基于XML和Web Service的异构数据集成研究与实现

马云峰,王 源
(中国电子科学研究院 公共安全系统部,北京 100041)

摘 要:计算机网络的迅猛发展使企业内部数据交换越来越频繁,然而,系统实现技术及实现时间上的差异造成了在不同的信息系统中存在着大量异构数据。异构数据源的存在给实现不同信息系统间数据互访带来了很大的不便。为了解决异构数据源共享和部署集成平台过于复杂的问题,在充分调研国内外信息集成文献的基础上,基于XML和Web Service技术实现了一个新型的异构数据集成平台。该平台采用XML文件存储元数据,部署时无需新建数据库,实现了轻量级部署;将中介器和包装器发布成Web Service,支持多种集成平台客户端;该平台能够屏蔽网络、操作系统、各种关系型数据库、XML文件的异构性,支持企业集成历史遗留数据、发布信息,并具有高度灵活性、轻便性和可重用性。

关键词:XML;Web Service;XQuery;中介器;包装器
中图分类号:TP302.1 **文献标识码**:A **文章编号**:1673-629X(2010)11-0042-05

Research and Implementation of Heterogeneous Data Integration Based on XML and Web Service

MA Yun-feng, WANG Yuan
(Public Safety Systems Division, China Academy of Electronics and Information Technology, Beijing 100041, China)

Abstract: The rapid development of computer networks makes internal data exchange in corporation more and more frequently, however, the difference of system implementation technology and time make that a lot of heterogeneous data exist in different information systems. The existence of heterogeneous data sources is a big inconvenience for exchange the data between different information systems. In order to share the heterogeneous data sources and simplify the deployment of the integration platform, based on the research on the information integration in domestic and overseas, a novel heterogeneous data integration platform which adopts XML and Web Service was proposed in this paper. The novel platform which uses XML file storage the metadata achieves a lightweight deployment because it is not necessary to build the database; Publishing the mediator and wrapper into Web Service to support multiple integration platform client; The novel platform can shield the heterogeneity of network, operating system, a variety of relational databases and XML documents, support enterprise integrate the history data, release information, and have a high degree of flexibility, portability and reusability.

Key words: XML; Web Service; XQuery; mediator; wrapper

0 引 言

随着信息系统和计算技术的进一步发展,对数据集成有了更高的要求。各组织使用的软件系统越来越多,数据定义、数据存储格式不统一。如何检索异构的数据信息,消除"信息孤岛"[1,2],为信息系统提供统一的数据视图,这是当前研究的一个热点。

目前集成异构数据源的体系结构主要有三种:联邦数据库,数据仓库和中介集成系统[3]。联邦数据库中数据源之间使用数据转换接口来实现数据互访。在集成的数据源种类和个数限定的情况下容易实现,但在数据源种类繁多并且经常变化的情况下,扩展性差;数据仓库集成将几个异构数据源的数据存储到数据仓库中,主要适用于海量数据的统计分析,但是由于集成到一起的数据是原有数据的副本,一方面增加了存储的负担,另一方面不能反映各异构数据源的实时情况;中介集成系统[4]不存储数据副本,只提供一个查询沟通的机制。当用户进行数据查询时,必须到各异构数据源中得到相关的数据,然后利用这些数据构造用户需要的结果,主要应用于对数据响应速度要求不高的

收稿日期:2010-03-08;**修回日期**:2010-06-06
作者简介:马云峰(1983-),男,河北定州人,助理工程师,硕士,研究方向为信息集成;王 源,高级工程师,博士,研究方向为信息集成、SOA、计算机软件等。

情况[5]。

文中在中介集成系统的基础上提出了一种采用 XML[6]和 Web Service[7,8]技术构建数据集成平台的方案。

1　总体设计

1.1　设计目标

在对国内外企业各应用系统运行中存在的问题和将来的发展规划对数据的需求进行深入分析之后,提出数据集成平台的设计目标:数据集成平台不应直接修改或改进现有系统,而应针对现有系统,建立独立的系统对现有系统的数据进行抽取、转换和集成;数据集成平台应可灵活配置各种需集成的数据源;集成框架应支持多种操作系统。

