

Matlab 环境下的 Oracle 数据库访问技术

陈 静,范乃吉,袁晓东,蒋一岚*

(中国工程物理研究院 激光聚变研究中心,四川 绵阳 621900)

(* 通信作者电子邮箱 jiangyilan1023@163.com)

摘 要:在 Matlab 开发环境下,采用 ODBC、JDBC 和 ADO 三种数据库访问方式,实现了与 Oracle 数据库的交互功能。给出了访问过程中典型问题的处理方法,解决了 MATLAB Database Toolbox 无法对 LOB 型大对象数据进行访问与操作的难题。最后,比较分析了三种访问方式的特点与使用范围,得出结论:ODBC 使用简单但访问速度慢且可移植性差;JDBC 继承于 ODBC,使用与 Matlab 一致的 Java 语言风格,支持跨平台作业,可移植性好,且与 Matlab 的兼容性最好;而 ADO 比 ODBC、JDBC 访问速度快,扩展性好,能处理 LOB 型大对象数据,但 Matlab Database Toolbox 对 ADO 不提供函数支持。

关键词:数据库访问;Oracle;ODBC;JDBC;ADO

中图分类号: TP302.7 **文献标志码:** A

Oracle database access technology based on Matlab environment

CHEN Jing, FAN Naiji, YUAN Xiaodong, JIANG Yilan*

(Laser Fusion Research Center, China Academy of Engineering Physics, Mianyang Sichuan 621900, China)

Abstract: Three database access technology including ODBC, JDBC and ADO are applied to communicate with Oracle database in Matlab development environment. Solutions of typical connecting problems were presented and an approach to access Large Object Binary (LOB) data was proposed. In the end, the characteristics and scope of the three database access methods in Matlab were compared and analyzed to conclude that ODBC is simple to use but has low access speed; JDBC inherits from ODBC, takes full advantage of the Java language style as Matlab with portability, safety and compatibility; ADO has higher access speed, better scalability performance than ODBC and JDBC, and can process data of LOB type, but Matlab Database Toolbox provides no functions to support such application.

Key words: database access; Oracle; Open Database Connectivity (ODBC); Java Database Connectivity (JDBC); ActiveX Data Object (ADO)

0 引言

随着计算机技术的迅猛发展,数据库技术的应用也越来越广泛,而数据库技术应用中最基本、最重要的一个方面就是对数据库的访问。数据库访问是一种将数据库外部与应用程序通信的过程抽象化,通过提供访问接口,简化客户端访问数据库过程的技术。

Matlab 支持多种通用数据库访问技术,包括: ODBC、JDBC、DAO、RDO、OELDB、ADO 等。面对各类复杂的数据库访问技术,在 Matlab 开发环境下采用何种技术来连接访问 Oracle 数据库一个是值得研究的问题。

Matlab 平台下有一个为关系型数据库和 Matlab 进行数据交互提供接口函数支持的工具包 Database Toolbox,它作为 Matlab JVM 的一部分随 Matlab 同步安装。同 Matlab 一样, Database Toolbox 也是基于 Java 语言编写的。在 WINDOWS 环境下, Database Toolbox 支持 JDBC 驱动和 ODBC 驱动(实际上是通过使用 JDBC/ODBC 桥来连接数据库的 ODBC 驱动);在 UNIX 环境下,它只支持 JDBC 驱动^[1]。

根据 Database Toolbox 提供的数据库接口函数,Matlab 集成了一个与数据库访问交互的 GUI 界面可视查询生成器 (Visual Query Builder, VQB),VQB 提供了基本的查询和插入功能,用户不需要掌握结构化查询语言 (Structure Query Language, SQL)即可与数据库进行简单的通信访问。

Database Toolbox 使用简单,功能强大,通过其提供的接口函数极大简化了 Matlab 与关系型数据库间的数据交互,而且它支持同时与多个数据库的并行任务和大量数据集的分段输入。然而 Database Toolbox 并不支持非结构化大型对象 (Large Object, LOB) 数据的存储,为此,本文提出了在 Matlab 环境下以 ADO 数据库访问方式来存储管理 LOB 的方法。在 Matlab 环境下,采用 JDBC 和 ADO 两种数据库访问方式同时与 Oracle 数据库进行交互,分工合作,对字符数据、日期数据和 LOB 数据进行存储管理。在此申明,本文所有测试均在 Matlab 2012a 版和 Oracle 11g 企业版下进行。

在 Matlab 环境下,本文实现了通过 ODBC、JDBC 和 ADO 访问 Oracle 数据库的功能,并比较分析了它们访问 Oracle 数据库的各项性能,针对数据库访问过程中出现的一些问题给

