

# 一种基于模式匹配的超媒体查询模型\*

周学海<sup>1</sup>, 李光亚<sup>2</sup>, 李 曦<sup>1</sup>, 龚育昌<sup>1</sup>, 赵振西<sup>1</sup>

<sup>1</sup>(中国科学技术大学 计算机科学与技术系,合肥 安徽 230027);

<sup>2</sup>(上海万达信息股份有限公司,上海 200233)

E-mail: xhzhou@ustc.edu.cn

http://www.ustc.edu.cn

**摘要:** 在基于语义网络的超媒体模型基础上,通过引入数据库模式、模式约束图以及分层模式依赖图等概念,提出了一种基于模式匹配的超媒体查询模型,给出了图形代数的形式化定义,并证明了该模型的查询能力.该模型具有较高的表达能力,能有效地降低用户的认知负载.

**关键词:** 超媒体系统;数据库模式;模式约束图;图形代数;关系完备性

中图法分类号: TP311 文献标识码: A

随着 Internet 网的普及,超媒体网络的搜索和查询问题变得日益突出,因此,浏览机制和查询机制的有机结合是目前超媒体技术研究亟待解决的问题<sup>[1]</sup>.在查询机制方面需解决的关键问题是查询表达能力与查询描述手段有效性间的权衡问题.

本文在文献[2]所提出的基于语义网络的超媒体模型的基础上,又提出了一种新的超媒体查询模型.为了能够使用户方便地描述查询请求,我们并没有像文献[3]那样提出一种查询语言,而是通过图形用户界面来表达用户的所有查询请求.为了有效地提高查询能力,我们提出了模式依赖图的概念,并讨论了基于模式约束图概念的查询算法.通过给出图形代数的形式化定义,证明了该模型的完备性.最后对相关工作进行了比较分析,并给出了进一步的研究方向.

## 1 超媒体数据库模型

Dexter 模型<sup>[4]</sup>是在内部成员层存储节点的数据信息,在存储层存储超媒体结构信息,将节点内容与结构信息划分为模式层与数据库层,其中模式层用类型化的节点与链所组成的有向标号图来表示,而数据库层则包含不同应用对模式层中相同数据库模式的不同实例.随着超媒体技术在不同应用领域的应用,超媒体系统已经从单纯的浏览机制发展到表达各个信息系统中信息单元间关系的一种集成机制<sup>[5]</sup>,这要求节点具备一定的语义信息,链表达信息单元间的语义关系.语义网络是一种灵活的知识描述机制,它与超媒体之间的关系类似<sup>[6]</sup>.我们在语义网络的基础上提出了超媒体数据库模型.

**定义 1.** 一个超媒体数据库模式  $S$  是一个弱连通有向标号图:  $S = (N_S, E_S, \psi_S, \lambda_S, T_S, C_S)$ . 其中  $N_S = N_{Sa} \cup N_{Sp}$  代表节点集合,  $N_{Sa}$  代表抽象节点,  $N_{Sp}$  代表属性类型节点.  $E_S$  代表边集合,  $E_S = E_{Sa} \cup E_{Sp}$ ,  $E_{Sa}$  表示抽象节点间的关系,  $E_{Sp}$  表示抽象节点与其属性类型节点之间的关系.  $\psi_S$  代表边与节点之间的关联函数,

\* 收稿日期: 2000-09-20; 修改日期: 2001-04-17

基金项目: 国家自然科学基金资助项目(69503005);安徽省自然科学基金资助项目(99043302)

作者简介: 周学海(1966 - ),男,江苏武进人,博士,教授,主要研究领域为信息系统,计算机体系结构;李光亚(1972 - ),男,安徽无为,博士,高级工程师,主要研究领域为数据库系统,软件工程;李曦(1963 - ),男,福建漳州人,副教授,主要研究领域为计算机体系结构,数据库技术;龚育昌(1943 - ),男,上海人,教授,博士生导师,主要研究领域为数据库技术;赵振西(1937 - ),男,山东济南人,教授,博士生导师,主要研究领域为数据库技术.

