计算机科学 2001Vol 28№ 2

半结构化数据的模式研究综述

Schema of Semistructured Data A Survey

王 静 孟小峰

(中国人民大学信息学院数据与知识工程研究所 北京 100872)

Abstract With the rapid development of the Internet, semistructured data have recently become an important topic of study. In semistructured data, there is no predefined, separete schema. In contrast, schema of semistructured data is implicit, and is embeded in data. To facilitate the operations of semistructured data, adding some schemas may be helpful. This survey will cover a number of issues surrounding schema of semistructured data, including schema formulation, schema extraction, schema application, etc.

Keywords Semistructured data, OEM, DataGuide, Graph schema, XML

1 引言

近年来,Internet 的飞速发展已经给人类的生活带来了翻天覆地的变化,网络迅速成为一种重要的信息传播和交换的手段。在 Internet 上,存在着大量的各种形式的数据,如文本文件,HTML 文档,各种数据库等,如何快速准确地在网上查找所需的信息,成为一个重要的问题。为了解决这个问题,各种搜索引擎维护了大量网站和网页的索引信息,主要采用关键字匹配的方法为用户提供查询服务,但关键字匹配技术存在一些很明显的缺陷,如返回的结果集过大,不能对语义内容和结构进行查询等,这些缺陷决定了搜索引擎所能提供的查询能力是极其有限的。为了更好地对 Internet 上的数据资源进行管理和查询,数据库界的研究人员开始考虑将数据库的概念和技术引入到该领域。数据库中已有的许多比较成熟的思想和技术,在 WWW 的新环境下,需要进行扩展和调整,以适应其特点。

Web 可以被看成是一个巨大的、异构的、分布的、由超文本链接所连接的文档集合,对这样的数据进行查询与传统的数据库查询有着明显的不同。首先,已有的数据模型不能很好地适应网上数据的特点,需要引入新的数据模型;其次,由于 Internet 上的许多数据经常缺乏明确的模式,存在不规则的数据形式,这就给查询和处理提出了新的挑战,由此人们提出了半结构化数据的概念。半结构化数据是界于严格结构化的数据(如关系数据库中的数据)和完全无结构的数据(如声音,图像文件)之间的数据形式,它具有如下一些特点:

(1)隐含的模式信息。半结构化数据具有一定的结构,但其结构与数据混在一起,没有显式的模式定义,如 HTML 文件。

(2)不规则的结构。一个数据集合可能由异构的元素组成,例如学生集合中某些学生有电子邮件地址,而另一些学生则没有。同样的信息可能由不同类型的数据表示,例如某些姓名是字符串,而另一些则是由 first name 和 last name 组成的复杂结构。

(3)没有严格的类型约束。由于没有一个预先定义的模式,以及数据在结构上的不规则性,所以缺乏对数据的严格类型约束。

目前国内外关于半结构化数据的研究主要集中在新的数据模型、查询模式、存储技术以及优化技术等方面[1,2]。在众多的研究课题中,对半结构化数据的结构的研究是一个非常重要的方向。半结构化数据存在一定的结构,但这些结构或者没有被清晰地描述,或者是经常动态变化的,或者过于复杂而不能由传统的模式定义来表现。半结构化数据的模式与传统的关系或面向对象数据的模式不同,主要有如下一些特点:

(1)对半结构化数据来说,是先有数据后有模式;

(2)半结构化数据的模式是用于描述数据的结构信息,而不是对数据结构进行强制性的约束;

(3)半结构化数据的模式是非精确的,它可能只描述数据的一部分结构,也可能根据数据处理的不同阶段的视角而不同;