收稿日期:2014-09-04;修回日期:2014-11-01。
作者简介:陈静(1990-),女,重庆人,研究实习员,硕士,主要研究方向:数据库设计与管理; 范乃吉(1987-),男,黑龙江哈尔滨人,研究实习员,硕士,主要研究方向:数值计算、结构设计; 袁晓东(1966-),男,四川绵阳人,研究员,博士,主要研究方向:高功率光纤激光技术; 蒋一岚(1987-),女,四川绵阳人,助理研究员,硕士,主要研究方向:数据库设计与管理。

出了可行的解决方法。

1 JDBC 访问技术

JDBC(Java Database Connectivity)是 Java 与数据库的接口规范,它定义了一个支持标准 SQL 功能的通用应用程序编程接口(API)。JDBC API 允许 Java 程序员发送 SQL 指令。通过驱动程序管理器,JDBC API 可利用不同的驱动程序连接不同的数据库系统,实现对数据库的跨平台访问^[2]。

由于 Matlab 和 JDBC 都是基于 Java 语言开发的,因此 Matlab 的 Database Toolbox 无条件支持 JDBC 驱动。

1.1 JDBC 驱动

要通过 JDBC 来存取某一特定的数据库,必须有相应的 JDBC 驱动,它往往是由生产数据库的厂家提供,是连接 JDBC API 与具体数据库之间的桥梁。

加载 JDBC 驱动过程如下:1)在安装好的 Oracle 软件的磁盘文件里找到含 JDBC 驱动的文件(其根目录下的 jdbc 文件夹里的 lib 子文件夹内);2)在 Matlab Java Classpath 中设置驱动路径,设置驱动路径分静态和动态两种。

1.1.1 Matlab Java Classpath 的静态配置方式

首先建立驱动配置文件存储路径,然后配置 matlab 的 classpath.txt 文件,具体过程是,打开... \matlab \toolbox \local 目录,编辑 classpath.txt 文件。添加行: \$ matlabroot \ dbtools \ ojdbc6.jar,并保存,最后重启 Matlab,这种改变将会保存。

1.1.2 Matlab Java Classpath 的动态配置方式

首先建立驱动配置文件存储路径,然后执行 javaaddpath('.\ojdbcX.jar')。在退出 Matlab 的时候,该种改变将不会被保存。该方法往往用于 Matlab 当前工作区或者特定工作区需要导入数据库。

如果要在 VQB 中使用 JDBC 方式访问数据库,JDBC 数据源文件配置过程如下:1)输入 querybuilder 或 confds,对话框打开后,点击 Query 下的 Define JDBC Data Source;2)点击 Create New File(初次使用时,一旦配置完成保存后,再次使用时点击 Use Existing File 即可);3)Specify new JDBC data source MAT file 对话框打开后。建立一个 Matlab MAT 文件来存储 VQB 中 JDBC 数据源的信息;4)在 Define JDBC Data Sources 对话框中,完成 Name, Driver, and URL 信息录入,在 Name 中键入数据源名,在 Driver 中键入 oracle.jdbc.driver.OracleDriver,在 URL 中键入 jdbc:oracle:thin:@ IP:port:orcl:dbname 点击 test,键入正确的信息,点击 ok。显示测试成功,至此,JDBCDataSourceFile 配置完成,再次需要使用时调用即可。

1.2 访问 Oracle 数据库

加载 JDBC 驱动成功后就可以连接访问数据库了。

连接代码如下:

```
conn = database('dbname','username','password',
... 'driver','databaseurl');
```

举例说明,使用 JDBC thin 驱动连接 Oracle 数据:

```
conn = database('dbname','user','password',...
'oracle.jdbc.driver.OracleDriver',...
'jdbc:oracle:thin:@machine:port:dbname');
```

当 val.AutoCommit 的值为'on'时,数据库连接成功。若需连接其他类型的数据库,则将 JDBC 驱动和数据库资源定位符(URL)改动成相应的,可参考 Matlab HELP 中的函数

database 的用法。

数据库连接成功后,利用 Database Toolbox 提供的接口函数结合可执行的 SQL 语句与 Oracle 数据库即可进行数据交互。

部分常用函数命令如下:

```
curs = exec(conn,'select columnname from tablename');
//从数据表 tablename 中读取列 columnname 的数据
ans = fetch(curs); //取出数据记录集
ans.data; //显示数据
insert(conn, table, colnames, exdata);
//将元数据 exdata 的值存入 tablename 中的 colnames
curs = exec(conn,'drop table ' tablenames); //删除表 tablename
curs = exec(conn,'delete from tablename where columnname = ');
//删除数据表 tablename 中的数据
curs = exec(conn,'commit'); //提交
close(curs); //关闭游标
close(conn); //关闭数据库连接
```