1.2　整体框架设计

在集成框架中,需集成的数据称为局部数据源,分布于各地,它们在存储方式、组织方式上各不相同。为了将这些异构的局部数据源集成起来,从各个局部数据源抽取公共部分建立新的数据模型,并在局部数据源和新的数据模型之间建立映射关系,这个新的数据模型被称之为全局数据源并呈现给用户,用户对全局数据源的操作都可以通过映射关系对应到具体的局部数据源中,从而屏蔽了数据的异构性。

系统体系结构如图 1 所示,从下到上可分为数据层、逻辑层和展示层[9]。

(1)数据层包括各种异构的局部数据源和包装器,其中,局部数据源可以是 oracle、mysql、sqlserver 等关系型数据库,也可以是 XML 文档。与以往中介系统对每个数据源分别建立包装器不同,图 1 的架构中只采用一个包装器来对所有局部数据源进行包装,收到逻辑层传过来的查询指令后,从元数据库中查找映射信息操作局部数据源。

(2)逻辑层包括中介器和元数据库,中介器一方面负责将展示层传来的针对全局数据源的查询转换为对各局部数据源的查询,另一方面将数据层返回的结果整合后发送给展示层;元数据库存放局部数据源、全局数据源及它们之间的关系。中介器以 Web Service 形式发布,便于展示层的调用。

(3)展示层直接面向用户,由于逻辑层的中介器采用 Web Service 发布,所有可以调用 Web Service 的客户端都可以用于展示层,因此,展示层可以采用 JSP,也可以采用 Java 编写的窗口客户端程序,它能够向上为用户提供友好的操作界面,接收用户发出的 XQuery[10,11]查询指令,将 XQuery 指令发送到下面的逻辑层,并接收逻辑层发来的查询结果。

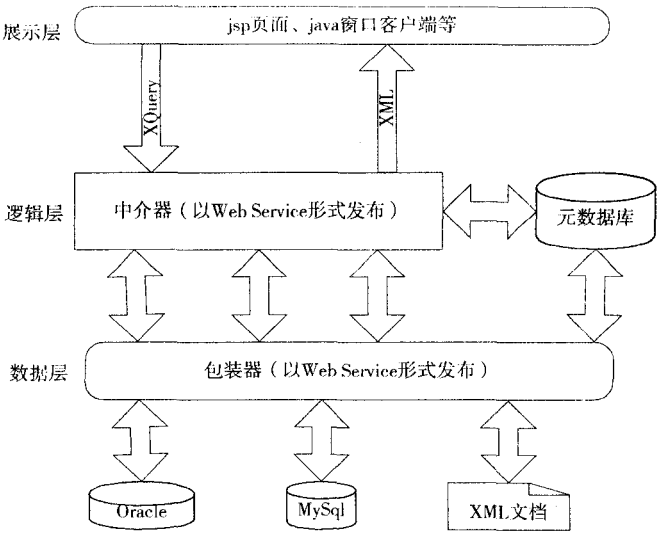

图 1　系统架构图

2　分模块设计

2.1　元数据库设计

元数据库位于逻辑层,用于存储全局数据源、局部数据源及它们之间的映射。为了实现轻量配置,采用 XML 文件充当存储介质。

元数据库向下层的包装器提供局部数据源的地址信息、数据源类型、用户信息等实现一个包装器对多个局部数据源的包装,向同层的中介器提供全局数据源和局部数据源的映射信息实现全局查询向局部查询的转换。

