

一种有效的大规模 XML 数据发布系统中订阅管理方法^{*}

董 彪¹ 陈金辉² 孙亚民¹

(南京理工大学计算机科学与技术学院 南京 210094)¹ (南京信息工程大学信息与控制学院 南京 210044)²

摘 要 随着 Internet 上 XML 文档传输量的激增,有效地订阅管理成为整个订阅/发布系统的关键技术之一。把用户海量的订阅条件聚集到一个较小的集合上,这是系统大规模和高效性的要求。对基于 XPath 树模式的订阅聚集进行系统分析,定义了树模式间的祖孙、包含等语义关系,给出了一个新颖的、优化的树模式包含算法。研究成果能被扩展到 XML 查询处理等领域。

关键词 订阅管理,树模式,聚集,包含算法

Efficient Subscription Management for Large-scale XML Data Dissemination

DONG Biao¹ CHEN Jin-hui² SUN Ya-min¹

(School of Computer Science & Technology, Nanjing University of Science & Technology, Nanjing 210094, China)¹

(School of Information & Control, Nanjing University of Information Science & Technology, Nanjing 210044, China)²

Abstract With the rapid growth of XML-document traffic on the Internet, efficient subscription management is critical for the overall performance of the Publish/Subscribe system. System scalability and efficiency mandate the ability to aggregate the set of consumer subscriptions to a smaller set of content specifications, so as to both reduce their storage-space requirements as well as speed up the document subscription matching process. In this paper, we provide the systematic study of subscription aggregation where subscriptions are specified with tree patterns. Some kinds of semantic relationships are defined, such as ancestor/descendant and containment, etc. As part of our solution, we also develop a novel optimized algorithm for tree-pattern containment. The usefulness of our results on tree patterns and their aggregation is not limited to content-based routing, but also extends to other application domains such as the optimization of XML queries.

Keywords Subscription management, Tree pattern, Aggregation, Containment algorithm

1 引言

随着 XML 成为数据表示和交换的标准,Internet 中出现了大量的与 XML 数据流相关的应用,如基于内容的 XML 路由、XML 文档分发、订阅/发布的主动服务系统^[1-6]等。在订阅/发布系统中,用户使用一些 XML 模式语言(如 XPath)表明订阅条件,有效的订阅管理是整个订阅/发布系统中的关键技术之一^[7-8]。

订阅聚集本质上是聚合一个原始的订阅集 S 到一个较小的集合 A ,以致对于任何一个文档,当它匹配 S 中的订阅时,也匹配 A 中相应的订阅。订阅聚集增加了路由的复杂性,路由器需判断一个新的订阅条件与路由表中的已有订阅集之间的包含关系。文[9]给出了树状订阅条件之间包含关系的算法。当注册的一个新订阅 S 时,路由器必须与路由表中的所有订阅 $\{S_i\}$ 进行比较。该算法的时间复杂度为 $O(\sum_{i \in n} |S_i| \cdot |S|)$, $|S_i|$ 是 S_i 的结点数,该迭代执行是无效的。

文[10]中提出了基于(属性,谓词)对的平面模式方式。文献[11]中研究了多播环境中的查询合并问题,考虑查询的图形化,而未考虑空间的限制。文献[12,13]中研究了 XML 树模式集的最小化算法,但其树模式不支持‘*’,对‘//’处理过于复杂。文献[14,15]中使用“编辑距离”计算两棵数据结

构树的相似性,未涉及树模式的相似性。文献[16]中运用 PST 结构和 trie 路径编码,研究了单文档中对元素 XPath 查询问题,未涉及所有文档的模式相似性。

与上述方法相比,本文的主要特点:

- (1) 基于 XPath 树模式,要求更加复杂的聚集技术;
- (2) 定义了树模式间的包含、祖孙、等价关系,该定义是基于语义的,不依赖于特定的 XPath 语法;
- (3) 定义了树模式间的同态映射,利用同态估值矩阵和距离矩阵对包含算法进行有效的优化。给出解决冗余订阅的形式定义;
- (4) 给出树模式的动态订阅、注销算法。