1.3 测试结果

为验证在 JDBC 访问方式下 Database Toolbox 提供的函数包的可行性和稳定性,使用了 1.2 节提供的函数在多种操作系统下的 Matlab 2012a 命令窗里进行反复的测试,得到了一系列的结果。

对于 Oracle 数据库里普通的字符型和数字型数据,任一操作系统均可以进行查询、插入、更新和删除;

对于二进制大型非结构化数据(Binary Large Object, BLOB)对象数据,Windows XP 和 Windows7 下均不能插入,查询取出时显示 <1 x1 oracle.sql.BLOB>;

对于 DATE 和 TIMESTAMP 日期型数据,Windows XP 和 Windows7 环境下采用静态 JDBC 驱动能正常查询显示和插入,采用动态 JDBC 驱动时查询显示结果不稳定,有时只能显示类似 oracle.sql.Timestamp@6sde56 的字符串,可使用 to_char() 函数将 DATE 和 TIMESTAMP 型数据在查询显示时强制转化为时间字符串来解决。

以上的实验现象充分证实了 Database Toolbox 和 VQB 以 JDBC 方式访问 Oracle 数据库时不能处理 BLOB 型数据,对 DATE 和 TIMESTAMP 日期型数据的处理最好采用静态 JDBC 驱动方式,以确保输出结果能正常显示。

1.4 注意事项

在 VQB 里配置 JDBC 数据源文件时需要注意以下两点

1)若不慎误删 JDBCDataSourceFile 配置文件,则需要使用命令 setdbprefs('JDBCDataSourceFile',''),将 JDBC 数据源文件置空,然后重新配置 JDBCDataSourceFile 数据源文件。

2)使用命令 setdbprefs('JDBCDataSourceFile',...,'\$ path \$ \datasource.mat')可以将之前配置好的 JDBC 数据源文件另存到 \$ path \$ 下,这里需要特别注意,path 里不能含中文字符,否则 VQB 不识别配置的 JDBC 数据源文件。

在数据库连接过程中可能会出现以下问题:

1)无法解析指定的连接标识符。

问题来源 连接标识符包含了连接数据库所必备的信息,如果连接标识符有误或者连接标识符指定的数据库并未在局域网中,监听在数据库连接过程中便找不到指定的数据库。

解决方法 打开 \$ database home \$ \NETWORK\ADMIN 下的 tnsnames.ora 文件,查看连接标识符是否与数据源信息匹配或者查看连接标识符指定的数据库所在服务器是否可连

通^[3]。

2) 监听程序当前无法识别连接描述符中请求的服务。

问题来源 连接标识符中指定的服务名没有在监听配置文件里注册,这种情况可能会在监听启动后,而数据库实例名被监听配置文件注册之前短暂出现。

解决方法 一段时间后再次进行连接;检查服务名是否与监听中的数据源信息匹配;通过重启服务的方式重新启动数据库,或者在 listener.ora 文件中指定监听的实例名,则即使数据库处于关闭状态,仍然可以连接。

3) 无监听程序或网络适配器不能建立数据库连接。

问题来源 监听服务未启动;服务器 ID 地址与监听配置文件里的 IP 地址不匹配。

解决方法 启动监听服务;重新配置监听。

4) 实例监听程序无法启动。

问题来源 修改机器名称或 IP 地址后,监听未更新。

解决方法 删除原有监听,重新建立一个,将 \$ database home \$ \NETWORK\ADMIN 下的 tnsnames.ora 和 lisenter.ora 文件下的所有 IP 信息改成正确的 IP 地址,然后重新启动监听和其他服务。

2 开放式数据库连接

开放式数据库连接(Open Database Connectivity, ODBC)是 Microsoft 的 Windows 开放服务体系(WOSA)的数据库部分,是提供给客户端应用程序访问关系数据库时使用的一个接口。应用程序通过调用 ODBC API 函数可以访问任何提供了 ODBC 驱动程序的数据源。ODBC 体系的特点是:通过统一的接口实现对 DBMS 的访问,使用不同的 ODBC 驱动程序可以保证数据库的独立性^[4-6]。

由于微软的 ODBC 不是用 Java 语言来编写的,不能在 Matlab 开发环境下直接访问数据库,因此 SUN 公司提供了一个 JDBC/ODBC 桥,使基于 Java 语言编写的 Database Toolbox 可以通过这个桥调用 ODBC 接口访问数据库。下面介绍通过 ODBC 访问 Oracle 数据库的一般步骤。

