XML挖掘：聚类、分类与信息提取

潘有能 著

ZHEJIANG UNIVERSITY PRESS
浙江大学出版社
全国百佳图书出版单位

国家自然科学基金资助项目

XML 挖掘：聚类、分类与信息提取

潘有能 著

ZHEJIANG UNIVERSITY PRESS
浙江大学出版社

图书在版编目(CIP)数据

XML 挖掘:聚类、分类与信息提取 / 潘有能著. —
杭州:浙江大学出版社,2012.7
ISBN 978-7-308-10254-4

Ⅰ.①X… Ⅱ.①潘… Ⅲ.①可扩充语言—程序设计
Ⅳ.①TP312

中国版本图书馆 CIP 数据核字(2012)第 159122 号

XML 挖掘:聚类、分类与信息提取

潘有能 著

责任编辑 吴昌雷
封面设计 刘依群
出版发行 浙江大学出版社
(杭州市天目山路 148 号 邮政编码 310007)
(网址:http://www.zjupress.com)
排 版 浙江时代出版服务有限公司
印 刷 浙江云广印业有限公司
开 本 710mm×1000mm 1/16
印 张 10
字 数 196 千
版 印 次 2012 年 7 月第 1 版 2012 年 7 月第 1 次印刷
书 号 ISBN 978-7-308-10254-4
定 价 32.00 元

前　言

XML是W3C于1998年正式推出的一种标记语言，自发布之日起，XML就以其良好的可扩展性受到业界的普遍欢迎和支持，逐渐成为Web上的通用语言，在数据交换、Web服务、内容管理、Web集成等方面得到了重要应用。随着XML应用的日益广泛，XML文档的内容和文档之间的关系结构也日趋复杂。聚类和分类等数据挖掘技术不但可以增强网络中XML文档的组织性，从而为网络信息资源的搜集、整理及检索利用提供良好的技术支持，还可以在海量网络信息中发现XML文档间隐含的知识，确定XML文档内部标记的真实语义信息，为本体论和语义网的发展奠定坚实基础，因此具有较大研究意义。

本专著是在作者作为负责人的国家自然科学基金项目“基于标记树的XML文档自动聚类和分类研究”(批准号：70803046，2009.1—2011.12)研究成果，项目执行期间课题组成员已发表多篇期刊论文，这些成果为本专著的写作奠定了坚实基础。

专著内容分为8章，第1章先对XML和数据挖掘技术作简要介绍。在对XML文档进行挖掘之前，需要先进行文档解析及文档标记消歧，即为第2章的内容。第3章和第4章分别介绍XML挖掘的两项主要功能：聚类与分类。和HTML中的超链接一样，XML文档之间也具有链接性，第5章介绍利用链接挖掘XML文档间结构的方法。针对XML文档的查询、检索以及信息提取有利于用户准确、快速、有效地利用XML文档，本书的第6章即讨论XML查询与信息提取技术；第7章和第8章则介绍基于XML数据挖掘建模、知识表示以及

Web 日志挖掘。

本专著的部分内容来自于作者的博士论文，另外，作者指导的硕士研究生滕海明对本书的第 2 章和第 3 章有所贡献。

潘有能

2012 年 2 月

目 录

第1章 XML与数据挖掘概述

XML(Extensible Markup Language,可扩展标记语言)是近年来得到广泛应用的一种基于 Internet 的元数据置标语言。自 W3C(World Wide Web Consortium)于 1998 年发布 XML 1.0 以来,XML 以其简单、可扩展、开放、通用等特点逐渐成为互联网数据表示和数据交换的标准。随着 XML 的广泛应用,网上 XML 文档的数量呈现出爆炸性增长的态势。面对日渐增多的 XML 文档,人们迫切需要从中发现有价值的信息和知识。数据挖掘技术在 XML 文档中的应用从而成为当前的研究热点之一。

1.1 XML

XML 是 W3C 制定的一种可扩展元置标语言,与 HTML 一样,XML 也源自于 SGML(Standard Generalized Markup Language),是 SGML 的一个应用子集。XML 集 SGML 和 HTML 的优势于一身,已经逐渐成为 Web 上的通用语言。

