基于部分匹配的 XML 文本文档向量检索模型

吴 劲,陈泽琳
(广州华南理工大学计算机科学与工程学院,广东广州 510640)

摘 要: 本文提出了部分匹配模式的 XML 文本文档向量检索模型,给出了 XML 文本文档树以及子文档树的向量表示和查询以及子查询的向量表示,并由此提出了查询中的祖先-后代关系映射到文档中的祖先-后代关系的部分匹配模式的检索方式,给出了基于此匹配处理过程的相似度计算,以判断文档与查询的相关程度.在构造的检索原型系统中的实验表明,该检索模型具有较好的查全率和查准率.

关键词: XML 文本文档; 向量检索模型; 信息检索; 部分匹配模式; 查询

中图分类号: TP391.72; TP393.09 **文献标识码:** A **文章编号:** 0372-2112 (2002) 12A-2169-03

Vector Retrieval Modeling Using Partial Match Pattern for Text-rich XML Documents

WU Jin, CHEN Ze-lin
(*Dept. of Computer Science & Engineering, South China Univ. of Tech, Guangzhou, Guangdong* 510640, *China*)

Abstract: This paper presents the vector retrieval modeling using the partial match pattern for text-rich XML documents. The vector representation of the tree of text-rich XML documents and sub-documents and the tree of queries and sub-queries are decribed. The partial match pattern is mapping from the relations of ancestors and descendants for a query to the relations of ancestors and descendants for documents. The paper calculates the similarity in the processing of the partial match. The results of experiemental analyses with different collections and queries show that it is effective in the completeness search and the precision search.

Key words: text-rich XML document; vector retrieval modeling; information retrieval; partial match pattern; query

1 引言

XML(eXtensible Markup Language)是由 SGML(Standard Generalized Maarkup Language, ISO8879)衍生而来的,是面向语义的标记语言,可扩展性的自定义标签增强了信息语义的描述能力.通过标签结构化的层次表示,XML 将大量信息组织成为具有确定意义的整体——XML 文档.

另一方面,自然语言(文本)一直以来是人们描述与交换信息的基本手段,基于 XML 强大的语义描述能力,在未来的互联网世界中,将出现大量使用 XML 组织的文本信息——XML 文本文档.本文将 XML 文档中标签之间的层次关系称为结构.显而易见,XML 文本文档的语义信息同时存在于文档的文本与结构之中.

近几年,面向 XML 文本文档检索的研究已取得了一定的研究成果.A.Theobald 和 G.Weikum[1]设计了 XXL 查询语言以及相应的检索方法,他们通过在 XML-QL 中加入相似比较的操作,从而提供非精确匹配,并由此计算文档信息与提交查询的相关程度.Yoshihiko Hayashi[2]介绍了一个 XML 文本文档检索系统的实现技术,由于作者认为任意结构的 XML 文档信息检索是难于实现的,文中限定了文档结构信息的索引,限定了提交查询中的结构信息,实现了一个适用于专业文档检索的 XML 文本文档检索系统.Norbert Fubr[3]提出了一种数据检索与文本检索相结合的检索语言——XIRQL,并在基于逻辑的概率检索模型上对 XML 文本信息进行分析,得出了相关概率.在文中作者分析了 XML 的应用环境,在 XQL 的基础上加入四个信息检索的特征:权重计算和排序、面向相关的检索、语义相对和模糊谓词,给出了 XIRQL.文献[5]给出了一种检索最匹配子树的 XML 数据检索方法.

本文在 XML 文本文档检索模型中采用部分匹配模式,结合传统信息检索技术,给出相似度的计算,提出了一种新的面向 XML 文本文档的检索模型.

2 XML 文本文档检索模型

2.1 XML 文本文档与查询描述

为了简化 XML 文本文档检索的模型设计和系统复杂度,这里不考虑 XML 中的引用语义,并忽略兄弟节点之间关系.虽然这样的限制比较苛刻,但仍符合 XML 基本的层次化信息组织形式.将 XML 文本文档描述为一棵树,称之为 XML 文本文档树[5]:

$$d=(r, SN, TN, T, \prec_{snh}, \tau, \sigma)$$

收稿日期:2002-06-06;修回日期:2002-11-21

其中：r 为树中的虚拟根节点，表示整个文档；SN 为树中结构节点(元素或元素属性)的集合；TN 为树中文本节点(元素文本值或者属性文本值)的集合；T 为树中所有结构节点的类型(元素标签)的集合；$\prec_{snh}$ 描述树中结构节点之间的父亲儿子关系，设有 $sn_i, sn_j \in SN$，且 sn_i 是 sn_j 的儿子节点，则有 $sn_i \prec_{snh} sn_j$；τ 为从 SN 到 T 的映射；σ 为从 SN 到 $TN \cup \{NULL\}$ 的映射.