即  $\psi_S : (E_{Sa} \rightarrow N_{Sa} \times N_{Sp}) \cup (E_{Sp} \rightarrow N_{Sa} \times N_{Sp})$ ,  $T_S$  是节点和边的标号集合, 即  $T_S = T_{SN} \cup T_{SE}$ , 而  $\lambda_S : N_S \cup E_S \rightarrow T_S$  是相应的标号函数.  $C_S$  表示数据库模式  $S$  所必须满足的约束条件, 它描述对于节点、链或超媒体子网进行操作的条件. 我们可以定义  $C_S = BoolOP(C_1, C_2, C_3, \dots, C_n)$ , 其中  $BoolOP = \{\neg, \wedge, \vee\}$  是逻辑关系运算符, 而  $C_i$  为定义于  $N_{Sp}$  之上的逻辑表达式.

对于数据库模式中的每一个节点标号  $t \in T_{SN}$ , 它实际定义了一个节点类型,  $dom(t)$  用来表示对应于此类型所有可能值的集合, 即该类型的值域.

**定义 2.** 超媒体数据库  $G$  为数据库模式  $S$  的实例. 它是一个有向标号图:  $G = (N_G, E_G, \psi_G, \lambda_G, T_G)$ . 其中  $N_G (= N_{Ga} \cup N_{Gp})$  代表节点集合, 而  $E_G (= E_{Ga} \cup E_{Gp})$  代表边集合,  $\psi_G$  代表节点和边之间的关联函数, 即  $(E_{Ga} \rightarrow N_{Ga} \times N_{Ga}) \cup (E_{Gp} \rightarrow N_{Gp} \times N_{Gp})$ ,  $T_G (= T_{GN} \cup T_{GE})$  是由节点和边的标号所构成的集合,  $\lambda_G : N_G \cup E_G \rightarrow T_G$  是标号函数.

超媒体数据库  $G$  与数据库模式  $S$  之间的关系可以用映射函数  $\tau$  来描述,  $\tau : G \rightarrow S$ , 即

(1)  $\forall v \in G, \exists n \in S$ , 即  $\tau(v) = n$ , 且  $\lambda_G(v) \in dom(\lambda_{S(n)})$ ,

(2)  $\forall l \in G, \exists e \in S$ , 即  $\tau(l) = e$ , 即若  $\psi_G(l) = v \times v'$ , 则有  $\psi_S(e) = \tau(v) \times \tau(v')$ .

## 2 查询机制的实现

### 2.1 模式约束图的定义及操作

在超媒体系统中, 降低用户认知负载的方法之一是限定用户的浏览空间. 为此, 我们给出了模式约束图的定义, 它限定了与其所对应的超媒体数据库中的实例.

**定义 3.** 模式约束图  $P$  是一个有向的弱连通图. 即  $P = (N_P, E_P, \psi_P, \lambda_P, T_P, C_P)$ , 其中  $N_P (= N_{Pa} \cup N_{Pp})$  是节点集合,  $E_P (= E_{Pa} \cup E_{Pp})$  是边集合,  $\psi_P$  是节点和边之间的关联函数, 即  $E_P \rightarrow N_P \times N_P$ ,  $T_P$  是节点和边的标号集合,  $\lambda_P$  是标号函数,  $C_P$  是定义于  $N_{Pp}$  之上的计算约束.

$P$  与其在数据库中的每个实例  $I$  之间是一一对应的关系, 即存在一一映射函数  $\theta : P \leftrightarrow I$ .

我们针对模式约束图引入  $DefNode$  和  $DefEdge$  两种操作来支持定义于模式约束图之上的模式约束图, 其目的是为了有效地支持知识的可重用性.

**定义 4.** 对于一个模式约束图  $P$  的一个节点  $n$  所做的  $DefNode$  操作为

设  $P = (N_P, E_P, \psi_P, \lambda_P, T_P, C_P)$ ,  $P' = DefNode(P, n, label)$ ,  $P' = (N_{P'}, E_{P'}, \psi_{P'}, \lambda_{P'}, T_{P'}, C_{P'})$ , 则有  $N_{P'} = N_P$ ,  $E_{P'} = E_P$ ,  $\psi_{P'} = \psi_P$ , 而  $T_{P'} = T_P - \{\lambda_P(n)\} + \{label\}$ ,  $\lambda_{P'} = \lambda_P - \{n \rightarrow \lambda_P(n)\} + \{n \rightarrow label\}$ ,  $C_{P'} = C_P$ .