2 基本概念和定义

XML 可以看作是结点标记的有向树 $T(VT, ET)$, 其中 VT 表示结点的集合, ET 表示边的集合。设 t 为 T 中结点, 根为 T_{root} , $SubTr(t, T)$ 表示 T 中以 t 为根的子树。

Xpath 标识 XML 文档的内容和结构条件,称为树模式。本文中,树模式是一个无序的带结点标记的树。Nodes(P)表示树模式 P 中结点的集合,根为 P_{root} ; $SubTr(v, P)$ 表示 P 中以 v 为根的子树,称为 P 的子模式。从 P_{root} 开始的一条路径,称为一个简单的 XPath 表达式。树模式的边有两种类

^{*} 本研究得到南京信息工程大学科研基金资助项目(Y640)资助。董 彪 博士生,主要研究方向为中间件技术;陈金辉 副教授,主要研究方向为系统分析与集成;孙亚民 教授,博士生导师,主要研究方向为网络工程。

型,相应于/和//,记为 $\text{Edges} = \text{Edges} \setminus \cup \text{Edges} \setminus$ 。

设 x 为 T (或 P)中结点, $\text{Child}(x)$ 表示 x 的孩子结点的集合, $\text{Des}(x)$ 表示 x 的子孙结点(包括孩子结点), $\text{Label}(x)$ 表示 x 的标记。当 x 为 P 中结点时, $\text{Lbl}(x)$ 的标记值为标记元素、* (任意的元素)之一。本文分别用 P, Q 表示树模式, S, S' 表示树模式的集合。

定义 1 T 在结点 t 满足 $\text{SubTr}(v, P)$, 记为 $(T, t) \text{SubTr}(v, p)$, 满足以下条件:

- (1) 当 $\text{Lbl}(v)$ 是元素时, $\exists t' \in \text{Child}(t), \text{Lbl}(t') = \text{Lbl}(v)$, 并且对 $\forall v' \in \text{Child}(v)$, 有 $(T, t') \text{SubTr}(v', p)$;
- (2) 当 $\text{Lbl}(v) = ' * '$ 时, $\exists t' \in \text{Child}(t), \text{Lbl}(t') =$ 任意元素, 并且对 $\forall v' \in \text{Child}(v)$, 有 $(T, t') \text{SubTr}(v', p)$;
- (3) 当 $\text{Lbl}(v) = ' // '$ 时, $\exists t' \in \text{Des}(t)$, 并且对 $\forall v' \in \text{Child}(v)$, 有 $(T, t') \text{SubTr}(v', p)$ 。

定义 2 T 满足 P , 记为 $T \text{SubTr} P$, 当且仅当对于 $\forall v \in \text{Child}(P_{\text{root}})$, 满足以下条件:

- (1) 当 $\text{Lbl}(v)$ 是元素时, $\text{Lbl}(t_{\text{root}}) = \text{Lbl}(v)$, 并且对 $\forall v' \in \text{Child}(v)$, 有 $(T, t_{\text{root}}) \text{SubTr}(v', p)$;
- (2) 当 $\text{Lbl}(v) = ' * '$ 时, $\text{Lbl}(t_{\text{root}}) =$ 任意元素, 并且对 $\forall v' \in \text{Child}(v)$, 有 $(T, t_{\text{root}}) \text{SubTr}(v', p)$;
- (3) 当 $\text{Lbl}(v) = ' // '$ 时, $\exists t' \in \text{Des}(t_{\text{root}}), T' = \text{SubTr}(t', T) \Rightarrow T' \text{SubTr} P$ 。

定义 3 Q 包含 P , 记为 $Q \supseteq P$, 当且仅当对于任意一棵 XML 树 $T, T \text{SubTr} P \Rightarrow T \text{SubTr} Q$ 。称 Q 为包含模式, P 为被包含模式。