(4)半结构化数据的模式可能规模很大,甚至超过源数据的规模,而且会由于数据的不断更新而处于动态的变化过程中。

没有强制性的模式的限制,使半结构化数据具有很大的灵活性,能够满足网络这种复杂分布环境的需

要，但是也给数据处理带来了很大的困难，使得数据处理的效率低下，很难具有实用性。半结构化数据模式在实际的数据处理中有着很广泛的用途，主要有如下一些：

(1)用户界面。由于半结构化数据没有明确的模式，给用户查询带来了很大的困难。模式信息有助于用户了解数据的结构，从而提出更精确和有效的查询；

(2)查询优化处理。模式信息有助于查询处理器对查询计划进行优化，大大缩减查询的搜索空间；

(3)改进数据存储。了解模式信息，可以更好地设计数据的物理存储结构以及索引，从而提高查询执行的效率；

(4)异构数据源的集成。了解不同的数据源的模式信息，有助于选择适当的集成模式和定义转换规则。

由于认识到半结构化数据模式的重要性，近年来学者们已经在这方面做了很多研究工作，有许多工作目前仍在进行当中。

2 模式的描述形式

对于半结构化数据的模式，目前已经提出了多种描述形式，比较有代表性的有基于逻辑的形式和基于图的形式。无论是哪种描述形式，其讨论的基础都是采用带标记的有向图作为半结构化数据模型，该类模型的典型代表就是OEM模型[5]。

2.1 基于逻辑的描述形式

在已经提出的半结构化数据模式的描述形式中，基于逻辑的描述形式是重要的一类，如一阶逻辑，描述逻辑以及Datalog等。它们非常类似，但在表达能力等方面有所差别，这方面比较典型的是基于Datalog的模式描述形式[6]。

Datalog是一种数据库语言，也可以看作基于逻辑的一种数据模型。采用Datalog规则来描述半结构化数据的模式，其主要思想就是通过指明应有的入边和出边来定义对象的类型，所给出的模式定义就是一组Datalog规则。下面用一个例子来进行描述，假设我们想定义三种类型的对象：

Root类型：有标记为"company"的出边指向类型为Company的对象；有标记为"person"的出边指向类型为Person的对象。

Person类型：有标记为"name"和"position"的出边指向类型为String的原子对象；有标记为"workfor"的出边指向类型为Company的对象；有来自Company类型对象的标记为"employee"的入边。

Company类型：有标记为"name"和"address"的出边指向string类型的原子对象；有标记为"employee"的出边指向类型为Person的对象；有来自Person类对象的标记为"workfor"的入边。

以上三个类型定义可用如下的一组Datalog规则来描述：

root(X):ref(X,person,Y),person(Y),ref(X,company,Z),company(Z)

person(X):company(Z),ref(Z,employee,X),ref(X,workfor,U),company(U),ref(X,name,N),String(N),ref(X,position,P),String(P)

company(X):person(Z),ref(Z,workfor,X),ref(X,employee,E),person(E),ref(X,name,N),String(N),ref(X,address,A),String(A)

这里的每条规则对应于一个类型定义。这种描述形式也可以等价地用一阶逻辑来表示，如上述对Person类型的规则定义可以被等价地写为：

person(X)⇔∃Z(ref(Z,employee,X)∧company(Z))∧∃Y(ref(X,workfor,Y)∧company(Y))∧∃N(ref(X,name,N)∧string(N))∧∃P(ref(X,position,P)∧string(P))

对于类型定义的规则集合，所要考虑的另一个问题是如何确定数据对象与类型的所属关系。这里可以应用最大不动点概念，从包含已有的知识和所有可能的类型划分的模型出发，计算其最大不动点，得到对数据对象的一个类型划分。

2.2 基于图的描述形式

半结构化数据模式的另一种重要描述形式是基于图的形式，文[7]中对其进行了详细的探讨。由于半结构化数据一般采用带标记的有向图来表示，所以这种模式描述形式的一个显著优点是模式和数据采用同一