其中  $n$  为所要改变标号的节点,  $label$  为改变后的标号, 改变标号的目的则是为了区分模式约束图  $P$  中节点  $n$  所对应的数据库中的实例集合与其他模式约束图中与  $n$  具有相同标号的节点所对应的实例集合.

**定义 5.** 对于一个模式约束图  $P$  两个节点  $n_1$  和  $n_2$  所做的  $DefEdge$  操作为

设  $P = (N_P, E_P, \psi_P, \lambda_P, T_P, C_P)$ ,  $P' = DefEdge(P, n_1, n_2, e, label)$ , 其中  $P'$  在  $n_1$  和  $n_2$  之间增加一条新的边  $e$ ,  $P' = (N_{P'}, E_{P'}, \psi_{P'}, \lambda_{P'}, T_{P'}, C_{P'})$ , 则有  $N_{P'} = N_P$ ,  $E_{P'} = E_P + \{e\}$ ,  $\psi_{P'} = \psi_P + \{e \rightarrow n_1 \times n_2\}$ ,  $T_{P'} = T_P + \{label\}$ ,  $\lambda_{P'} = \lambda_P + \{e \rightarrow label\}$ ,  $C_{P'} = C_P$ .

为了对本文下面给出的图形成代数及查询操作提供支持, 我们提出了标号层次的概念, 该层次的建立是通过每一次  $DefNode$  和  $DefEdge$  操作而逐步建立的. 设  $P' = DefNode(P, n, label)$ , 则在该层次中增加一条从  $label$  节点指向  $\lambda_P(n)$  节点的边. 而对于  $DefEdge$  操作, 设  $P' = DefEdge(P, n_1, n_2, e, label)$ , 若  $n_1$  和  $n_2$  之间原来无任何边, 该操作对标号层次无任何影响, 若  $n_1$  和  $n_2$  之间原来有边  $l$  相连, 则在标号层次中建立一条从  $label$  指向  $\lambda_P(l)$  节点的边. 这些操作对于节点标号或边标号是正规表达式时同样适用, 很显然, 所有的标号层次是树型结构.

### 2.2 查询的定义及实现算法

查询的定义与模式约束图的定义是统一的, 针对模式约束图  $P$  的查询即是当前超媒体数据库中满足  $P$  的所有实例. 从定义 3 可以看出, 模式约束图  $P$  中的节点和边的标号可以由其他模式约束图通过  $DefNode$  和  $DefEdge$  生成, 这样, 所有的模式约束图在一起也组成了层次结构, 有如下定义:

**定义 6.** 分层模式依赖图中的每个节点代表超媒体系统中的一个模式约束图, 其中模式约束图  $P_1$  和  $P_2$  之间

存在  $P_1 \rightarrow P_2$  的边当且仅当  $P_2$  的定义中引用了  $P_1$  中通过 *DefNode* 和 *DefEdge* 操作所生成的节点或边的标号。

很显然,该模式依赖图是 DAG(有向无环图).讨论建立在模式依赖图之上的查询算法如算法 1,由于模式依赖图中所有的模式约束图节点之间是偏序关系,因此需按一定的次序进行模式匹配。

算法 1. 查询实现算法.

输入: 模式依赖图 SDG, 查询  $Q$ . //假定  $Q$  的定义是由集合  $P_Q$  中所有模式约束图中生成的节点标号和边标号组成的

输出:  $Q$  在数据库  $G$  中的所有实例  $I$ .

