第 32 卷 第 21 期 计 算 机 工 程 2006 年 11 月
Vol.32 No.21 Computer Engineering November 2006
·软件技术与数据库· 文章编号：1000—3428(2006)21—0088—03 文献标识码：A 中图分类号：TP311

基于 Web 的数据挖掘算法与数据仓库的接口设计

刘新颖，王丽亚
(上海交通大学工业工程与管理系，上海 200030)

摘 要：提出了一种基于 Web 的数据挖掘系统中数据挖掘算法与数据仓库的接口设计方案，解决了算法与数据仓库的接口问题，实现用户通过 Web 浏览器动态调用算法，算法在 Web 环境下对数据仓库数据进行挖掘，发现有用的知识。该接口方案的通用性增强了数据挖掘系统的扩展性，有利于系统快速添加更多的新算法，以满足各种挖掘需求。

关键词：数据挖掘；接口；算法；数据仓库

Interface Design for Web-based Data Mining Algorithm and Data Warehouse

LIU Xinying, WANG Liya
(Department of IEM, Shanghai Jiaotong University, Shanghai 200030)

【Abstract】 Web-based data mining on data warehouse is becoming increasingly important. The interface between data mining algorithms and data warehouse is a critical issue. An interface approach is presented. By applying the interface, a Web-based data mining system is developed. Browsers of the client to find knowledge can call the algorithms in the system dynamically. And also the new algorithms can be conveniently added in the Web-based data mining system.

【Key words】 Data mining; Interface; Algorithm; Data warehouse

基于 Web 的数据挖掘系统通过 Internet 技术与数据挖掘技术相结合，借助浏览器环境对企业存放在数据仓库中的数据进行有效的分析，实现跨平台挖掘知识，有利于企业做出及时的决策。由于数据挖掘系统中的算法要在 Web 环境下对企业数据仓库进行挖掘并输出结果，因此需解决基于 Web 的数据挖掘算法和数据库管理系统的接口问题。本文提出了一种接口方案，并在此基础上开发了数据挖掘系统的原型软件。

1 基于 Web 的数据挖掘系统的构架

基于 Web 的数据挖掘系统采用 B/S 构架，利用面向数据仓库的知识发现技术从数据仓库中提取隐含的、未知的和潜在有用的能被人们理解的规则和模式，并建立企业的知识库，提高决策系统的智能性。系统选用 Visual Studio .NET 2003 作为开发平台，C#作为系统集成开发语言，数据仓库、数据挖掘库和知识库建立在 MS SQL Server 上。

本系统由数据访问层、业务逻辑层、用户界面层组成，如图 1 所示。

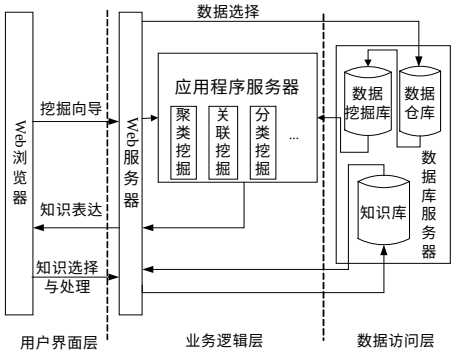

图 1 基于 Web 的数据挖掘系统的构架

数据访问层包括数据仓库、数据挖掘库和知识库。其中，数据仓库是一个面向主题的、集成的、非易失性的且随时间变化的数据集合，数据源来自于企业各种信息系统的数据库、事务文档库以及企业与企业，企业与客户之间所进行的电子商务活动记录等[1]。业务逻辑层负责处理客户端的请求及返回结果：Web服务器把收到的浏览器的http请求发送给应用程序服务器和数据库服务器；应用程序服务器是一个开放的数据挖掘算法库，可通过接口与数据仓库连接，实现挖掘功能，最终输出结果。用户界面层采用网络浏览器作为用户与系统之间的桥梁。

业务逻辑层要对数据仓库进行有效的知识发现和数据挖掘，必须解决算法和数据库管理系统的集成问题[2]。为了各挖掘算法能够在.NET平台下访问数据仓库，需要对基于Web的挖掘算法与数据仓库之间的接口进行设计。