定义 XML 文本文档树中的路径 $p = (sn_1, sn_2, sn_3, \cdots, sn_m)$ 表示结构节点 sn_1 到结构节点 sn_m 之间的一条通路，描述了 sn_1 与 sn_m 之间存在的祖先-后代关系；$head(p)$ 和 $tail(p)$ 分别表示路径的起点与终点；节点之间的距离定义为：$dist(x, y) = |p| - 1$，其中 $head(p) = x$，$tail(p) = y$，$|p|$ 表示路径中节点的个数.

在 XML 文本文档树中，一个结构节点的子孙节点的集合为：

$$DESC(sn) = \{sn_i \mid \exists p((head(p) = sn) \wedge (tail(p) = sn_i))\}$$

一个结构节点的文本信息集合为：

$$CONTENT(sn) = \{tn_j \mid (sn_i \in DESC(sn) \cup \{sn\}) \wedge (tn_j = \sigma(sn_i)) \wedge (\sigma(sn_i) \neq NULL)\}$$

子文档定义为：$sd_n = (n, SN_n, TN_n)$，其中 $n \in SN$，且 $SN_n = DESC(n)$，$TN_n = \{tn \mid (tn = \sigma(sn)) \wedge (sn \in SN_n)\}$，即子文档 sd_n 为文档树中以节点 n 为根的子树，而子文档的类型为节点 n 的类型：$TYPE(sd) = \tau(n)$；子文档的内容为：$CONTENT(sd_n) = CONTENT(n)$.

与 XML 文档描述类似，我们将提交的含有结构信息的查询描述为一棵查询树，查询树的每个节点为一个子查询，且在提交的查询中必须有文本信息的描述.

2.2 XML 文本文档向量检索模型

检索中，从下至上逐级处理查询树上的每一个子查询.同时利用传统检索技术中的向量检索，计算子查询与子文档之间的相似度(SIM)，并最终求出文档与整个查询之间的相似度.

首先，子查询 q 的向量表示为：$q = (sq_1, sq_2, sq_3, \cdots, sq_m)$，且有 $sq_i \in DESC(q)$；对应的子文档 sd 的向量表示为：$sd = (result(sq_1), result(sq_2), \cdots, result(sq_m))$，其中 $result(sq)$ 表示子查询 sq 的检索匹配结果：

$$result(sq) = \{n \mid (n \in SN) \wedge ((\tau(n) = sq) \vee (sq \in CONTENT(n))) \wedge relD(n, sq)\}$$

其中，$relD(n, sq)$ 表示检索出的结构节点 n 应满足如下关系，即：

$$relD(n, sq) = \exists n_d \exists sq_d((n_d \in DESC(n)) \wedge (sq_d \in DESC(sq)) \wedge (n_d \in result(sq_d)))$$

从检索匹配结果集合 $result(sq)$ 中可以看出，本文提出的 XML 文本文档检索模型提交的查询，在 XML 文本文档匹配时并不需要完全满足，即检索出的结果文档树可以不完全包含查询树.因此，只要结果节点 n 之间的关系满足任何子查询中的祖先-后代关系，就被认为是满足查询要求的，从而实现了“部分匹配”的过程.

检索模型中对查询树的处理过程是对每一个被访问子查询(向量)在文档树中按后根遍历的顺序构建其对应的结果子文档(向量)，同时计算子查询与子文档的相似度.这里定义相似度函数为 $SIM(sd, q)$，表示子文档 sd 与其匹配的子查询 q 之间的相似度.为了计算相似度函数，我们需要进一步确定向量中元素的取值.