过程: {在 SDG 中找到以  $P_Q$  中每一个模式约束图为根的子树;合并这些子树,生成一个 SDG 的子图  $O_G$  //

While ( $Q$  在  $O_G$  中的入度  $\neq 0$ ) do

{ find 所有入度为 0 的节点;

对上述节点所代表的模式约束图与  $G$  进行匹配;

将图  $O_G$  中这些节点以及由这些节点出发的边删除;

}

对  $Q$  与  $G$  中的实例进行匹配,生成  $I$ ;

}

在上述算法中,匹配算法是基于子图同构<sup>[7]</sup>的算法,在每一步匹配过程中都可能要用到前面步骤中已经生成的匹配结果,即通过 *DefNode* 和 *DefEdge* 所生成的节点或边已经找到的匹配,可以用来作为下一步子图同构算法的初始条件,这可以明显地减少算法的执行时间.具体的关于子图同构算法的描述及时间复杂度的分析,请见文献[11].

### 3 图形代数

与关系数据库理论中的关系代数类似,我们提出了一种图形代数的概念,它是定义于模式约束图之上的单元或二元运算,用它可以生成更加复杂的模式约束图.在给出具体的图形代数之前,先给出标号兼容的定义.

定义 7. 两个节点(或边)的标号是兼容的,当且仅当这两个标号处于同一棵标号层次树中;而一个模式约束图  $P_1$  同另一个模式约束图  $P_2$  是兼容的,当且仅当首先  $P_1$  与  $P_2$  所对应的无标号图是同构的,其次,在同构的对应节点对和边对之间,它们的标号也是兼容的.

为了简化该模型,我们在图形代数中只考虑验证本模型的关系完备性所必需的  $\{\pi_s, \sigma_s, \cup_s, -, \times_s\}$  这 5 种操作.

$\pi_s$ : 设  $A$  是所要投影的属性节点集合,显然  $A \subseteq N_{P_p}$ , 设  $P' = \pi_s(P)$ ,  $P = (N_p, E_p, \psi_p, \lambda_p, T_p, C_p)$ ,  $P' = (N_{p'}, E_{p'}, \psi_{p'}, \lambda_{p'}, T_{p'}, C_{p'})$ , 其中  $N_{p'} = N_{p_a} + A$ ,  $E_{p'} = (E_{p_p} - \{\text{指向}(N_{p_p} - A)\text{的边}\}) + E_{p_a}$ ,  $\psi_{p'} = \psi_{p_p} - \{(E_p - E_{p'}) \rightarrow N_{p_a} \times (N_{p_p} - A)\}$ ,  $T_{p'} = T_p - \lambda_p(N_{p_p} - A)$ ,  $\lambda_{p'} = \lambda_p - \{(N_{p_p} - A) + (E_p - E_{p'})\} \rightarrow (T_p - T_{p'})$ ,  $C_{p'} = C_p$ . 显然,要投影的属性不能包含在  $C_p$  当中.

$\sigma_s$ : 设  $F$  是对模式约束图  $P$  进行选择的条件,  $P = (N_p, E_p, \psi_p, \lambda_p, T_p, C_p)$ ,  $P' = \sigma_s(P)$ , 则  $P' = (N_p, E_p, \psi_p, \lambda_p, T_p, C_p \wedge F)$ .

$\cup_s$ : 设  $P_1$  和  $P_2$  是要进行  $\cup$  操作的两个模式约束图,则要求  $P_1$  与  $P_2$  是兼容的,设  $P = P_1 \cup P_2$ ,  $P_1 = (N_{p_1}, E_{p_1}, \psi_{p_1}, \lambda_{p_1}, T_{p_1}, C_{p_1})$ ,  $P_2 = (N_{p_2}, E_{p_2}, \psi_{p_2}, \lambda_{p_2}, T_{p_2}, C_{p_2})$ ,  $P = (N_p, E_p, \psi_p, \lambda_p, T_p, C_p)$ , 故有  $N_p = N_{p_1}$  (或  $N_{p_2}$ ),  $E_p = E_{p_1}$  (或  $E_{p_2}$ ),  $\psi_p = \psi_{p_1}$  (或  $\psi_{p_2}$ ),  $\lambda_p = (E_p + N_p) \rightarrow T_p$ , 而  $T_p$  中每个节点或边的标号是  $T_1 | T_2$  的形式,其中  $T_1, T_2$  为两个同构的模式约束图中相对应的节点或边的标号,这时,可以通过 *DefNode* 和 *DefEdge* 操作来改变这些正规式的标号,  $C_p = C_{p_1} \vee C_{p_2}$ .