2.2　中介器设计

从图 1 中可以看出,中介器是集成系统的核心模块,根据功能,将中介器划分为两个部分:元数据处理器和查询处理器,中介器的结构如图 2 所示。

元数据处理器负责操作元数据库中的数据,包括全局数据源管理、局部数据源管理和映射关系管理。查询处理器包括两方面的工作,一方面向上接收展示层发送的全局查询语句,将其分解为针对局部数据源的查询语句发送到数据层;另一方面向下接收数据层的查询结果,对其进行全局数据源封装之后,发送至展示层。

中介器工作流程如下:

1)展示层根据系统提供的全局数据源,编写并发送 XQuery 查询语句;

2)查询处理器中的查询解析部件收到 XQuery 后,验证其语法,提取相应的关键字,判断该语句是否合法,如果合法,将 XQuery 传到查询分解器,转第三步,否则返回第一步;

3)查询分解器通过调用元数据处理器查询元数据

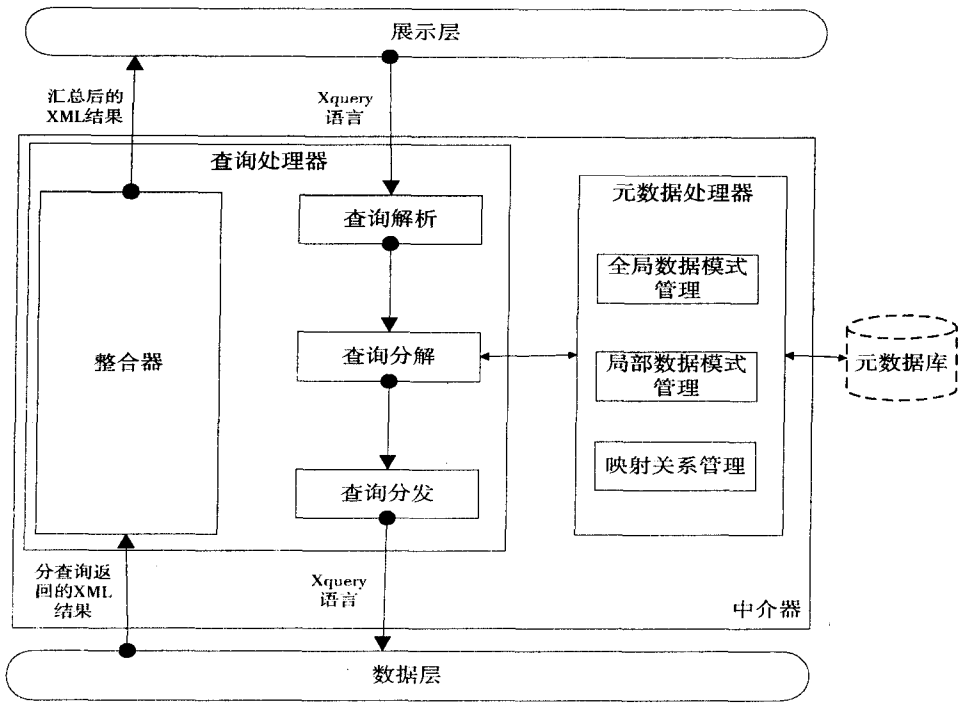

图 2 中介器结构图

库中的全局数据源与局部数据源的映射关系，将针对全局的 XQuery 分解为针对局部的 XQuery；

4）查询分发器将针对局部的 XQuery 发送到数据层的包装器；

5）数据层的包装器从元数据库中获取局部数据源信息，执行局部查询，以 XML 形式返回查询结果，整合器收到查询结果后，根据全局数据源与局部数据源的映射关系，按照全局数据源样式将其封装，返回涵盖多个局部数据源的集成数据。

2.3 包装器设计

包装器是中介器与局部数据源联系的桥梁，用来封装局部数据源，为中介器提供统一的查询接口，按照功能可以分为查询语言转换器、查询执行引擎、JDBC Driver 和结果转换器四部分，如图 3 所示。