根据查询 q 与其后代子查询在查询树中的距离，确定查询向量各元素的取值.为了简化原型系统的实现，本文的取值为距离的倒数，即查询向量元素的取值为：$w_{qi} = QW(dist(q, sq_i)) = 1/dist(q, sq_i)$，表示当查询之间的距离越大，则满足此下文关系的子文档匹配查询的程度越小，即与用户查询的相关程度越低.

子文档向量 sd 在属性空间中各向量元素的取值与对应的子查询匹配结果集相关.设 $w_{result(sqi)}$ 为子文档向量各元素在子查询 sq_i 上的取值，则子文档向量可以表示为：

$$sd = (w_{result(sq1)}, w_{result(sq2)}, \cdots, w_{result(sqm)})$$

对于关键词子查询，用传统文本分析技术求出子文本向量各元素的值.根据 XML 文本文档的特殊性，我们对文[4]中的计算方法修改为：

$$w_{result(qt)} = \frac{f_{qt}}{|CONTENT(sd_{sup})|} \cdot \log \frac{n_{sup(qt)}}{N_{sup}}$$

其中，sd_{sup} 表示包含关键词 qt 且满足先前子查询的子文档，f_{qt} 表示在文本信息集合 $CONTENT(sd_{sup})$ 中关键词 qt 出现的频率，$|CONTENT(sd_{sup})|$ 表示子文档 sd_{sup} 中包含的文本信息的总长度，$n_{sup(qt)}$ 表示包含关键词 qt 且匹配查询 q_{sup} 的子文档数量，N_{sup} 表示所有匹配查询 q_{sup} 的子文档数量.

最后，用余弦匹配系数法得到子文档与子查询的相似度：

$$SIM(sd, q) = \sum_{i=1}^{n} w_{qi} \cdot w_{result(qi)} \Big/ \sqrt{\sum_{i=1}^{n} w_{qi}^2 \cdot \sum_{i=1}^{n} w_{result(qi)}^2}$$

在从下至上处理子查询的过程中，通过上述方式得到文档与提交查询之间的相似度，再按相似度递减顺序输出结果.

3 原型系统与实验

用 Visual C ++6.0 和 Microsoft SQL Server 7.0 构造了上述检索模型的一个原型检索系统，并在以下两个 XML 文本文档集合上进行了检索实验.

(1)在传统文本文档检索测试集(OHSUMED.9[6])基础上，对文档集合进行适当的变换，使其为 XML 文档集合，该文档集合包含 74,337 篇医学文章的参考信息.该集合的特点是数量大，结构单一，含测试用例；

(2)由宗教书籍集合、莎士比亚剧作、ACM SIGMOD 书目信息共同组成的混合 XML 文本文档集合.该集合的特点是数量小，结构复杂，不含测试用例.

实验一的检索结果如图 1 所示，其中 x 坐标轴表示检索的完全性(查全率)，y 轴为检索的准确性(查准率)，$Qterm$ 曲线代表只含关键词的查询的检索曲线，Qab 曲线代表包含一定结构信息的查询的检索曲线，$Qab\&t$ 代表包含更多结构信息的查询的检索曲线，$Qerr$ 代表包含错误结构信息(不存在的结构)的查询的检索曲线.

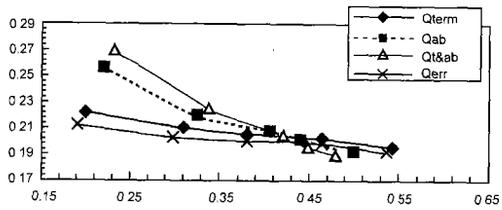

图 1　实验一检索结果分析

在图 1 中,比较了不使用结构信息的查询曲线 *Qterm* 与使用结构信息的查询曲线 *Qab* 与 *Qt&ab*,可以看到,当在查询中加入正确结构的结构信息,则在较低的查全率下面,可以获得比较高的查准率.因为,检索过程为一个文档排序的过程,同时,检索用户按照顺序浏览检索结果,以找到自己需要的文档.那么,在较低查全率下获得较高的查准率,则表示能较快地将与提交查询相关的结果文档检索出来.同时也可以发现,在提交查询中加入更加丰富的结构信息,则检索将获得更高的查准率(见曲线 *Qt&ab*).第四类查询为文档集合中不存在查询结构,但检索模型仍然能够返回一定的结果,而不是 0 结果返回,但结果比不使用任何结构信息的检索结果要差(见曲线 *Qterm*),因为在检索模型中对不符合用户提交的查询结构赋予较低的权重.