定义 4 Q 与 P 等价, 记为 $P \equiv Q$, 当且仅当 $P \subseteq Q \wedge Q \subseteq P$ 。

定义 5 S 包含 S' , 记为 $S \subseteq S'$, 当且仅当对于 $\forall P \in S, \exists P' \in S' \wedge P \subseteq P'$ 。

定义 6 Q, P 具有祖孙关系, 记为 $Q < P$, 当且仅当对于任意一棵 XML 树 $T, \forall y \in \text{Nodes}(P) \wedge (T, y) \text{SubTr}(P_{\text{root}}, P) \Rightarrow \exists x \in \text{Nodes}(Q) \wedge (T, x) \text{SubTr}(Q_{\text{root}}, Q) \wedge y \in \text{Des}(x)$

树模式间的包含、祖孙、等价关系是基于语义的, 不依赖于特定的 Xpath 语法表示。

定理 1 P 和 Q 之间的祖孙、包含关系具有以下性质:

$$Q < P \Leftrightarrow Q // * \supseteq P \quad Q \supseteq P \Leftrightarrow Q / a < P / a / *$$

其中 a 是不出现在 Q 中的任何一个元素。证明从略。

定义 7 S 上的订阅聚集图 AG 是一个二元组 $AG (AGN, AGE)$, 其中:

(1) AGN 是结点集合。树模式的等价关系把 S 划分为等价类, 结点与等价类一一对应, 结点 N_i 对应的等价类记为 $[s_i] =$;

(2) AGE 是边的集合, 有虚边和实边两种。 $[s_i] = \supseteq [s_j] =$, 从 N_i 到 N_j 的边为虚边; $[s_i] = < [s_j] =$, 从 N_i 到 N_j 的边为实边。

定义 8 AG 中一条边是冗余边, 当且仅当这条边能从下面的蕴含规则推断:

$$\begin{aligned} [s_1] = \supseteq [s_2] = \wedge [s_2] = \supseteq [s_3] = &\Rightarrow [s_1] = \supseteq [s_3] = \\ [s_1] = < [s_2] = \wedge [s_2] = < [s_3] = &\Rightarrow [s_1] = < [s_3] = \\ [s_1] = < [s_2] = \wedge [s_2] = \supseteq [s_3] = &\Rightarrow [s_1] = < [s_3] = \\ [s_1] = \supseteq [s_2] = \wedge [s_2] = < [s_3] = &\Rightarrow [s_1] = < [s_3] = \end{aligned}$$

删除 AG 中所有的冗余边后的图, 称约简的聚集图 AG 。

3 包含算法及订阅管理

3.1 包含算法

订阅管理的核心是包含算法, 检测是否 $P \supseteq Q$ 。如果直接运用定义 3 计算包含关系, 需要检查所有可能的 XML 树 t 。下面研究实际的包含关系检测算法。

定义 9 Q 与 P 同态当且仅当存在一个从 Q 到 P 的同态映射 Ω , 并且满足以下条件:

- (1) $\forall x \in \text{Nodes}(Q), \exists y \in \text{Nodes}(P), y = \Omega(x), \text{Lbl}(x) = \text{Lbl}(y) = a \vee \text{Lbl}(x) = ' * ', a$ 为元素;
- (2) Q 的/边映射到 P 中的/边; Q 中的//边映射到 P 中的一条路径;
- (3) Q 的返回结点映射到 P 的返回结点。

例 1 $Q = /a_1/a_2[///b_1]/ * [/a_3/b_2]/c, P = a_1/a_2/[//c_1]/d[///a_3[/b]/a_4]/c_2$, 其中相同元素用下标区分。结点间同态映射关系: $\{(a_2, a_2), (*, d), (c, c_2), (b_1, b), (b_2, b), (a_3, a_3)\}$ 。可进一步验证边的同态映射关系。