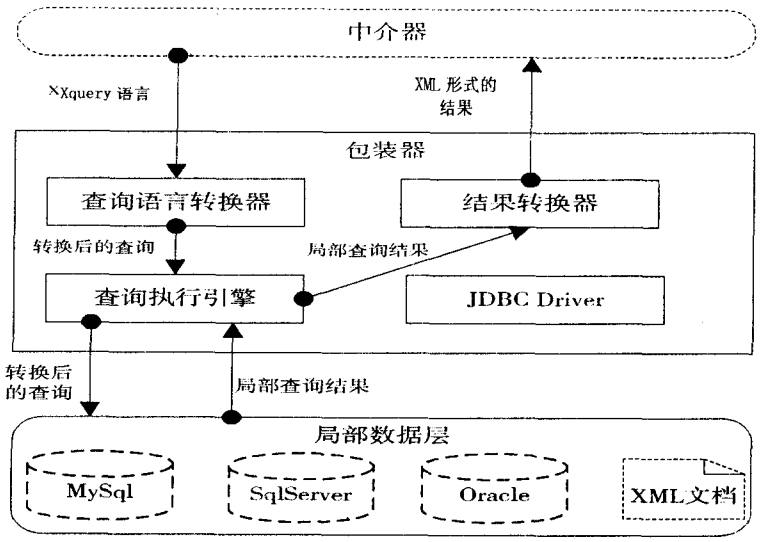

图 3 包装器结构图

图 3 中的包装器收到中介器分解后的局部 XQuery 查询语言后，通过查询语言转换器[12]，将 XQuery 语句翻译成本包装器对应数据源的查询语言，如对于 mysql 等关系数据库，将 XQuery 转换为 SQL 语句，对于 XML 文档，则无需进行转换，直接用 XQuery 进行查询。转换后的查询语言被发送到查询执行引擎。

查询执行引擎根据不同的数据源，选择不同的 JDBC驱动程序，与对应的局部数据源建立连接、执行查询语言转换器转换后的查询语言，并将查询结果发送到结果转换器。

结果转换器将查询结果按照一定的规则包装成统一的 XML 格式，返回给中介器。

3 系统实现

为了验证上述设计方案的可行性，我们编程实现了一个原型系统。系统采用 XML 文件存储元数据，中介器和包装器被发布成 Axis 实现的 Web Service，Web 服务器是 Tomcat6.0，应用层使用 JSP 和用户交互并调用 Web Service。除了使用 JSP 以外，还可以编写窗口程序访问发布的 Web Service，体现了设计方案的可扩展性。

3.1 元数据库

采用 XML－SCHEMA 定义元数据存储结构，采用 XML 文件作为元数据的存储介质。

3.1.1 全局数据源

图 4 中所示的全局数据源 GlobalSources 包含全局数据源元素 GlobalSource，其属性 name 用来唯一标识全局数据源元素。Attributes 为 GlobalSource 的属性集合，Attribute 为 GlobalSource 的具体属性