1.1.1 XML 的发展与特点

1. XML 的发展

1996 年夏天,Jon Bosak 在 W3C 中组织了一个关于在 Web 上使用 SGML 的委员会。同年 10 月,他们完成了 SGML 简化版本的初始设计。这个简化版本保留了 SGML 的主要实用功能(特别是与 Web 信息发布有关的功能),同时大大降低了 SGML 的复杂性,这就是 XML。1998 年 2 月,W3C 正式推出了 XML 1.0,得到了 Microsoft、Sun 等浏览器生产商和网络管理联盟的支持,并在使用中得到了各方肯定,给 Web 应用注入了新的活力。2004 年 2 月,XML 1.1 成为 W3C 的推荐标准。

XML 目前的发展已经远远超过了 1996 年最初制定标准时的范围,其应用涵盖了从通信到计算机技术的每一个角落。

2. XML 的特点

作为新一代的 Web 语言，XML 具有通用性、可扩展性、易用性、自描述性、异构性、开放性等特点。

(1)通用性。首先，XML 可以在多种系统平台上使用，两大主流浏览器 IE 和 Netscape 中都加入了对 XML 的支持；其次，XML 支持绝大部分字符编码的标准，因此可以在各个国家不同的语言环境中使用；再次，XML 可以用多种工具进行解释，因为 XML 文档具有良好的结构，内容提供商在定义了自己的 DTD (Document Type Definition，文档类型定义)之后，可以在别的语法分析器的基础上做一些改动，以很小的代价建立自己的语法分析器。

(2)可扩展性。XML 和 HTML 最大的区别就在于 XML 是可扩展的，突破了 HTML 中固定标记集合的约束，用户既可以根据需要定义自己的标记集来描述文档中的数据元素，也可以使用几个附加的标准对 XML 进行扩展，这些附加标准可以向核心的 XML 功能集增加样式、链接、参照等功能。在此基础上，可以定义各个不同行业和领域的 XML 规范，用于横向和纵向的信息交流和数据传输。

(3)易用性。作为 SGML 的简化产物，XML 去掉了 SGML 中很少用到的、多余的、执行困难的部分，SGML 最权威的参考手册“*The SGML Handbook*”有 600 页之厚，而 XML 的规范只有 40 页，这使得开发者花很少的时间就能掌握 XML 规范，迅速开始工作。另外，XML 采用的是一种层次化的嵌套结构，管理起来非常方便，在需要增加子项或对子项进行更深层次的细化时，开发者只需要对相关部分进行修改和扩充，而不必对结构本身做大的修改。

(4)自描述性。XML 的另一个重要特性就是可以提供自描述信息，如安全性、语言、作者、文件内容等，尽管这不是必需的，但带有自描述信息可以增强 XML 文件的可检索性，更有利于数据的交换和处理，以达到资源共享的目的。

(5)异构性。在现有的技术条件下，Web 应用仍然不能满足异质数据库之间数据交换的需要，要求所有的机构和人员使用同一数据库，以求无缝实现数据共享，也是不现实的。由于 XML 本身具有的可扩展性和自描述性，我们就可以采用一种统一的数据输入输出格式，从而将来自于不同数据库和应用系统中的数据进行整合，以便于更好地应用和交流。

(6)开放性。XML 的开放性主要体现在两个方面：首先，XML 标准自身在 Web 上是完全开放的，可以免费获取和使用；其次，XML 文档也具有较强的开放性，对于一个结构良好的 XML 文档，任何开发人员和机构都可以进行语法分析，如果提供了 DTD，则不但可以立刻了解整个 XML 文档的体系结构，还可以对这个文档进行校验。当然，开发者也可以用特定方式创建便于自身应用的 XML，或以自己的方式为数据加密，但这样也就失去了使用 XML 的意义，开放