2 基于 Web 的数据挖掘算法与数据仓库的接口设计

所谓基于 Web 的数据挖掘算法是指基于 Web 模式运行的挖掘算法。当挖掘算法集成到 B/S 结构的数据挖掘系统后，成为基于 Web 的数据挖掘算法。客户端利用浏览器实现对数据仓库的跨平台操作，在 Web 环境下获取数据仓库的数据，运行后将结果输出到 Web 页面。

数据挖掘算法可以由C/C++、Java、Delphi等多种数值计

基金项目：国家自然科学基金资助项目“基于数据挖掘的大规模定制的产品多样化决策”(70471022)；NSFC/RGD 项目“大规模定制的生产组织与管理的理论、方法与关键技术”(70418013)
作者简介：刘新颖(1978－)，女，硕士生，主研方向：数据挖掘，决策支持；王丽亚，教授
收稿日期：2005-12-05 **E-mail：**liuxinying@sjtu.edu.cn

算语言编写，各算法可以独立地实现挖掘功能。系统中的算法要在C#开发的.NET平台下运行，但有些算法在C#中不是托管代码，不能被C#直接调用[3]。C＃可以通过互操作实现对其他语言编写的组件的调用。为此，将系统的算法编译封装成独立的组件，多个算法组件组成算法库，根据用户的挖掘任务从算法库中选择相应的算法组件。C#通过接口调用算法，可以在挖掘系统中动态载入算法，提高系统算法的可扩展性。

不同的算法，可以编译生成不同的组件，算法组件可以是一个进程内组件，即一个独立封装的动态链接库文件，带.dll后缀，C#调用算法DLL组件时使用命名空间“System.Runtime.InteropServices”中的“DllImport”特征类来实现互操作；组件也可以是一个进程外组件，即一个可执行性文件，带.exe后缀，C#调用算法EXE组件时使用“System.Diagnostics.Process”类执行互操作，当组件被调用后，以独立进程的状态运行。由于进程内组件与调用它的程序在同一个进程空间，因此性能比进程外组件要佳[4]。无论哪种形式的算法组件，都可以通过设计的接口被C#开发的系统调用，能够实现代码重用性，独立于系统而存在，便于维护、升级和扩展。

一般来说，算法的输入数据有两种来源：可以通过ODBC/ADO获取数据库的数据进行操作，也可以对文件中的数据进行运算。与之相对应，算法也有两种输出方式。而在基于Web的数据挖掘系统中，用于挖掘的数据来自数据仓库，算法要通过与数据仓库的接口技术来实现挖掘功能。为此，设计两种模式的算法应用程序与数据仓库的编程接口，分别为模式1(见图2)和模式2(见图3)。其中，模式1的算法直接对数据挖掘库进行操作，模式2的算法通过Web服务器的文件接口来间接获得数据挖掘库的数据。

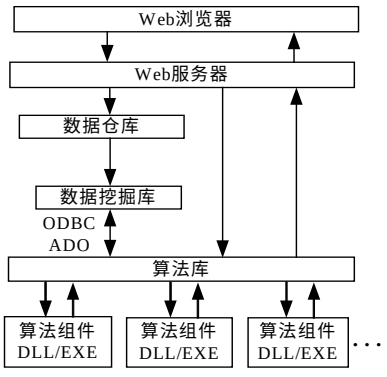

图2 模式1接口的架构

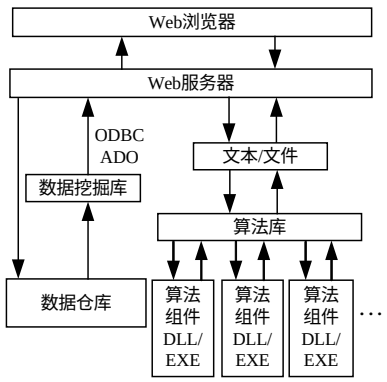

图3 模式2接口的架构

(1)模式1的接口设计

算法库的输入接口有两个：与Web服务器的输入接口和数据挖掘库接口。Web服务器通过接口向算法库传递挖掘指令，调用指定挖掘算法组件；算法库与数据挖掘库的接口用来向算法传递输入数据，算法通过ODBC/ADO连接数据挖掘库，用SQL语句操作数据库。而数据挖掘库是根据浏览器端的用户所选字段来筛选数据仓库里的数据组成的。

