实现的还较少,由于分布式系统和 Linux 内核本身的复杂性,作为一种解决方案还需要大量工作要做。

参考文献:

[1] Liu C L, Layland J W. Scheduling Algorithms for multiprogramming in a Hard-Real Time Environment[J]. Journal of the ACM,1973, 20(1):46-61.

[2] Tanenbaum A S, Van Steen M. Distributed Systems Principles and Paradigms [M]. Beijing: Tsinghua University Press, 2002.

[3] Baruah S, Cohen N, Plaxton C G, et al. Proportionate progress: A notion of fairness in resource allocation[J]. Algorithmica,1996(15):600-625.

[4] Baruah S, Gehrke J, Plaxton C G. Fast scheduling of periodic tasks on multiple resources[C]//In Proc. of the 9th Int'l Parallel Processing Symp. Washington: IEEE Computer Society,1995:280-288.

[5] Anderson J, Srinivasan A. Mixed Pfair/ERfair scheduling of asynchronous periodic tasks[C]//In Proc. of the 13th Euromicro Conf. on Real-time Systems. North Carolina: University of North Carolina,2001:76-85.

[6] Bovet D, Cesati M. Understanding the Linux Kernel[M]. 3rd edition. [s.l.]: O'Reilly Publishers, 2005.

[7] Holman P, Anderson J. Adapting Pfair scheduling for symmetric multiprocessors [J]. Journal of Embedded Computing, 2005,1(4):543-564.

[8] Holman P, Anderson J H. Implementing Pfairness on a symmetric multiprocessor[C]//Real-Time and Embedded Technology and Applications Symposium, 2004. Proceedings RTAS 2004 10th IEEE. [s.l.]:[s.n.],2004:544-553.