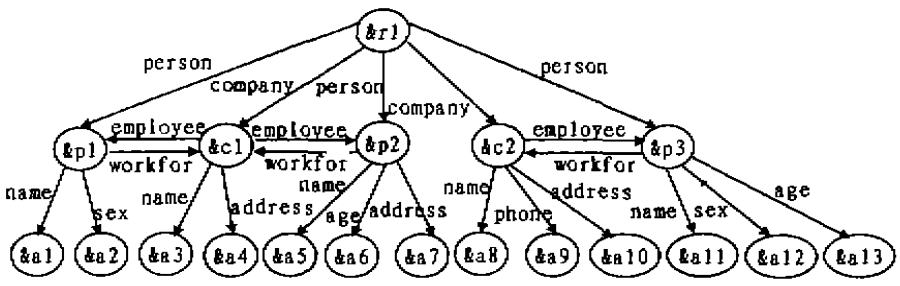

图1 Data Graph

种数据模型(图模型),给处理带来了很大的方便。模式图通常是一个有根、边上带标记的有向图,其边上的标记可以与数据图相同,也可以加以扩充,如允许类似于"name | address"的形式,或采用特定形式的规则(如一元谓词),等等。模式图中的节点可以加以一定的注释,表明其代表的语义或其他特定的含义。考虑图1中的数据图,可以看出图中主要有两种类型的对象:person 和 company,该数据图可能的一个模式图如图2所示,该模式图中的节点都加了注释,来说明相应的类型。有许多学者还提出了其他形式的模式图,但本质上基本相同,这里就不再一一介绍。

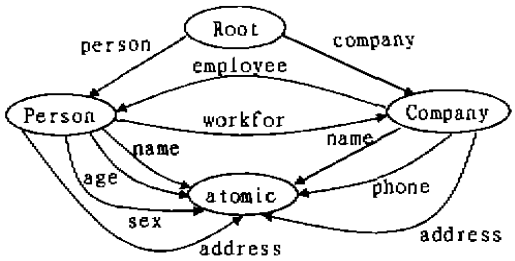

图2 Schema Graph

采用图模型来表示模式,所要研究的两个问题是:(1)如何判断数据实例是否符合模式图?(2)如果数据图符合模式图,如何得到数据图中的对象与模式图中的类型之间的对应关系?目前对这两个问题的研究是基于相似的概念。简单来说,相似就是两个图 G1和 G2的节点之间的一个关系,在这个关系下,G1中的每条边在 G2中都有一条对应的边。具体到数据图和模式图的相似,这个定义可以做一些改动,如数据图和模式图中边的标记可以按特定的关系来对应,数据图的根节点对应于模式图的根节点,以及原子对象必须对应于原子类型等。如果在数据图和模式图之间存在满足上述定义的相似关系,就可以认为数据图符合模式图,而在相似关系中两图的节点之间的对应关系就是数据对象和类型的对应关系。下面的列表给出了图1中的数据图和图2中的模式图之间的相似关系。例如,&r1和 Root 类型有相似关系,那么数据图中从 &r1发出的边 person(&r1,&p1),person(&r1,&p2),person(&r1,&p3)对应于模式图中的从 Root 发出的边 person(Root,Person),而且 &p1,&p2,&p3与 Person 也具有相似关系,从 &r1发出的标记为"company"的边情况也相同。

Data node	Schema node
&r1	Root
&c1,&c2	company
&p1,&p2,&p3	Person
&a1,&a2,&a3,&a4,&a5,&a6,&a7,&a8,&a9,&a10,&a10,&a11,&a12,&a13	Atomic

关于相似的一些算法在图的研究领域中已经有人探讨[1,2]。

3 模式的抽取

前面已经提到,半结构化数据是先有数据后有模式,所以模式研究的一个重要方面就是如何从数据中得到模式,即模式的抽取。模式抽取所研究的问题是:给定一个数据实例,在没有任何事先知识的情况下,自动地计算数据的相应模式;如果存在多个可能的模式,选择能最好地描述给定数据的模式。目前提出的模式抽取方法主要有 DataGuide[4],基于 Datalog 规则的抽取[6],以及一些聚类(clustering)和分类方法,如文[8]中提出的概念聚类方法。

3.1 图模式的抽取

为了从半结构化数据中得到模式图,需要进行抽取工作,在这方面比较有代表性的是 DataGuide[4]。DataGuide 是斯坦福大学在半结构化数据库管理系统 Lore 中已采用的模式图,它本身也是 OEM 对象,是对数据图结构的精确概括,满足如下两个特性:

(1)精确性。凡是在数据图中出现的路径都在 DataGuide 中出现,凡是在 DataGuide 中出现的路径都在数据图中存在。

(2)简洁性。在 DataGuide 中每条路径只出现一次。

从一个给定的数据源中创建一个 DataGuide 等价于将非确定性有限自动机(NFA)转化为确定性有限自动机。根据自动机理论,我们可以知道一个数据源可能会有多个 DataGuide。在可能的多个 DataGuide 中,Lore 系统选择了 Strong DataGuide。所谓的 Strong DataGuide,就是在满足 DataGuide 的基本特性的基础上引入了源数据图中的目标集合和模式图中的节点的一一对应关系,即在源数据图中沿同一路径所能达到的节点的集合对应于模式图中的一个节点。图3给出了一个数据图和它的两个 DataGuide,子图(c)是一个一般的 DataGuide,子图(b)是 Strong DataGuide。Strong DataGuide 的创建算法很简单,可以按深度优先的策略遍历整个源数据图,检查所有可能的路径及其对应的节点集合,生成相应的模式图中的节点和边。

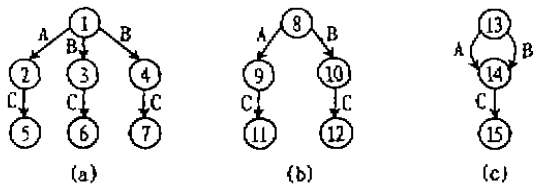

图3 A Source 和 Two DataGuides

DataGuide 在源数据图是树型结构的情况下规模

较小，比较实用，但当源数据图中存在较多的参照和环路时，其抽取算法的执行效率比较低，而且产生的模式图可能过于庞大，甚至比源图更大，这是 DataGuide 存在的严重缺陷。针对这个问题，斯坦福大学的学者也提出了一些改进方案，如近似的 DataGuide[15]等。

3.2 Datalog 规则的抽取

当采用模式的另一种描述形式 Datalog 时，模式的抽取问题就表现为如何对一个给定的半结构化数据源自动地得到最适合的类型定义（即一组 Datalog 规则）。对这个问题，学者们也进行了一些探讨，在文[6]中提出，从半结构化数据中抽取说明其类型定义的 Datalog 规则集的过程如下：

(1)首先，为数据图中的每个复杂对象定义一条规则，这些定义严格地依照每个对象的入边和出边情况。

(2)对已有的知识集和所有可能的类型划分，计算其最大不动点，得到对数据对象的一个类型划分。

(3)如果在得到的类型划分中，有两个类型的对象集相同，则合并这两个类型。由此可以减少类型的数目。

(4)如果经过上述过程，所得到的类型数目仍然较多，可以选择适当的方法对类型进行聚类，进一步减少类型数目，然后再给数据图中的对象指定相应的类型。

通过这个处理过程，可以得到近似的类型定义，与精确的类型相比，它的规模较小，比较易于在实际的数据处理中使用。

3.3 基于聚类的模式抽取

一些半结构化数据研究领域的研究人员将人工智能领域的技术引入，提出了一些基于机器学习方法的模式抽取方法，主要采用聚类或分类的方法。香港中文大学所提出的概念图 CG(Conceptual Graph)[8]就是引入了机器学习中的增量概念聚类思想。CG 是半结构化数据的近似图模式，它的抽取采用机器学习中提出的增量概念聚类的方法，将数据图中的对象根据其入边和出边集合的相似程度进行聚类。该算法每次处理数据源图中的一个对象，考虑将对象加入现有的类或新建一个类所得到的效用函数，选择效用函数值最大的方案，将对象加入所选择的类，并根据各类中对象之间存在的有向边为类之间添加相应的带标记的边。由于算法是增量学习式的，所以当遍历源数据图的顺序不同时，产生的聚类结果可能会不同。