算法库的输出接口：算法库接收到输入数据后，运行并将计算结果返回到Web服务器，继而传递到用户的Web浏览器，根据用户的选择决定是否存储到知识库。

模式1接口的优点：

1)算法直接操作数据库，快捷实时地获取数据，不需要系统缓冲过程，从而减少了交互，加快了响应速度，可以较小的开销实现连续的执行，从而改善性能。

2)接口形式简单，数据更新等逻辑由数据库直接控制。

3)Web服务器的负荷较小。通过包含算法库的应用程序服务器连接数据库，增加了系统的稳定性。

模式1接口的缺点：

1)对内存要求高。数据量太多时，内存响应速度降低。

2)对算法设计要求高，数据格式要依赖数据库。同时如果已有的成熟算法不直接操作数据库，需重新设计算法，改动较大。

(2)模式2的接口设计

算法库的输入接口为Web服务器上的文本或其它格式的文件。Web服务器与数据仓库连接，根据用户的选择调用数据仓库的数据组成数据挖掘库，反馈到Web服务器。Web服务器通过与算法库之间的文本/文件接口向算法库传递挖掘指令、输入数据流和输入参数，调用指定挖掘算法的DLL文件或EXE文件。

算法库的输出接口也是Web服务器上的文本或文件。算法运行后，将计算结果输出到文本/文件中，Web服务器访问文件接口，将文本/文件里的数据传递到用户的Web浏览器，根据用户的选择决定是否存储到知识库。

模式2接口的优点：

1)算法提取数据速度快。无论数据库的速度如何，从Web服务器的硬盘读取数据通常比从数据库中检索数据要快。因此，将输入、输出数据缓存在Web服务器磁盘的文件中，可以提高性能。

2)当数据太多时，无法全部缓存在内存，将数据以文本或其它格式的文件缓存在Web服务器的硬盘上，便于操作。

3)输入比较简单。文件的内容是由Web服务器根据用户的选择动态生成的，算法不必考虑输入数据的筛选问题，可以直接操作。

模式2接口的缺点：

1)增加了对Web服务器内存的操作次数，需要多次通过内存读取数据。

2)各算法需要单独开发自己的输入与输出的文本/文件的格式，以适用于算法的输入输出需要。

3 系统算法与数据仓库接口的实现技术

3.1 模式1接口的实现技术

采用模式1接口的算法可以直接操作数据库。算法如果以动态链接库的形式集成进系统，算法的DLL文件中需要定义调用接口的入口函数及算法的运算函数，并包含数据库操作语句；算法还可以以可执行文件的形式集成，也要包含操作数据库的语句。算法操作数据库的语句步骤如下：

(1)分配环境句柄；

(2)分配连接句柄；

(3)连接数据源；

(4)分配语句句柄；

(5)对数据库进行操作，选择数据；

(6)断开连接；

(7)释放 ODBC 环境。

C#调用算法 DLL 组件时使用命名空间“System.Runtime.InteropServices”中的“DllImport”特征类；C#调用算法 EXE 组件时使用“System.Diagnostics.Process”类。

3.2 模式 2 接口的实现技术

采用模式 2 接口的算法不能直接操作数据库，通过 Web 服务器的文件接口获取输入数据及并输出结果到文件。算法如果以动态链接库的形式集成进系统，则算法的 DLL 文件中需要定义输入、输出文件的名称及 Web 服务器上的物理地址；算法还可以以可执行文件的形式集成，也要包括输入、输出文件的名称及 Web 服务器上的物理地址。

在.NET 平台，Web 服务器与文件接口相连，并对数据仓库进行以下操作：

(1)引入 ADO 库定义文件；

(2)用 SQLConnection 对象连接数据仓库；

(3)利用建立好的连接。通过 Command 对象执行 SQL 命令，通过文件流语句将挖掘数据导入到算法的输入文件；根据用户的选择决定是否将输出结果从文件中导入到数据库。

(4)关闭连接，释放对象。C#调用算法 DLL 组件时使用命名空间“System.Runtime.InteropServices”中的“DllImport”特征类；C#调用算法 EXE 组件时使用“System.Diagnostics.Process”类。

4 系统应用接口模式集成的数据挖掘算法

系统两种模式的接口方案对各算法是通用的，增强了数据挖掘系统的扩展性，有利于系统添加更多的新算法，以满足各种挖掘需求。目前系统应用接口模式已经集成了聚类挖掘算法、关联挖掘算法和关联挖掘算法。