2.1 ODBC 数据源

为了连接到某个数据库,开发人员需要在程序中指定数据源名称(Data Source Name, DSN)。DSN 是包含了有关某个特定数据库信息的数据结构,这个信息是 ODBC 驱动能够连接到数据库上必需的信息。DSN 存储在注册表或作为一个单独的文本文件,DSN 里面包含的信息有名称、目录和数据库驱动器,以及用户 ID 和密码(根据 DSN 的类型)。

应用程序与数据库之间通过 JDBC/ODBC 桥来连接 ODBC 驱动,而 Matlab 与数据库之间的桥连接就是建立在配置成功的 DSN 基础之上的。其连接流程如图 1 所示。

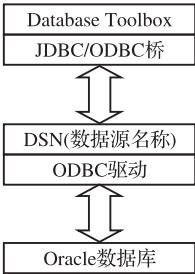


图 1 Matlab 访问 Oracle 数据库的流程

具体创建过程如下。

对于 Windows XP 和 Windows7 32 位,打开控制面板中的管理工具,单击 ODBC 数据源,或者在 VQB 中单击 Query 选项里的 Define ODBC Data Source,弹出 ODBC 数据源管理器窗口后,单击添加按钮,选择相应的驱动程序。

将数据库信息录入,点击确定,创建数据源完毕。

2.2 测试结果

在 Matlab 环境下执行命令 conn = database(...'DSN', 'username', 'password')。该命令以 ODBC 方式连接数据库,连接过程中可能出现的问题及解决方法同第 1.4 节;Oracle 数据库连接成功后,与 Matlab 的数据交互操作同 1.2 节。

为验证在 ODBC 访问方式下 Database Toolbox 提供的函数包的可行性和稳定性,使用了 1.2 节提供的函数在多种操作系统下的 Matlab 2012a 命令窗里进行反复的测试,得到了一系列的结果。

目前主要有 Microsoft ODBC for Oracle 和 Oracle in OraDB11g_home1 两个驱动可供选择。

1) 微软的 Microsoft ODBC for Oracle 驱动。

对于 Oracle 数据库里普通的字符型和数字型数据,任何一种操作系统均可以进行查询、插入、更新和删除;

对于 BLOB 对象数据,Windows XP 和 Windows7 32 位均不能插入,查询时报错 General error, Windows7 64 位系统内无微软的 64 位 Microsoft ODBC for Oracle 驱动,采用 32 位 Microsoft ODBC for Oracle 驱动时,报错“指定的 DSN 中,驱动程序和应用程序之间的体系结构不匹配”,目前,本文还没有解决这个问题。

对于 DATE 和 TIMESTAMP 日期型数据,Windows XP 和 Windows7 32 均出现查询时报错 General error 的情况,Windows7 64 出现的问题与查询 BLOB 型数据出现的问题相同。

2) 甲骨文公司的 Oracle in OraDB11g_home1 驱动。

对于 Oracle 数据库里普通的字符型和数字型数据,任何一种操作系统均可以进行查询、插入、更新和删除;

对于 BLOB 对象数据,任一操作系统均不能插入,查询取出时显示 <1 x 1 oracle.sql. BLOB > 字符串;

对于 DATE 和 TIMESTAMP 日期型数据,Windows XP 和 Windows7 能插入和正常查询显示。

以上的实验现象充分证实 Database Toolbox 和 VQB 以 ODBC 方式访问 Oracle 数据库时不能处理 BLOB 型数据,而且对 DATE 和 TIMESTAMP 日期型数据只能用甲骨文公司提供的驱动进行处理。

对于 Microsoft ODBC for Oracle 和 OraDB11g_home1 两种 ODBC 的驱动而言,Matlab 都是利用 JDBC/ODBC 桥来调用 ODBC,但 OraDB11g_home1 与 Matlab 兼容性更好。

3 ActiveX 数据对象

由于 Database Toolbox 和 VQB 对 LOB 大对象数据的存储管理不提供支持,而 LOB 大对象的存储管理的重要性却日益突出。为此,研究者们开发了 ADO 用来存储管理 LOB 大对象数据。

ADO (ActiveX Data Object)是 Microsoft 提出的继 ODBC、DAO、RDO、OLE 之后的又一种数据库访问应用程序接口 API。它是一种高层访问技术,可以使用任何一种 ODBC 数据源,即不止适合于 SQL Server、Oracle 等关系型数据库,也适