这种方法所得到的模式图可能包含源图中不存在的路径，或者重复路径，所以是不精确的。不精确性带来的好处是模式图的规模大大缩小，使其实用性大大增强。CG 在查询优化中的应用，以及其与 DataGuide 的性能比较在文[8]中有详细的讨论。

4 半结构化数据与 XML

XML(Extended Markup Language)作为一种新的网上数据交换的标准，正在引起人们极大的关注。XML 是标准的通用标记语言 SGML(ISO 8879)的一个子集，用于支持 Internet 上有结构文档的交换。和 HTML 相比，XML 是面向内容的，它具有更多的语义，良好的可扩展性，简单而易于掌握，自描述等特点，适用于 Web 上的数据交换，所以我们可以预言 XML 将成为数据组织和交换的事实标准，并且大量的 XML 数据将很快出现在 Web 上，目前对 XML 数据的存储和查询方面的研究正方兴未艾[9]。XML 数据模型与半结构化数据模型有很多的相似性，可以说，XML 是 WWW 上的半结构数据，它既为半结构化数据的研究提供了广阔的应用前景，同时也推动了半结构化数据研究的发展。

XML 图是一种非常灵活的数据模型，一个 XML 图必须满足下列条件：图的顶点由唯一的字符串标识，称为对象标识(OID)；图的边用 element tag 标记；图的结点用一组属性值标记；图的叶结点由值(字符串)标记；图有一个根结点。XML 图非常适合描述分布的、多态的、动态改变的 Web 数据，而另一方面，数据(包括不规则数据)与 XML 图也能很方便地直接映射。在 OEM 模型与 XML 图之间的对应非常简单：OEM 对象对应于 XML 中的元素(element)，OEM 中的子对象关系反映了 XML 中的元素嵌套，它们之间的不同之处在于 XML 的子元素可能是有序的，以及 XML 元素可能包含(属性，值)的列表。为了支持 XML 的这两个特点，可以在 OEM 模型中引入如下三个新特性：有序的子对象，(属性，值)列表，以及参照边(reference edge)，就可以成为支持 XML 的数据模型了。

为了更有效地进行 XML 数据的处理，学者们提出了许多关于 XML 数据模式描述的方案，其中最主要的是文档类型定义(DTD)[10]。与半结构化数据的模式相比，DTD 的优点是它的正则语法支持定义半结构化的数据，如：(!ELEMENT e(a,b?,c+))说明元素 e 由一个子元素 a，可选的子元素 b，要求出现或连续出现多次的子元素 c 组成。DTD 作为模式，其缺点可以总结为以下几条：

(1)元素的定义包含不必要的顺序；

(2)参照无法定义约束；

(3)对原子类型支持不够，只能支持文本或字符串类型，并且不支持范围说明(如限定 age 的值在整数0和100之间)；

(4)元素的标记在整个文档范围内是全局唯一的，但是在对象数据库中，一个 name 域既可以出现在

person 类中，也可以出现在 course 类中，而在 XML 中，这只能用两个不同的标记来实现：personname 和 coursename。XML 可以使用 XML 名空间的形式解决这个问题，即将标记变为：person：name 和 course：name，person 和 course 是 XML 名空间的名称。

除了 DTD 外，有很多研究采用 XML 1.0标准[17]提出了更适合表达 XML 数据模式的方法，如 XML-Data、XML-Schema、DCD[16]等，这些方法的共同点是：(1)与 DTD 兼容并提供更丰富的描述能力；(2)使用 XML 1.0的语法规范定义自身。

同时，这些研究又各有侧重：

(1)XML-Schema 侧重从文档语言的角度给出 XML 模式定义语言的规范说明，它包括两个部分：结构部分(描述 XML 1.0文档结构并限定其内容)和数据类型部分(描述 XML 语言应支持的数据类型)。

(2)XML-Data 除了可以定义 XML 的语法模式之外，更侧重从数据模型的角度解决概念模式的问题，因此它是语法、数据库与概念模式的交叉。它可以描述数据的类层次、定义主码与参照、定义同义名与关联(类似于"反向指针")、增加新的数据类型定义和值约束，还可以忽略 XML 元素的顺序等等。