性正是 XML 最大的优点之一。

3. XML 和 SGML、HTML 的比较

与 HTML 一样，XML 也源自于 SGML，HTML 是 SGML 的一个应用，它使用 SGML 的规则定义一组元素类型和实体，它们将以标签的形式被记录在文档中。实际上，HTML 只不过是 SGML 的一些文档类型定义：一组 SGML 元素、属性、实体的声明和表述内容的定义，以及使用说明。当然，在实际应用中，许多浏览器支持 W3C 的 HTML 标准以外的一些标记，但其中很多不为其他浏览器支持。与 HTML 不同，XML 不是 SGML 的一个应用，而是 SGML 的一个子集。SGML 提供了多种方法来对标记语言的各个方面进行定义，XML 保留了其中的一些，去掉了那些过时的、多余的部分。例如，在 SGML 中，除了通常使用的尖括号，还可以指定其他的字符来作为标签分界符，也可以在文档中使用各种特定的字符集，如 ASCII、ISO 8859，甚至于 EBCDIC 字符集等，而在 XML 中，分界符和字符集就被固定下来了：尖括号和 Unicode 字符集。因为 XML 只支持 SGML 的部分而不是全部的类型，所有有效的 XML 文档都是有效的 SGML 文档，但并不是所有有效的 SGML 文档都是有效的 XML 文档，在 XML 和 SGML 之间存在着许多重要的区别。

(1)XML 需要 Unicode 字符集的支持。虽然 XML 文档可以包含其他字符集的内容，但 XML 分析器只保证读取 Unicode 类型的文档(UTF-16 或UTF-8 编码)，因为 ASCII 文档是合法的 UTF-8 数据，对信息提供商来说这个限制并不是什么很严重的问题，何况这是向着建立世界性数据网的目标迈进的一大步。

(2)XML 要求所有的标记都必须出现在文档中，不可以省略。这不但使得 XML 便于学习和使用，而且使得相关软件的开发难度大大降低。

(3)出于简化的目的，XML 排除了许多 SGML 的特性，其中绝大部分是很少能用到的，这种简化不但体现在规则上，同时也体现在 XML 软件中。

(4)在 XML 中，一个语法分析器即使不从 DTD 中读取正式的元素声明，也能可靠地找到所有元素的开始和结束位置，因此 DTD 在 XML 文档中只是一个可选项，而不像 SGML 中的那样是必需的。

虽然都是从 SGML 中衍生出来的，XML 和 HTML 在很多地方还是有着较大的区别。

(1)XML 和 HTML 最大的区别在于 XML 是可扩展的，而 HTML 必须受固定标记集合的约束，用户可以根据自己的需要定义任何标记来描述文档中的数据。和 HTML 相比，XML 为开发者提供了更强的控制文档设计的能力。

(2)HTML 着重描述的是 Web 页面的显示，而 XML 着重描述的是文档的内容。HTML 可以提供丰富多变的显示效果，而 XML 的特长在于描述任意层次结构的数据，赋予原本杂乱无章的信息一种清晰而通用的结构。

(3)HTML 的语法分析器容错性较强，可以接受非法的数据，并能自动从错误中恢复。XML 语法分析器一般拒绝错误格式的输入，最为常见的就是元素的交叉，在下例中，“b”和“i”分别表示两个元素：

XML 和HTML的联系与区别

我们可以看出，“b”和“i”元素交叉了，在 HTML 中，这段文字可以正常显示，效果是：“XML 和 ***HTML的联系与区别***”，这样我们就很难察觉到其中有错误；但是在 XML 语法分析器中，这些错误都将被检测出来并报警。

(4)由于开发者可以在 XML 中定义自己的标记，在页面上显示时 HTML 不再是用户唯一的选择，他们可以选择一种最适合于达到自己目标的标记语言(或标记集合)。当然，HTML 将继续得到广泛应用，但其他的文档类型定义，包括 EAD 和 TEI 等，也将成为用户考虑的对象。某些有着特定需要的用户甚至可以设计一套自己的标记集，提供自己的样式表以告诉浏览器应该如何显示这种 XML 文档。

综上所述，XML 是专门为 Internet 应用而设计的一个 SGML 的子集，像 SGML 一样，它可以用来定义无穷多的标记语言，因此被称为可扩展标记语言，或是元标记语言。而 HTML 是 SGML 定义的一种专门用于 Internet 的标记语言。