3.1.2 局部数据源

图 5 中，局部数据源 LocalSources 包含子节点局部

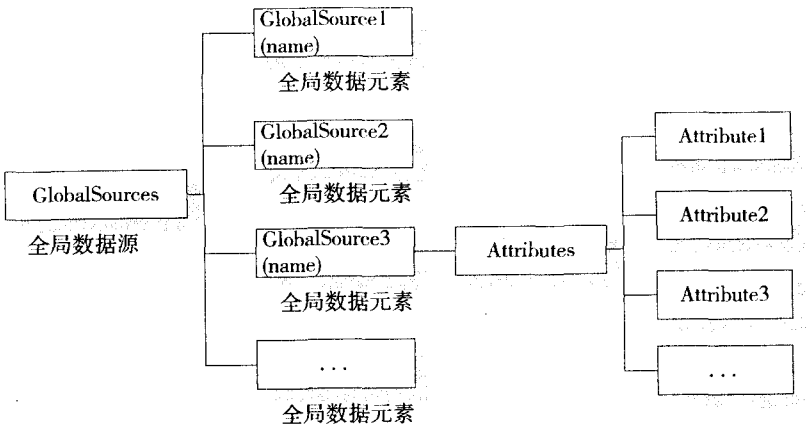

图 4　全局数据源定义

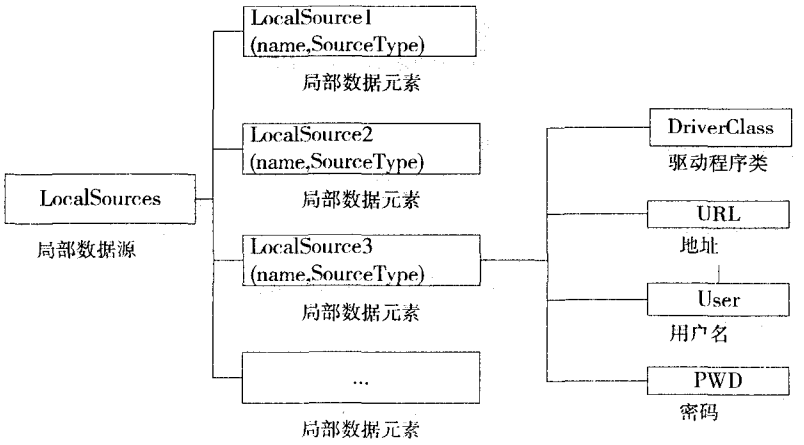

图 5　局部数据源定义

数据源元素 LocalSource。

LocalSource 属性包括 name(用来唯一标识该元素)、SourceType(标明该元素的类型,RDB 代表为关系数据库,XMLDOC 为 xml 文档)。

LocalSource 子节点包括 DriverClass(标识操作该实例的 jdbc 驱动程序)、URL(标识该实例的地址)、User 和 PWD(提供了该实例的访问用户名和密码)。

3.1.3　映射关系

如图 6 所示,GlobalSource 代表全局数据元素,它包含 Origins(记录该全局数据元素的不同来源)和 Attributes(记录该全局数据元素的属性)两个子节点。Origin 为具体的来源,包括 LocalName(与该全局数据元素对应的局部数据元素的名称)和 Entity(与该全局数据元素对应的局部元素中具体的实体对象)两个子节点。Attribute 为该全局元素具体的属性,包括 LocalAttrs 子节点,LocalAttrs 记录全局数据元素属性对应的局部数据元素的属性 LocalAttr,通过以上的存储结构,在全局数据元素与局部数据元素之间建立了映射关系。

3.2　中介器

3.2.1　全局数据源管理

建立全局数据源管理 Web Service,封装对全局数据模式 XML 文件的操作。在 Web Service 中,通过 dom4j 对 XML 文档(GlobalSource. xml)进行编辑,包括全局数据模式的添加:AddSource(String name,String attr[ ]),其中,name 为全局数据模式的名称,相当于关系数据库中的表名,attr 为该全局数据模式的属性,相当于关系数据库中的表的列名;删除:DelSource(String name),其中,name 为全局数据模式的唯一标识,修改:ModifySource(String name,String gsname,String[ ]gsattr),其中,name 为待修改的全局数据模式的名字,gsname 为新名,gsattr 为对应的属性。