定理 2 如果存在一个从 Q 到 P 的同态映射, 那么 $Q \supseteq P$ 。证明从略。

从 Q 到 P 的同态映射, 可以检查树模式包含关系。

定义 10 从 Q 到 P 的同态估值矩阵 C , 对于 $\forall x \in \text{NODES}(p) \wedge \forall y \in \text{NODES}(p')$, 在 C 中存在一项 $C(x, y)$ 。当且仅当 y 到 x 存在一个同态映射时, $C(x, y) = \text{true}$ 。

定理 3 $C(x, y)$ 依赖于 $C(x', y')$, 其中 y' 是 y 的孩子结点, x' 是 x 的孩子结点或子孙结点。 $C(x, y) = \text{true}$, 满足以下条件:

- (1) $\text{Lbl}(y) = ' * ' \vee \text{Lbl}(x) = \text{Lbl}(y)$;
- (2) 如果 $(y, y') \in \text{Edges}_{//}(Q)$, 那么 $\forall (x, x') \in \text{Edges}_{//}(P) C(x', y')$;
- (3) 如果 $(y, y') \in \text{Edges}_{/}(Q)$, 那么 $\forall (x, x') \in \text{Edges}_{/}(P) C(x', y')$, 其中 $\text{Edges}_{/}^+(p)$ 表示 $\text{Edges}(P)$ 的传递闭包。证明从略。

定义 11 从 Q 到 P 的同态距离矩阵 D , 当 $\exists x' \in \text{Des}(x), C(x', y') = \text{true}$ 时, $D(x, y')$ 等于从 x 到 x' 的路径长度。

定义 12 $K(y, y')$ 为从 y 到 y' 连续的 * 串长, $y, y' \in \text{Nodes}(P)$ 。若从 y 到 y' 存在一条路径 $y_0 e_0 y_1 e_1 y_2 \dots e_{n-1} y_n$, 其中 $a_i = ' * ', e_i = ' // '$ 或者为 $' / '$, 则 $K(y, y') \geq n$ 。

定理 4 C, D 和 K 之间有如下关系: $C(x, y) = (\text{Lbl}(y) = ' * ' \vee \text{Lbl}(x) = \text{Lbl}(y)) \wedge \bigwedge_{(y, y') \in \text{Edges}_{/}(p')} (C(x', y') \vee \bigwedge_{(x, x') \in \text{Edges}_{/}(p')} C(x', y')) \wedge \bigwedge_{(y, y') \in \text{Edges}_{//}(p')} (D(x, y') \geq 1 + k(y, y'))$

证明从略。

算法 1 计算从 Q 到 P 的同态距离。

```
CompD(x, y)
if C(x, y) then d = 0 else d = -∞;
Return(
    max(d, 1 + max_{(x, x') ∈ Edges_{/}(p)} D(x', y),
    1 + max_{(x, x') ∈ Edges_{//}(p)} (k(x, x') + D(x', y))))
```

定理 5 若 $C = \text{true}$, 则 $Q \supseteq P$ 。证明从略。

算法 2 计算从 Q 到 P 的同态映射。

```
FindH(Q, P)
for x ∈ Nodes(P) do
    for x' ∈ Nodes(Q) do
        C(x, y) =
            (Lbl(y) = ' * ' ∨ Lbl(x) = Lbl(y)) ∧
            (∧_{(y, y') ∈ Edges_{/}(p')} (C(x', y') ∨
            ∧_{(x, x') ∈ Edges_{/}(p')} (D(x, y') ≥ 1 + k(y, y'))))
            if C(x, y) d = 0 else d = -∞;
        D(x, y) = max(d, 1 + max_{(x, x') ∈ Edges_{/}(p)} D(x', y),
        1 + max_{(x, x') ∈ Edges_{//}(p)} (k(x, x') + D(x', y)));
    return C(P_{root}, Q_{root})
```