1.1.2 XML 文档的结构

与 HTML 不同，XML 对文档结构有严格的规定，只有当一个 XML 文档符合“格式良好(well-formed)”的基本要求时，处理程序才能对它加以分析和处理。这一标准保证了 XML 严密的条理性、逻辑性和良好的结构性，从而大大提高了 XML 应用处理程序处理 XML 数据的准确性和效率。

(1)“格式良好”的 XML 文档和“有效的”的 XML 文档。XML 允许用户建立自己的标记集，但一旦这个标记集建立起来，就必须严格遵守 XML 的语法和标记集的规定。一个 XML 文档必须保证是“格式良好”的，处理程序才可以对它加以分析和处理。所谓的“格式良好”，是指要遵守 XML 标准中的语法规则，具体而言，格式良好的 XML 文档应该满足以下主要条件。

①一个 XML 文档必须包含一个或多个元素，其中有且只有一个包含全部其他元素的元素，一般称为根元素或文档元素，该元素不能有任何部分出现在其他元素中。

②元素必须正确关闭，其中含有数据内容的元素必须有起始标记(<标记名>)和结束标记(</标记名>)。对于不包含数据内容的元素，可以把它的起始标记和结束标记合并为一个空标记，记为“<标记名/>”。

③元素之间可以嵌套，但不能交叉。

④属性值必须加引号。

其他保证文档格式良好的要求还有很多，在此就不一一列举了。检验一个XML文档是否符合格式良好的要求，最简单的方法就是用支持XML的浏览器，如IE 5和Netscape 5，打开当前文档，若能正常显示，表明此文档是格式良好的；若报错，则说明此文档未达到XML标准。

"格式良好"是对XML文档的一个基本要求，除此以外，假如XML文档遵守文档类型定义(DTD)或是Schema中规定的语法规则，则可说这个XML文档是"有效"的，"有效"的XML文档是"格式良好"的XML文档的一个子集，如图1.1所示。其中，X表示所有包含XML的Web资源，W表示"格式良好的"XML文档，V表示"有效的"XML文档。

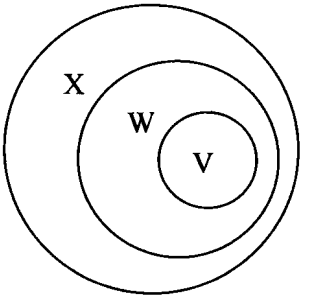

图1.1 XML文档类型

(2)XML文档的结构。XML文档主要包括三个部分：XML声明、处理指令和XML元素。其中XML声明和处理指令都是可选的，而XML元素则是一个格式良好的XML文档所不可或缺的。

①XML声明。XML声明也可以看作是一种特殊的处理指令，其作用是告诉XML处理程序，此文档是按照XML标准对数据进行置标的。XML声明主要包括三个属性：version属性、encoding属性和standalone属性。version属性指明文档采用的XML标准的版本号，是XML声明中必不可少的部分，并且在属性列表中必须排在最前面；encoding属性指出文档中用到的字符集采用的编码标准，一般采用压缩的Unicode编码UTF-8或压缩的UCS编码UTF-16，假如内容中包含简体中文，就必须采用GB2312，包含繁体中文必须采用BIG5，虽然XML标准只要求XML解析器支持UTF-8和UTF-16，但目前比较流行的几种XML解析器，如微软的msxml、James Clark的SP等都提供了对GB2312及BIG5的支持，为我们提供了极大方便；standalone属性表明该文档是否和一个外部的文档类型定义(DTD)相连，其值可为"yes"或"no"，"yes"表明该文档是一个独立的文档，没有配套的DTD，所有的实体声明都必须包含在当前文档中。一个最简单的XML声明如"<? xml version="1.0"? >"，一个完整的XML声明如"<? xml version="1.0" encoding="GB2312" standalone="no"? >"。

②处理指令。处理指令是用于给处理XML文档的应用程序提供信息的一种机制，XML解析器本身可能不对这些信息进行处理，而是把这些信息传递给应用程序，由相关的应用程序进行处理。处理指令的格式为：

<? 处理指令名称 处理指令内容>

需要注意的是，因为在XML声明中已经将"xml"作为系统保留的处理指令

名称，所以其他的处理指令名称不能再用“xml”。下面这个处理指令指定了与XML文档配套使用的样式表的类型为xsl，样式表的文件名为“mystyle.xsl”，其中处理指令名为“xml-stylesheet”。