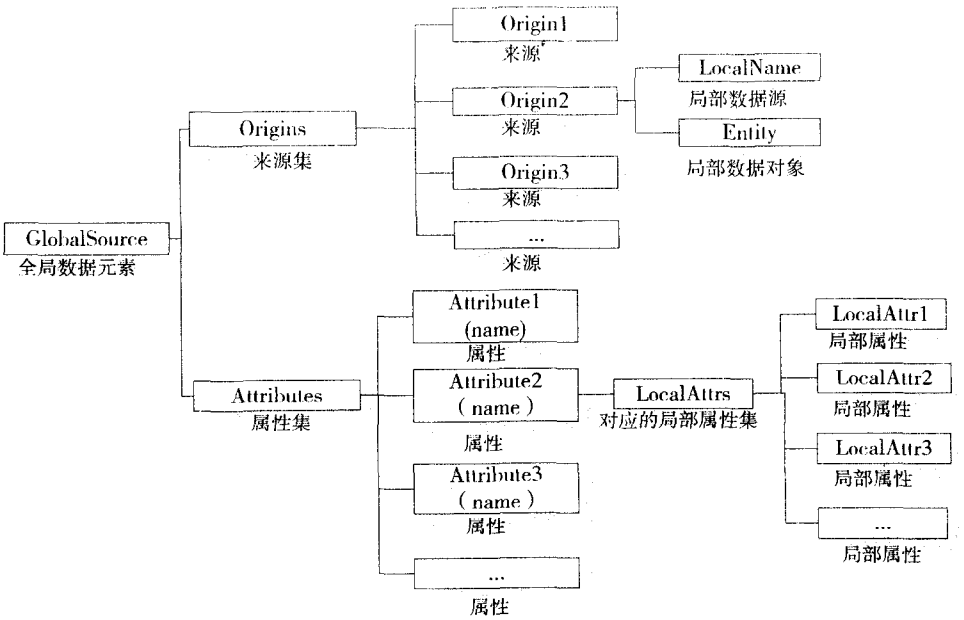

图 6　全局数据源与局部数据源映射关系模式定义

3.2.2　局部数据源管理

建立局部数据源管理 Web Service,封装对局部数据模式 XML 文件的操作。在 Web Service 中,通过

dom4j 对 XML 文档(LocalSource. xml)进行编辑,包括局部数据模式的添加:AddSource();删除:DelSource(),修改:ModifySource()。

3.2.3 映射关系管理

建立映射关系管理 Web Service,封装对映射关系 XML 文件的操作。在 Web Service 中,通过 dom4j 对 XML 文档(Mapping. xml)进行编辑,包括映射关系的添加:AddMapping();删除:DelMapping();修改:ModifyMapping()。

3.2.4 查询管理器

1)查询解析。

分离出其中的关键字,判断是否符合词法、语法要求。

2)查询分解。

将通过解析的查询语句根据元数据库将对全局数据模式的请求转换为对局部数据模式的查询。

3)查询优化。

将分解后的针对各个局部数据源的查询语句进行必要的优化处理。

4)查询调度。

将优化后的查询语句发送到包装器,包装器对每个数据源执行查询。

5)结果汇总。

查询管理器收到各个局部数据源执行查询语句后返回的结果后,首先根据元数据库,将各个分结果转换为针对全局数据模式的结果,然后再将全局化的结果进行合并。

3.3 包装器

3.3.1 查询语言转换器

局部数据源涉及关系数据库和 XML 文件,查询关系数据库使用 SQL 语言,查询 XML 则采用 XQuery 语言。元数据库中对局部数据源的类型进行了记录,包装器根据局部数据源的名称,从元数据库中查询该局部数据源的类型,如果是关系数据库,则进行 XQuery 语言到 SQL 语言的转换,如果是 XML 文档,则无需转换。下面介绍从 XQuery 语言到 SQL 语言的转换。

FLOWR 表达式中的 for 语句转变成 SQL 中的 from 语句,where 语句重写成 SQL 中的 where 语句,order 重写成 SQL 中的 order 语句,return 语句重写成 SQL 中的 select 语句。

3.3.2 结果转换器

局部数据源执行查询语句后,返回查询结果,需要对查询结果进行转换,经过综合考虑,我们决定,所有的查询结果都转换成 XML 的形式,如下所示:

```
<? xml version="1.0" encoding="GBK"? >
<results>
<row><列名>列值</列名>…</row>…
</results>
```