$-$ : 设  $P_1$  和  $P_2$  是要进行  $-$  操作的两个模式约束图,则要求  $P_1$  与  $P_2$  是兼容的,设  $P = P_1 - P_2$ ,  $P_1 = (N_{p_1}, E_{p_1}, \psi_{p_1}, \lambda_{p_1}, T_{p_1}, C_{p_1})$ ,  $P_2 = (N_{p_2}, E_{p_2}, \psi_{p_2}, \lambda_{p_2}, T_{p_2}, C_{p_2})$ ,  $P = (N_p, E_p, \psi_p, \lambda_p, T_p, C_p)$ , 故有  $N_p = N_{p_1}$ ,  $E_p = E_{p_1}$ ,  $\psi_p = \psi_{p_1}$ ,  $\lambda_p = \lambda_{p_1}$ ,  $T_p = T_{p_1}$ , 而  $C_p = C_{p_1} \wedge (\neg C_{p_2})$ .

$\times_s$ : 设  $P_1$  和  $P_2$  是要进行  $\times$  操作的两个模式约束图,该操作的前提条件是  $P_1$  与  $P_2$  至少有一个兼容的节点对,当存在多个相兼容的节点和边对时,该操作可以让用户指定所要进行  $\times$  操作的节点和边对,设为  $P-S = \{(n_1, n_1') (n_2, n_2') \dots (n_k, n_k')\}$ , 设  $P = P_1 \times P_2$ ,  $P_1 = (N_{p_1}, E_{p_1}, \psi_{p_1}, \lambda_{p_1}, T_{p_1}, C_{p_1})$ ,  $P_2 = (N_{p_2}, E_{p_2}, \psi_{p_2}, \lambda_{p_2}, T_{p_2}, C_{p_2})$ ,  $P = (N_p, E_p, \psi_p, \lambda_p, T_p, C_p)$ , 故有  $N_p = N_{p_1} + (N_{p_2} - \{n_1', n_2', \dots, n_k'\})$ ,  $E_p = E_{p_1} + \{E_{p_2} - \{n_1', n_2', \dots, n_k'\}\}$ ,  $\psi_p = \psi_{p_1} + \psi_{p_2}$ , 对于  $N_p$  或  $E_p$  中不属于  $P-S$  的边 1 或节点  $n$ , 有如果  $n \in N_{p_1}$ , 则  $\lambda_p(n) = \lambda_{p_1}(n)$ , 若  $n \in N_{p_2}$ , 则  $\lambda_p(n) = \lambda_{p_2}(n)$ , 若  $n \in \{n_1, n_2, \dots, n_k\}$ , 则  $\lambda_p(n) = \lambda_{p_1}(n) + \lambda_{p_2}(n)$ . 对于边

也有同样的定义,  $C_p = C_{p1} \wedge C_{p2}$ . 与  $\cup$  操作类似, 我们可以用 *DefNode* 和 *DefEdge* 操作对具备正规表达式标号的节点或边进行标号的重定义.

#### 4 模型的查询能力

我们已经提到过, 查询能力是一个很重要的因素. 首先, 我们来考虑其完备性. 在关系数据库中, 完备性被认为是非常重要和有用的特性. 根据 Codd 最早提出的定义<sup>[8]</sup>, 我们考虑  $(\pi, \sigma, \cup, -, \times)$  5 种操作. 为了将关系数据库模型映射到我们的超媒体数据库模型, 我们给出映射函数  $\xi: DBMS \rightarrow S$ , 对于关系数据库中的每一张表  $t$ , 在本模型中均可以用抽象节点  $n(=\xi(t))$  来表示,  $\lambda_S(n)$  的值即是  $t$  的表名, 而每个  $t$  中的域均可用与  $n$  相连的属性类型节点来表示, 连接  $n$  与属性类型节点的边  $l$  的  $\lambda_S(l)$  的值即是  $t$  中的域名, 属性类型节点的标号即为相对应域的定义域; 在关系数据库中, 表和表之间的关联是通过主键和外键来实现的, 即若一个关系  $x$  中存在属性是另一关系  $y$  的键值, 则在模式约束图中建立一条从  $\xi(y)$  节点指向  $\xi(x)$  节点的边, 该边建立了这两个关系之间的关系. 下面分别来看这 5 种关系代数运算符:

$\pi$ : 显然  $\pi$  是  $\pi_S$  的一个特例, 这时的模式约束图即为表所对应的抽象节点以及所有与之相连的属性类型节点.

$\sigma$ : 与  $\pi$  一样,  $\sigma$  也是  $\sigma_S$  的一个特例.

$\cup$ : 两个兼容表的并显然对应着两张表所对应的模式约束图之间的  $\cup_S$  操作.

$-$ : 两个兼容表的差显然对应着两张表所对应的模式约束图之间的  $-_S$  操作.

$\times$ : 对于连接操作, 则需要简单地建立一个以 *join* 为标号连接两张表所对应的模式约束图的边, 连接的条件可以用  $C_p$  来表示.

以上 5 种运算所生成的模式约束图在超媒体数据库中的所有实例即对应着关系代数的运算结果, 每一个实例即相当于关系代数运算结果表中的一个元组. 由上可见, 我们所提出的查询模型根据 Codd 的定义是关系完备的, 即它至少具备关系代数或关系演算所拥有的一阶谓词查询能力. 由于本模型中的  $C_p$  可以表达任意复杂的算术运算, 单这一点就可以证明它的查询能力要大于一阶谓词逻辑 (first order logic, 简称 FOL). 其次, 本模型中节点或边的标号可以为  $T^*$  的形式, 加上  $C_p$  所表示的约束可以用来进行通用传递闭包 (transitive closure, 简称 TC) 的计算. 综上所述, 本模型的查询能力不小于 FOL+TC.

#### 5 与相关工作比较

目前已经提出了不少以图为基础的超媒体查询模型<sup>[9-15]</sup>, 与本模型一样, 这些模型的共同点是以模式匹配为核心的超媒体结构查询. 如 Gram<sup>[9,10]</sup> 模型的基础是 HyperWalk, Gram 代数中为了保证其计算的封闭性, 计算结果必须为 HyperWalk, 这就限定了它的查询能力 ( $<FOL+TC$ ). 与本模型比较相似的有 Consens 所提出的 GraphLog 查询语言, 它通过 Distinguished Edge 来进行定义于查询之上的查询. 实际上, Distinguished Edge 可以用本模型的 *DefEdge* 操作来实现, 而且 GraphLog 实现的办法是将查询转换成谓词逻辑程序, 而本模型所用的是图形同构的概念. 此外 GraphLog 的查询图只能表示有限的算术运算, 本模型的约束条件可以表达任意复杂的算术运算与逻辑运算. 当然, 本模型中并没有像 GraphLog 那样提供了聚集函数和路径总结, 这是 GraphLog 强于我们模型的一个方面, 有待于在本模型的基础上实现. GOOD (graph-oriented object data) 模型<sup>[14]</sup> 是另一个相近的查询模型, 它的 *addnode* 和 *addedge* 操作与本模型的 *DefNode* 和 *DefEdge* 操作类似, 但是 GOOD 模型的主要目的是进行以图为基础的对象操作, 包括重结构化、修改、模式操作, 而不是查询. 与 GOOD 模型不同的是, 为了支持知识的可重用性, 我们使用了分层模式依赖图的概念, 这更降低了用户的认知负载, 也减少了迷路现象的发生 (针对超媒体系统). 另一个值得一提的模型是 MORE<sup>[15]</sup>, 虽然所基于的概念模式是相近的, 但是它提出的可视化信息检索操作是我们所提出的图形代数的一个实例 (即是建立在原始的数据库模式上的代数操作).