4.1 聚类挖掘算法

聚类(Clustering)就是将数据分组成为多个类(Cluster)。在同一个类内对象之间具有较高的相似度，不同类之间的对象差别较大。本系统使用基于距离的 K-means 算法和改进的 K 均值算法，对客户信息库、零件信息库、成本信息库、库存信息库等进行聚类。

4.2 关联规则挖掘算法

关联规则是形式如 X⇨Y 的一种蕴涵或规则，其中 X 和 Y 分别是两个项目集，且 X∩Y=ϕ。关联规则挖掘能够发现大量数据中各项目集之间的关联或相关联系。本系统中用于关联规则挖掘的算法为 Apriori 算法，进行多维关联规则挖掘。系统用于关联挖掘的事务集包括客户事务集和销售事务集等，数据模型采用雪花结构。

4.3 分类挖掘算法

分类的目的是通过分析在训练集中的数据表现出来的特性，为每一个类找到一种准确的描述和模型，形成分类规则。系统采用机器学习领域的决策树算法中的 CART(Classification And Regression Trees)算法，对客户信息库、客户偏好库、产品信息库等进行分类规则的挖掘。

5 算法的接口实现示例

为了更清楚地说明系统算法与数据仓库接口的实现技术，以系统 VC++编写的 K-means 算法为例，该算法可以直接操作数据仓库，因而采用模式 1 接口。算法生成了动态链接库，并定义接口如下：

```
#include “sqlext.h”
//定义 dll 文件的加载接口，从数据库取数据并预处理
extern "C" _declspec(dllexport) int LoadPatterns(int NumClust){
RETCODE retcode;
retcode=SQLAllocHandle(SQL_HANDLE_ENV,NULL,&henv);
//分配环境句柄
retcode=SQLAllocHandle(SQL_HANDLE_DBC,henv,&hdbc);
//分配连接句柄
retcode=SQLConnect(hdbc,(SQLCHAR)"KmeansDb",SQL_NTS,
(unsigned char *)
"sa",SQL_NTS,(unsigned char *)"1",SQL_NTS); //连接数据源
retcode=SQLAllocHandle(SQL_HANDLE_STMT,hdbc,&hstmt);
//分配语句句柄
retcode=SQLExecDirect(hstmt,(unsigned char*)"SELECT*
FROM KmeansTable",SQL_NTS); //抽取数据
InitClusters();
DistributeSamples();
CalcNewClustCenters();
ShowClusters();
SQLFreeHandle(SQL_HANDLE_STMT, hstmt);
SQLDisconnect(hdbc); //断开连接
SQLFreeHandle(SQL_HANDLE_DBC, hdbc);
SQLFreeHandle(SQL_HANDLE_ENV, henv); //释放 ODBC 环境
}
```

其中 InitClusters()用于标准化处理数据，DistributeSamples()用于分配样本，CalcNewClustCenters()用于聚类计算，ShowClusters()将结果输出到数据库。

在.NET 平台定义算法类，调用算法。

```
//算法类，定义 dll 文件的导入
public class dll
{ [DllImport("kmeantxt.dll",EntryPoint= "LoadPatterns")]
public static extern int LoadPatterns();
}
```

当需要调用算法的 DLL 文件时，执行语句：

```
dll.LoadPatterns();
```

6 结论

本文提出了一种基于 Web 的数据挖掘系统中用于连接算法与数据仓库的接口技术，解决了算法与数据仓库的集成问题。用户可以动态地调用算法库中的算法。算法通过接口技术在 Web 环境下对数据仓库进行挖掘，发现有用的知识，从而提高决策的智能性。通过应用这种接口方案，系统可以添加更多的新算法，保证系统的扩展性，增强挖掘功能。

参考文献

1 刘　浪, 王丽亚, 黄海量. 基于 Web 的数据仓库解决方案[J]. 计算机工程, 2005, 31(1): 92.

2 Sarawagi S, Thomas S, Agrawal R. Integrating Association Rule Mining with Relational Database Systems: Alternatives and Implications[J]. Data Mining and Knowledge Discovery, 2000, 4(2/3): 89.

3 Tapadiya P. Net Programming: A Practical Guide Using C#[M]. Pearson Education, 2002.

4 葛明铭, 傅育熙. 用进程代数描述 COM 接口调用[J]. 计算机工程, 2003, 29(13): 82.