XML 文档执行 XQuery 返回的结果,已经是符合要求的形式。因此,针对关系数据库执行 SQL 查询后返回的 ResultSet 对象的转换。Java 中的 ResultSetMetaData 对象,可用于获取关于 ResultSet 对象中列的类型和属性信息的对象。通过 ResultSetMetaData 对象,获取返回结果的列名,通过 ResultSet 对象,获取返回结果的值,利用 dom4j 建立 Document 对象,在 Document 基础上添加 Element 对象,最后,通过 Document 的 asXML()方法将 Document 对象转换为字符串,完成查询结果的形式统一化。

3.4 应用层

JSP一方面与用户交互,另一方面访问发布在 tomcat 中的以 Web Service 形式存在的中介器和包装器,整体框架是一个 B/S 架构,方便配置管理和软件维护。应用层的表现形式可以多样化,只要可以访问 Web Service,采用 C/S 架构也可以达到目的。B/S、C/S 架构的可选择性,充分体现了该方案的灵活性。

4 系统特色

按照上文设计的系统,具有如下的特色:

1)采用 XQuery 作为全局数据查询语言,实现了基于标准接口的统一查询;

2)中介器、包装器发布成 Web Service,可以充分发挥 SOA 的优势,可供多种客户端调用,增强系统扩展性;

3)采用 B/S 架构,利用 JSP 页面实现综合查询和元数据库的管理,用户或者管理员只需一个可以上网的浏览器即可实现查询或管理;

4)一改以往针对每个局部数据源分别建立包装器的局面,只采用一个包装器根据需要动态访问局部数据源,设计更加清晰。

5 结束语

主要研究了异构数据集成的相关理论和技术,对现有的数据集成方法和集成系统的体系结构进行了分析比较。提出了一个基于 XML 技术和 Web Service 技术的异构数据集成平台建设方案。

采用 XML 模式作为数据交换的公共模型,建立全局数据源和局部数据源的映射关系,屏蔽了底层数据源格式的差异;采用 XQuery 作为统一的查询语言,

(下转第 50 页)

曲线 1 对应的分类方法的分类效果最好。

5 结束语

利用参数法或非参数法建立 ROC 曲线，使用几率点欧氏距离、曲线下面积、最佳阈值点等指标对两类算法进行性能评估，克服了传统算法评估结果依赖阈值的局限性。通过数值模拟实验证明，利用 ROC 曲线进行两类分类算法的评估是高效可行的。目前，ROC 曲线主要应用于两类分类方法的评估问题，把 ROC 曲线应用于多目标分类算法的评估问题，是下一步的研究工作。

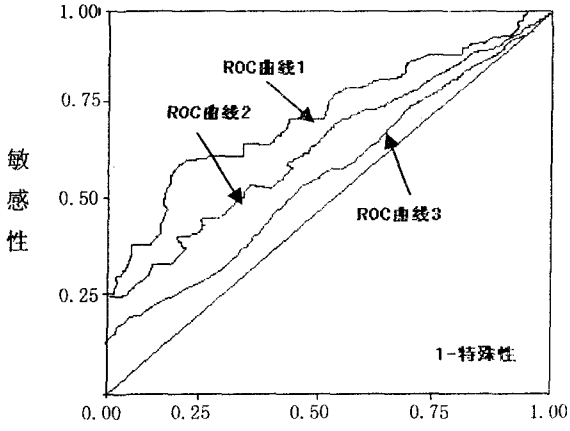

图 3 三条 ROC 曲线图

参考文献：

[1] Alsing S G. The Evaluation of Competing Classification[D]. US:Air Force Institute of Technology,2002.

[2] Provost F, Fawcett T. Robust Classification for Imprecise Environments[J]. Machine Learning, 2001, 42(3): 203 - 231.

[3] 宇传华. ROC 分析方法及其在医学中的应用[D]. 西安：第四军医大学,2000.