(3)DCD 基于 RDF 的数据模型定义 XML 文档内容的结构的约束，其表达范围包括了 XML-Data 的子集。

但是目前这些规范的研究还未成熟，在很多方面还未达成共识。相比之下，基本的 DTD 规范已被广泛接受。

在 XML 模式的研究领域，还有待于进一步的研究与交流，以形成功能完备、形式简洁并被一致认同的规范。

5 其他的研究方向

在半结构化数据的研究领域中，与模式相关的问题还有很多，目前都有人在研究，主要有如下一些：

(1)路径约束。在传统的数据库中，模式除了描述数据的结构和类型以外，还描述对数据的一定形式的约束，如在关系型数据库中的主码和外码约束，面向对象数据库中的 inclusion 约束和 inverse relationship。对半结构化数据而言，存在着路径约束，其一般形式如下：

$$\forall x(\alpha(r,x)\rightarrow\forall y(\beta(x,y)\rightarrow\gamma(x,y)))$$

这种路径约束可以表示在面向对象数据库中的 inclusion 约束和 inverse relationship。研究半结构化数据中的路径约束的分类，约束的推导，类型与约束的关系，以及路径约束对查询优化的作用等，是这个领域的研究重点，宾西法尼亚大学的 Peter Buneman 等人在这方面做了很多工作[13,14]。

(2)模式的应用。得到半结构化数据的模式后，如何将其应用到用户界面，查询优化的处理，构建索引等方面也是备受关注的研究方向，在这方面的工作将决定半结构化数据模式的实用意义。

结束语 本文主要介绍了目前在半结构化数据的模式这个研究领域的一些已有的研究成果，以及相关的研究方向，并提出了一些看法。这个领域的许多工作目前正在进行中，新的论文及观点不断出现，是一个非常活跃的研究领域。

参 考 文 献

1 Florescu D, Levy A, Mendelzon A. Database Techniques for the World-Wide Web: A Survey. ACM SIGMOD Record, 1998, 27(3)

2 Buneman P. Semistructured Data. In: Proc. of the ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems (PODS). Tucson, Arizona, 1997. 117～121

3 Papakonstamtinous Y, Molina H G, Widom J. Object exchange across heterogeneous information sources. In: Proc. of Int. Conf. on Data Engineering (ICDE). Taipei, Taiwan, 1995. 251～260

4 Goldman R, Widom J. DataGuide: Enabling Query Formulation and Optimization in Semistructured Databases. In: Proc. of the Intl Conf. on Very Large Data Bases (VLDB). Athens, Greece, 1997

5 Nestorov S, et al. Representative Objects: Concise Representations of Semistructured, Hierarchical Data. In: Proc. of Int. Conf. on Data Engineering (ICDE). Birmingham, U.K., April 1997. 79～90

6 Nestorov S, Abiteboul S, Motwani R. Extracting Schema from Semistructured Data. In: Proc. of ACM SIGMOD Conf. on Management of Data. Seattle, WA., 1998

7 Buneman P, et al. Adding structure to unstructured data. In: Proc. of the Int. Conf. on Database Theory (ICDT). Delphi, Greece, 1997

8 王秋月，Jeffrey Xu Yu，黄锦辉. Approximate Graph Schema Extraction for Semi-Structured Data. In: Proc. of EDBT 2000, Germany, March 2000

9 Suciu D. Semistructured Data and XML. Available at: http://www.research.att.com/～suciu, 1999

10 Nestorov S, Abiteboul S, Motwami R. Inferring structure in semistructured data. In: Proc. of the Workshop on Management of Semi-structured Data. 1997

11 Beeri C, Milo T. Schemas for integration and translation of structrued and semistructured data. In: Proc. of the Intl. Conf. on Database Theory (ICDT). 1999

12 Henzinger M, Henzinger T, Kopke P. Computing simulations on finite and infinite graphs. In: Proc. of the 20th Symposium, on Foundations of Computer Science. 1995. 453～462