<? xml-stylesheet type="text/xsl" href="mystyle.xsl">

③XML元素。元素是XML文档内容的基本单元，XML元素包括起始标记、结束标记及标记之间的数据内容，其基本形式为：

<标记>数据内容</标记>

元素实质上是XML对内容的包容器，它不但可以包含字符数据、字符引用、实体引用、处理指令、注释、CDATA等，还可以包含其他的元素，元素之间可以嵌套。此外，XML语法规定，每个文档都必须有一个，而且只能有一个根元素，一般也可以称作文档元素，这个元素是XML文档树中其他所有元素的父元素，而且不能被包含其他任何元素当中。属性是附加在XML元素上的某些信息，它们和元素本身所包含的信息内容有所不同。一个元素可以有任意多个属性，但属性名不能重复，每个属性都包括一个名称/值的组合，属性的名称和属性值之间用等号“=”连接，属性值必须用引号引起来，如：

<商品 类型="电脑" 品牌="联想">

1.1.3 DTD和Schema

DTD是由元素类型、属性、实体和符号等一系列定义组成的一个集合，它明确规定在XML文档中哪些是合法的，以及在什么地方是合法的。DTD既可以嵌入到XML文档中，也可以以独立的文档形式提供。XML文档的内容是以树状的层次结构进行组织的，这种树状结构通过标签和标签之间的关系来表示，而DTD的文档定义部分主要就是对标签的名字和文档的层次结构进行描述，因此，DTD是一个XML文档的基础。

XML的DTD可以分为内部DTD、外部DTD和混合型DTD三种不同的类型。内部DTD指的就是在XML文档内部说明的DTD，主要用于和主体有比较密切联系的部分；外部DTD指的是文档通过实体引用的外部文件，主要用来链接其他不同的需要；而混合型DTD是内部DTD和外部DTD的组合体，当外部DTD和内部DTD因为元素名称相同而发生冲突时，内部DTD的定义将被优先使用。在实际的应用中，外部DTD和混合DTD使用得较多，因为这样一个DTD可以被多个相关文档调用，从而避免了很多重复劳动，提高了工作效率，同时还减少了资源的浪费。

DTD是由标记声明、实体引用及开始和结束标记组成的。其中标记声明主要包括元素类型声明、元素属性声明、实体声明等。元素类型声明是DTD的基本组成部分，同时也是一个XML文档必不可少的声明，通过元素属性声明可以

对元素进行更加详细的定义和描述，实体声明说明了这个 XML 文档中将要使用的实体，一般可以分为内部实体和外部实体、通用实体和参数实体、解析实体和非解析实体等。

XML Schema 是一种模式定义语言，用来描述 XML 文档的结构、内容和语法规范，它由 XML 1.0 自描述，并且使用了命名空间，有丰富的内嵌数据类型和强大的数据结构定义功能，改造并扩展了 DTD 的文档结构定义功能。

和 DTD 相比，XML Schema 具有以下特点。

(1)丰富的数据类型。和 DTD 不同，XML Schema 规范定义了丰富的数据类型，不但包括 string，integer，boolean，time，date 等内嵌数据类型，还提供了定义新的数据类型的能力，如 complexType 和 simpleType。用户可以利用内嵌的数据类型和自定义的数据类型，对 XML 文档的属性和元素值进行定义和规范。

(2)继承和复用。在构造新的 XML Schema 时，可以从已有的 Schema 中继承某些类型，也可以在不需要继承时将获得的类型覆盖掉，从而达到复用的目的。同时，XML Schema 可以将一个 Schema 分解成多个单独的组件，这样在构造新的组件时，就可以直接引用已定义的组件，XML Schema 的继承和复用特性极大地缩短了 XML 软件的开发过程，提高了编程效率，且便于代码维护。

(3)易用性。由于 XML Schema 是用 XML 1.0 自描述的，不需要再去学习一套新的语法规范，易于理解和使用，可以利用 XML 解析器对 Schema 进行解析，用 XML 文档对象模型——SAX 和 DOM 对其进行操作，还可以使用 XSL 进行转换。