实验二中分别提交了如下查询:(1) *Q1*: *title*['*testament*'] (2) *Q2*: *book*[*title*['*testament*']'*Jesus*'](3) *Q3*: *author*['*king*'] (4) *Q4*: *article*[*author*['*king*'],'*database*'](5) *Q5*: *persona*['*king*']

查询结果如表 1 所示,粗体字表示命中.在结果中同样可以发现,在查询中包含准确的、丰富的结构信息,能有效地提高检索的查准率;错误的结构信息并不会导致查全率的过度下降(0 检索结果).

表 1　混合文档集测试结果(*testing result on mixed document set*)

查询 / 结果	*Q1*	*Q2*	*Q3*	*Q4*	*Q5*
查准率	0.2	0.2	0.5	0.5	1
结果 1	*Nt. xml*	*Bom*	*MOD212*	*MOD212*	*Dream*
结果 2	*Ot. xml*	*Nt. xml*	*MOD192*	*MOD192*	*LLL*
结果 3	*Timode*	*Hen_vi*	*MOD173*	*MOD202*	*Tempest*
结果 4	*Pericle*	*Ot. xml*	*MOD163*	*MOD173*	*Macbeth*
结果 5	*Quran..*	*Hen_v*	*MOD144*	*MOD182*	*Cymblin*
结果 6	*Bom*	*R_and_J*	*Bom*	*MOD163*	*Pericle*
结果 7	*Rich_ii*	*Rich_iii*	*Ot. xml*	*MOD144*	*Win*
结果 8	*J_casear*	*Hen_iv*	*Two_gen*	*MOD241*	*Hen_vi*
结果 9	*Hen_vi*	*Rich_ii*	*Tempest*	*MOD142*	*Hen_iv*
结果 10	*Hen_v*	*Hen_vi2*	*R and J*	*MOD152*	*Hen_iv2*

4　结论

通过在原型检索系统上进行的检索实验,初步验证了模型的有效性,获得了较高的检索查准率,同时避免了查全率的过度下降.

但在实验中还发现,当提交查询中出现错误(或不存在)的结构信息时,其检索的查准率比不使用结构信息时更差.另外,检索模型中的检索语言没有考虑布尔运算,使语义表达能力过于单薄.这些问题都有待于在日后的研究工作中进一步解决.

参考文献:

[1] A Theobald, G Weikum. Adding relevance to XML[A]. In proceedings of 3rd International Workshop on Web and Databases[C]. Dallus, USA, 2000.

[2] Yoshihiko Hayashi, Junji Tomita, et al. Searching text-rich XML documents with relevance ranking[A]. ACM SIGIR 2000 Workshop on XML and Information Retrieval[C]. Athens, 2000.

[3] Norbert Fuhr, Kai Grobjohann. XIRQL: A query language for information retrieval in XML documents[A]. In Proceedings of the 24th Annual International Conference on Research and development in Information Retrieval[C]. New York, USA, 2001. 172 – 180.

[4] Lee, J H. Combining multiple evidence from different properties of weighting schemes[A]. Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval[C]. 1995. 180 – 188.

[5] Torsten Schlieder. Similarity search in XML data using cost-based query transformations[A]. In Proceedings of the Fourth International Workshop on the Web and Databases (WebDB)[C]. Santa Barbara, USA, 2001.

[6] Hersh WR, Buckley C and Leone TJ. OHSUMED: An interactive retrieval evaluation and new large test collection for research[A]. Proceedings of the 17th Annual ACM SIGIR Conference[C]. 1994. 192 – 201.

[7] http://www.dcs.gla.ac.uk/idom/ir_resources/linguistic_utils/stop_words[DB/OL].

作者简介:

吴　劲　男,1976 年 12 月出生于广东省广州市,2002 年于华南理工大学获硕士学位,现在广州市电信局工作,主要从事信息管理与信息检索的研究与开发工作.

陈泽琳　女,1962 年 6 月出生于哈尔滨,副教授,华南理工大学计算机学院硕士导师,主要研究方向为信息组织与检索、图形与流媒体的协同工作技术等.