综上, 通过数据库模式、模式约束图以及分层的模式依赖图等概念的引入, 我们提出了一个基于模式匹配的超媒体查询模型, 并在此基础上给出了形式化的图形代数的定义, 证明了该模型的查询能力. 一方面, 针对超媒体数据库的特点, 我们提出了图形化的操作界面, 以降低用户的认知负载; 另一方面, 通过图形代数的提出, 增

强了该模型的查询能力.这些也是本系统的设计初衷.一些其他的工作如内容查询<sup>[16]</sup>(相对结构查询)可以结合按内容查询的方法,通过我们的模式约束来实现.另外,聚集函数、抽象<sup>[11]</sup>等概念的设计,以及如何将该模型与传统的查询理论结合起来从而支持不确定的查询或推理,如对边的加权、根据不同类型的用户提供不同的存取模式等,都是一些值得研究的方向.

#### References:

- [1] Halasz, F. Reflections on notecards: seven issues for the next generation of hypermedia systems. *Communications of the ACM*, 1988, 31(7):836~852.
- [2] Li, Guang-ya, Zhou, Xue-hai, Gong, Yu-chang, *et al.* A semantic network based open hypermedia system architecture. *Computer Science*, 1998,25(3):48~51 (in Chinese).
- [3] Celentano, A., Fugini, M.G., Pozzi, S. Knowledge-Based document retrieval in office environments: the Kabiria system. *ACM Transactions on Information Systems*, 1995,13(3):237~268.
- [4] Halasz, F., Schwartz, M. The dexter hypertext reference model. *Communications of the ACM*, 1994,37(2):30~39.
- [5] Li, Guang-ya, Zhou, Xue-hai, Gong, Yu-chang, *et al.* Research on open hypermedia system. *Computer Science*, 1997,24(4):38~42 (in Chinese).
- [6] Mayfield, J. SNITCH: augmenting hypertext documents with a semantic net. *International Journal of Intelligent & Cooperative Information Systems*, 1993,2(3):335~351.
- [7] Eppstein, D. Subgraph isomorphism in planar graphs and related problems. Tech-Report 94-25, Irvine: University of California. 1994.
- [8] Codd, E.F. *Relational Completeness of Data Base Sublanguages*. Nj: Prentice-Hall, 1972.
- [9] Amman, B., Scholl, M. Gram: a graph data model and query language. In: Amman, B., Cedex, L.C., eds. *Proceedings of the ACM Hypertext'92*. ACM press, 1992. 201~211.
- [10] Amann, B., Scholl, M. Schema-Based authoring and querying of large hypertexts. *International Journal of Human-Computer Studies*, 1995,43(3):281~299.
- [11] Gemis, M., Paredaens, J., Thyssens, I., *et al.* Concepts for graph-oriented object manipulation. In: *Advances in Database Technology. Lecture Notes in Computer Science 580*, 1992. 21~38.
- [12] Beeri, C., Kornatzky, Y. A logical query language for hypertext systems. In: Rizk, A., Streitz, N.A., André, J., eds. *Hypertext: Concepts, Systems, and Applications, Proceedings of the ECHT'90 European Conference on Hypertext*, INRIA. Cambridge: Cambridge University Press, 1990. 67~80.
- [13] Catarci, T. On the expressive power of graphical query languages. In: Knuth, E., Wagner, L.M., eds. *Visual Database Systems II*. Amsterdam: North-Holland, 1991. 411~421.
- [14] Gyssens, M., Paredaens, J., van Gucht, D.. A graph-oriented object model for database end-user interfaces. In: Garcia-Molina, H., Jagadish, H.V., eds. *Proceedings of the 1990 ACM SIGMOD International Conference on Management of Data*. Atlantic, NJ: ACM press, 1990. 24~33.
- [15] Lucarella, D., Zanzi, A. A visual retrieval environment for hypermedia information systems. *ACM Transactions on Information Systems*, 1996,14(1):3~29.
- [16] Zhou, Xue-hai, Li, Guang-ya, Zhao, Zhen-xi. A data model for image database systems with content-based retrieval. *Journal of Software*, 1998,9(3):186~189 (in Chinese).