[4] Swets J A. ROC Analysis Applied to the Evaluation of Medical Imaging Techniques[J]. Investigative Radiology, 1997, 14(2):109 - 121.

[5] 田 俊. 两个诊断指标的 ROC 曲线下面积的非参数检验方法[J]. 数理医学杂志,2002,15(3):201 - 204.

[6] 邹莉玲，沈其君，陈 峰，等. ROC 曲线下面积的 ML 估计与假设检验[J]. 中国公共卫生,2003,19(1):127 - 128.

[7] 王昌元，谢晋东，李月卿. ROC 曲线中 AZ 的物理意义及数学表达式[J]. 泰山医学院学报,2003,24(2):102 - 105.

[8] 邹洪侠，秦 锋，程泽凯. 二类分类器的 ROC 曲线生成算法[J]. 计算机技术与发展,2009,19(6):115 - 118.

[9] de sa J P M. 模式识别—原理、方法及应用[M]. 北京：清华大学出版社,2002.

[10] 孙长亮，何 峻，肖怀铁. 基于 ROC 曲线的目标识别性能评估方法[J]. 雷达科学与技术,2007,5(1):22 - 25.

[11] Marzban C. A Comment on the ROC Curve and the Area Under it as Performance Measures[EB/OL]. 2004. http://www.nhn.ou.edu/marzban.

[12] Hanley J A, McNeil B J. The Meaning and Use of the Area Under a Receiver Operating Characteristic(ROC) Curve[J]. Radiology,1982,143(1):29 - 36.

(上接第 46 页)

使基于统一接口的查询成为可能；开发语言基于 Java，易于实现程序的移植；用 Web Service 技术发布包装器和中介器，提高了软件的复用性；采用 B/S 架构，可以对全局数据源、局部数据源及它们之间的映射关系实现方便管理；对所有的局部数据源采用一个包装器进行封装，使增加新的数据源更加方便，增强了系统的扩展性。

参考文献：

[1] 史万江，吴 健. 一种异构数据集成框架及关键技术研究[J]. 微处理机,2008(4):167 - 173.

[2] El - Khatib H T, Williams M L, MacKinnon L M. Using a distributed approach to retrieve and integrate information from heterogeneous distributed databases[J]. Computer Journal, 2002, 45(4): 381 - 394.

[3] 方长江. 异构数据源集成技术在军事中的应用研究[D]. 济南：山东大学,2007.

[4] 王 燕，陈庆奎. 异构数据库集成中间件的研究与实现[J]. 计算机工程与设计,2008,29:5738 - 5744.

[5] May W. An integrated architecture for exploring, wrapping, mediating and restructuring information from the web[C]// In Australasian Database Conference (ADC 2000). Australia: Austr. Comput. Sci. Commun,2000:82 - 89.

[6] 李军怀，周明全，耿国华，等. XML 在异构数据集成中的应用研究[J]. 计算机应用,2002,22(9):18 - 24.

[7] 孙长俊，周晓峰. 基于 Web Services 的企业应用集成模型[J]. 计算机技术与发展,2006,16(5):209 - 210.

[8] 佘名高，贾秀峰，林坤江，等. 基于 Web 服务的企业应用集成[J]. 计算机技术与发展,2007,17(5):55 - 58.

[9] 熊玉庆，唐新怀. XCouple：一个新型异构信息集成平台[J]. 微电子学与计算机,2007,24:171 - 173.

[10] W3C Recommendation, XQuery 1.0: an XML query language[EB/OL]. 2007. http://www.w3.org/TR/xquery/.

[11] 李 烨，冯志勇. 基于 XQuery 的数据集成研究[J]. 微处理机,2008(3):120 - 122.

[12] 尚 蕾，孙志辉. 基于 XML 的异构数据集成系统的查询处理[J]. 计算机工程,2005(3):121 - 123.