1.1.4 Namespace

Namespace(命名空间)通过将 XML 文档中的元素或属性名称与特定的名称域相结合的方式来限定这些元素或属性的含义和用途。所谓名称域，是指事先定义的元素或属性名称的集合，可以由一个 URL 进行参照、确认和连接。

命名空间是通过保留属性来声明的，有两种声明方式：直接定义方式和缺省定义方式。

(1)直接定义方式：xmlns：＜命名空间前缀＞＝＜命名空间名＞

在直接定义方式下，命名空间声明的属性命由两部分组成，即保留属性名前缀“xmlns:”和＜命名空间前缀＞，其中，＜命名空间前缀＞必须是一个合法的 XML 名称。命名空间声明的属性值部分是一个 URI 引用，其功能是区分不同的命名空间，因此被称为命名空间名，它应该具有唯一性和持久性。例如：

```
＜联系人:联系人列表 xmlns:联系人 = "http://cssci.nju.edu.cn/research/联系人列表.dtd"＞
```

(2)缺省定义方式：xmlns＝＜命名空间名＞

在缺省定义方式下，命名空间声明的属性部分仅有保留属性名“xmlns”，属

性值部分和直接定义方式相同，例如：

<联系人列表 xmlns="http://cssci.nju.edu.cn/research/联系人列表.dtd">

如果是用直接方式声明的命名空间，那么在定义该命名空间作用域内的元素名或属性名前，加上带冒号的“<命名空间前缀>”，这些元素名便和“<命名空间名>”联系起来了。在这种声明中，“<命名空间名>”不能为空。

如果使用缺省方式声明的命名空间，那么属性值中的“<命名空间名>”就是该命名空间元素作用域内的缺省命名空间。在这种声明中，“<命名空间名>”可以为空。

1.1.5 CSS、XSL 与 XPath

在 XML 文档中，内容和表现形式是分开的，DTD 定义了元素，但对元素进行的处理不由 DTD 决定，所以当浏览器显示 XML 数据时，各个元素应采用怎样的显示格式，如字体、字号、颜色等，必须事先指定。XML 可用的格式指定语言有两种：CSS（Cascading Style Sheets，层叠样式表）和 XSL（Extensible Stylesheet Language，可扩展样式表语言）。

最初的 HTML 是一种描述文件结构和内容的语言，随着 Internet 的发展，人们对页面显示效果也越来越重视，导致网页设计人员往往忽视 HTML 标签本来的含义，而将精力集中在网页的表现形式上，以至于出现了各种各样依赖于特定浏览器的 HTML 使用方法，用某些工具开发的网页在不同的浏览器上显示的效果大相径庭，这样带来的严重后果就是设计人员经常需要针对不同的浏览器开发特定的网页，虽然这些网页需要表现的内容完全相同，因此大大地增加了工作量，使开发进度拉长并且难以控制。为了解决这种状况，HTML 4.0 中开始提供一种称为样式表的控制方法，使用这种方法后，网页在不同的浏览器上可以得到相同的显示效果。HTML 4.0 虽然没有限定格式表的语言，但 CSS 已经成为事实上的标准。CSS 1.0 是在 1996 年由 W3C 发布的，并且得到了主流浏览器的支持，IE 4.0 和 Netscape 4.0 以上的版本均支持 CSS。

随着 XML 的诞生，人们希望能更多的控制浏览器中内容的表现方式，从而引起了对 CSS 的进一步开发的热潮，以适用 XML 文档。CSS 2.0 在 1998 年 5 月完成，在 CSS 1.0 的基础上增加了 77 种属性，其中相当大的一部分是在现有功能范围内的简单增加，另一部分则定义了一些与输出相关的新特性，如格式布局、语音输出、打印输出等。同时还提供了精确的内容定位，可以下载字型、表格格式以及一些与用户界面有关的设置。

用于 XML 文档格式指定的另一种语言是 XSL，XSL 是在 DSSSL（Document Style Semantics and Specification Language，文档样式语义和规范语言，SGML 文档格式指定的国际标准）和 CSS 的基础上产生的，它的功能远远