算法 2 的时间复杂度为 $O(|p| |p'|)$ 。

3.2 订阅管理

订阅管理的有效性要求海量的用户订阅聚集到一个较小的集合,从而使路由表的大小适应主存,同时加速文档的过滤。构造 S 上的订阅聚集图 AG ,就是计算 S 上所有的包含和祖孙关系。订阅/发布系统中的树模式是动态增减的,因此须采取渐增法计算 AG ,每次插入一个树模式。设 $P_1, P_2, \dots, P_{n-1}$ 对应图 AG_{n-1} , 现把 P_n 插入到 AG_{n-1} 中的适当位置,构成图 AG_n 。

算法 3 构造聚集图,在 AG_{n-1} 中加入 P_n 后得到 AG_n 。

```
add( $P_n, AG_{n-1}$ )
{if Node $P_{equ}$  = EQVal( $P_n, G_0$ )
  merge( $P_n, NodeP_{equ}$ )
else if { if  $\Delta$  = ExistEdges( $P_n, AG_0$ ) 非空
  {AddEdges( $\Delta, AG_{n-1}$ );
   $G_{n-1}$  =  $G_{n-1} + \{P_n\}$ ; }
return( $AG_{n-1}$ )}
```

算法 3 中 AG_0 表示图 AG_{n-1} 中的根结点,即没有输入边的结点;ExistEdges(P_n, AG_0)表示所有从 P_n 到 G_0 中结点的边的集合;EQVal(P_n, AG_0)计算出 AG_0 中与 P_n 等价的某一个结点;merge($P_n, NodeP_{equ}$)把 P_n 并入结点 $NodeP_{equ}$;AddEdges(Δ, AG_{n-1})把 Δ 中的边并入 AG_{n-1} 。

算法 4 在 AG 中删除 P_i 。

```
remove( $p_i, AG$ )
{if  $P_i \in G_0$  {delete( $NodeP_i$ ); return( $G_0$ )}
else remove( $p_i, G - G_0$ ) }
```

算法 4 中 $NodeP_i$ 表示 P_i 在 G 中的对应结点。delete($NodeP_i$)在 G 中删除结点 $NodeP_i$ 及其出边。由于算法 3 是经优化后的算法,以及 AG 中结点数为订阅条件等价类的数目,因此这个算法有非常好的性能。

例 2 树模式的集合 $S = \{ /a/b, /a/b/c, /a/b/c/d, /a/b[c=6], /a/b[c=6]/d, /a/b/d \}$ 。图 1 为 S 对应的约简订阅聚集图,其中实边表示包含关系,虚边表示祖孙关系。

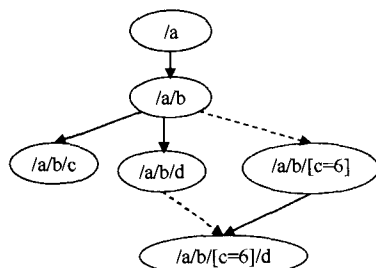


图 1 S 的订阅聚集图

结束语 本文系统地、形式化地研究了订阅/发布系统中的订阅管理问题。基于语义定义了树模式间的包含、等价和祖先关系、订阅聚集图及其约简形式以及树模式间的同态映射。根据同态映射设计了一种新颖的包含算法,利用本文定