#### 附中文参考文献:

- [2] 李光亚,周学海,龚育昌,等.一种基于语义网络的开放式超媒体系统结构. *计算机科学*,1998,25(3):48~51.
- [5] 李光亚,周学海,龚育昌,等.超媒体系统的开放性探析. *计算机科学*,1997,24(4):38~42.
- [16] 周学海,李光亚,赵振西.按内容检索的图像数据库系统数据模型. *软件学报*,1998,9(3):186~189.

## A Hypermedia Retrieval Model Based on Pattern Matching\*

ZHOU Xue-hai<sup>1</sup>, LI Guang-ya<sup>2</sup>, LI Xi<sup>1</sup>, GONG Yu-chang<sup>1</sup>, ZHAO Zhen-xi<sup>1</sup>

<sup>1</sup>(Department of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China);

<sup>2</sup>(Wonders Information Co. Ltd, Shanghai 200233, China)

E-mail: xhzhou@ustc.edu.cn

http://www.ustc.edu.cn

**Abstract:** Built on the semantic network based hypermedia model, a hypermedia retrieval model based on pattern matching is proposed in this paper by introducing the concepts of database schema, model constraint graph and stratified schema dependency graph. Furthermore the formal definition of graph algebra is given out, which guaranteed the model's retrieval ability. On the one hand, this model can decrease the users' cognitive overhead. On the other hand, it guarantees the expressive capability of the retrieval.

**Key words:** hypermedia systems; database schema; model constraint graph; graph algebra; relational completeness.

\* Received September 20, 2000; accepted April 17, 2001

Supported by the National Natural Science Foundation of China under Grant No.69503005; the Natural Science Foundation of Anhui Province of China under Grant No.99043302

### 敬告作者

《软件学报》创刊以来,蒙国内外学术界厚爱,收到许多高质量的稿件,其中不少在发表后读者反映良好,认为本刊保持了较高的学术水平.但也有些稿件因不符合本刊的要求而未能通过审稿.为了帮助广大作者尽快地把他们的优秀研究成果发表在我刊上,特此列举一些审稿过程中经常遇到的问题,请作者投稿时尽量予以避免,以利大作的发表.

1. 读书偶有所得,即匆忙成文,未曾注意该领域或该研究课题国内外近年来的发展情况,不引用和不比较最近文献中的同类结果,有的甚至完全不列参考文献.
2. 做了一个软件系统,详尽描述该系统的各个方面,如像工作报告,但采用的基本上是成熟技术,未与国内外同类系统比较,没有指出该系统在技术上哪几点比别人先进,为什么先进.一般来说,技术上没有创新的软件系统是没有发表价值的.
3. 提出一个新的算法,认为该算法优越,但既未从数学上证明比现有的其他算法好(例如降低复杂性),也没有用实验数据来进行对比,难以令人信服.
4. 提出一个大型软件系统的总体设想,但很粗糙,而且还没有(哪怕是部分的)实现,很难证明该设想是现实的、可行的、先进的.
5. 介绍一个现有的软件开发方法,或一个现有软件产品的结构(非作者本人开发,往往是引进的,或公司产品),甚至某一软件的使用方法.本刊不登载高级科普文章,不支持在论文中引进广告色彩.
6. 提出对软件开发或软件产业的某种观点,泛泛而论,技术含量少.本刊目前暂不开办软件论坛,只发表学术文章,但也欢迎材料丰富,反映现代软件理论或技术发展,并含有作者精辟见解的某一领域的综述文章.
7. 介绍作者做的把软件技术应用于某个领域的工作,但其中软件技术含量太少,甚至微不足道,大部分内容是其他专业领域的技术细节,这类文章宜改投其他专业刊物.
8. 其主要内容已经在其他正式学术刊物上或在正式出版物中发表过的文章,一稿多投的文章,经退稿后未作本质修改换名重投的文章.

本刊热情欢迎国内外科技界对《软件学报》踊跃投稿.为了和大家一起办好本刊,特提出以上各点敬告作者,并且欢迎广大作者 and 读者对本刊的各个方面,尤其是对论文的质量多多提出批评建议.