义的同态估值矩阵和同态距离矩阵对该算法进行了有效的优化。设计了订阅管理中用户注册和注销时的两个重要算法。本文的研究成果能被扩展到 XML 查询处理等领域。

参考文献

- [1] Diao Y, Fischer P. YFilter: Efficient and scalable filtering of XML documents//Proc. of the 18th Int'l Conf. on Data Engineering, 2002; 341-345
- [2] Chan C, Felber P, Garofalakis M, et al. Efficient filtering of XML document with XPath expressions//Proc. of the Int'l Conf. on Data Engineering, San Jose, IEEE Computer Society, 2002; 235-244
- [3] Green T J, Miklau G, Onizuka M, et al. Processing XML streaming with deterministic automata//Calvanese D, Lenzerini M, Motwani R. eds. Proc. of the Int'l Conf. on Data Theory, LNCS 2572, Springer-Verlag, 2003; 173-189
- [4] Gupta A K, Suciu D. Stream processing of XPath queries with predicates//Halevy A Y, Ives Z G, Doan A H, eds. Proc. of the 2003 ACM SIGMOD Int'l Conf. on Management of Data, ACM, 2003; 419-430
- [5] Nguyen B, Abiteboul S, Cobena G, et al. Monitoring XML data on the Web//Aref W G, ed. Proc. of the ACM/SIGMOD Conf. on Management of Data, 2001; 437-448
- [6] Chen J, Dewitt D, Tian F, et al. NiagaraCQ: A scalable continuous query system for internet databases//Chen W D, Naughton J F, Bernstein P A, eds. Proc. of the ACM/SIGMOD Conf. Management of Data, ACM, 2000; 379-390
- [7] Chand R, Felber P, Garofalakis M. Tree-pattern Similarity Estimation for Scalable Content-based Routing. ICDE 2007; 1016-1025
- [8] Chand R. Large scale diffusion of information in Publish/Subscribe systems. PhD thesis, University of Nice-Sophia Antipolis and Institut Eurecom, 2005; 69-71
- [9] Chan C Y, Fan W, Felber P, et al. Tree Pattern Aggregation for Scalable XML Data Dissemination//Proceedings of VLDB, August 2002; 826-837
- [10] Silberstein A, He H, Yi K, et al. Boxes: Efficient maintenance of order-based labeling for dynamic XML data//Proc. of ICDE, 2005; 285-296
- [11] Liu H, Jacobsen H A. A-ToPSS: A publish/subscribe system supporting imperfect information processing//Proc. of the 30th Int'l Conf. on Very Large Databases, Toronto, Morgan Kaufmann Publishers, 2004; 1281-1284
- [12] Amer-Yahia S, Cho S, Lakshmanan L V S, et al. Minimization of Tree Pattern Queries//Proc. of SIGMOD, Santa Barbara, California, May 2001; 497-508
- [13] O'Neil P, O'Neil E, Pal S, et al. ORDPATHs: Insert-friendly XML node labels//SIGMOD, 2004; 903-908
- [14] Shasha D, Zhang K. Simple Fast Algorithms for the Editing Distance Between Trees and Related Problems, SIAM Journal on Computing, 1989, 18(6); 1245-1262
- [15] Wu X, Lee M, Hsu W. A prime number labeling scheme for dynamic ordered XML trees//Proc. of ICDE, 2004; 66-78
- [16] Chen Z, Jagadish H, Korn F, et al. Counting twig matches in a tree//Proceedings of ICDE, 2001; 595-604

(上接第 249 页)

结束语 (1)本文对图像匹配的相关技术进行了介绍,并提出了川南石刻图像模板匹配的方法、步骤和识别流程;(2)本文对川南石刻图像模板匹配中几个关键性问题进行了重点讨论,并提出了具体的解决方案;(3)采用本文提出的模板匹配方法,识别精度高、计算速度快,通过实验取得了较好的识别效果;(4)本研究是计算机识别技术在艺术领域应用的初步尝试,具有一定的现实意义。但如何扩展识别的范围并改进现有匹配算法,是进一步的研究方向。

参考文献

- [1] 李雅梅,张春新,吴中福. 川南地区南宋墓葬石刻“椅子”造型. 西

南大学学报(社科版), 2007(5)

- [2] 王家文,曹宇. MATLAB6. 5 图形图像处理. 国防工业出版社, 2004
- [3] 张洪刚,陈光,郭军. 图像处理与识别. 北京邮电大学出版社, 2006
- [4] Li Nan, Huang Zhong. Afeature-based pencil method//Proceedings of the 1st international conference on computer graphics and interactive techniques in Australian and south East Asia, Melbourne, Australia, 2003; 135-141
- [5] 于泓,陈辉,赵辉. 基于形状模板匹配的图像拼接算法. 计算机工程与应用, 2006
- [6] 李雅梅,吴中福. 基于形态学变换等技术的川南石刻图像预处理方法研究. 计算机科学, 2008, 35(3)