合于文本文件、Excel 表格、图形文件和其他无格式的数据文件^[7-9]。

3.1 ADO 驱动

要通过 ADO 来存取 Oracle 数据库的数据,必须有相应的 ADO driver,本文选用甲骨文公司的 OraOLEDB. Oracle 驱动(微软的 MSDAORA 驱动不支持 LOB 型数据的存储)。

OraOLEDB 驱动存在于 Oracle 软件安装磁盘的根目录下;对于远程 Oracle 数据库服务器的访问,需要安装 Oracle 客户端,客户端内含 OraOLEDB 驱动。

3.2 COM. ADODB 组件对象

ADO 与层次型的 DAO 和 RDO 不同,它是一种平面型的面向对象的编程接口,作为 ActiveX 的一部分,ADO 也是 Microsoft 的组件对象模式 (COM) 的一部分,它的面向组件的框架用以将程序组装在一起^[10]。ADO 由以下对象组成: Connection 对象、Command 对象、Recordset 对象、Field 对象、Parameter 对象、Property 对象和 Error 对象。但是最主要的对象有三个,即 Connection、Command、RecordSet,对于初级用户来说,只需要掌握其中的 Connection 对象、Command 对象和 RecordSet 对象就可以实现基本的数据库操作。至于 Error、Field、Parameter、Property 几个对象作为前面 3 个对象的子对象进行访问^[11-14]。

通过调用 Connection、Command、RecordSet 3 个基本对象就可以利用 ADO 与 Oracle 数据库进行各种数据类型的数据交互(包括 BLOB 型数据的存储和查询)。

在 Windows XP 和 Windows7 系统下进行测试,测试结果说明利用 ADO 能够对各种数据类型(包括 TIMESTAMP 和 BLOB 型数据)进行插入和查询显示。

3.3 LOB 型数据存储

下面给出了在 Matlab 环境下通过 ADO 方式调用 COM. ADODB 组件对象存取 LOB 大对象数据的一个例子。

Matlab 要与数据库进行数据交互,首先得通过连接字符串建立数据库连接。

1) 常用数据库的连接字符串示例如下:

```
MS SQL Server
connection_str=[ ...
    'PROVIDER = SQLOLEDB; ' ...
    'Data Source = ' myServerAddress '; '...
    'Initial Catalog = ' myDataBase '; ' ...
    'User Id = ' myUsername '; ' ...
    'Password = ' myPassword];

MySQL
connection_str=[ ...
    'driver = MySQL ODBC 3.51 Driver; ' ...
    'Server = ' myServerAddress '; ' ...
    'Database = ' myDataBase '; ' ...
    'UID = ' username '; ' ...
    'PWD = ' password];

Access
connection_str=[ ...
    'PROVIDER = Microsoft. Jet. OLEDB. 4.0; ' ...
    'Data Source = ' C:\mydatabase.mdb '; ' ...
    'User Id = ' myUsername '; ' ...
    'Password = ' myPassword];

Access 2007
connection_str=[ ...
    'PROVIDER = Microsoft. ACE. OLEDB. 12.0; ' ...
    'Data Source = ' C:\mydatabase.accdb '; ' ...
    'Persist Security Info = ' False];
```

```
Oracle
connection_str=[ ...
    'PROVIDER = OraOLEDB. Oracle; '...
    'Data Source = ' MyOracleDB '; '...
    'User Id = ' myUsername '; '...
    'Password = ' myPassword ];
```

2) 构建一个 activeX 控件并且打开数据库连接:

```
if nargin <2, timeout = 60; end
ado_connection = actxserver('ADODB.Connection');
//初始化 ADO command 对象
ado_connection.CursorLocation = 'adUseServer';
ado_connection.CommandTimeout = timeout;
ado_connection.Open( connection_str);
```

3) 将矩阵 matrix 存入 Oracle 数据库的函数 myblob_to_db 代码如下:

```
function myblob_to_db (ado_connection, table, ...
    column, where, file, matrix)
    ado_command = actxserver ('ADODB.Command');
    command_text = ['SELECT ', column, ' FROM ',
        table, ' ', ... where];
    ado_command.CommandText = command_text;
        //将 SQL 命令传递给 connection 对象
    try
        ado_command.ActiveConnection = ado_connection;
    catch err
    end
    ado_recordset = actxserver ('ADODB.Recordset');
    ado_recordset.CursorType = 'adOpenStatic';
    ado_recordset.LockType = 'adLockOptimistic';
    ado_recordset.Open (ado_command);
    ado_stream = actxserver ('ADODB.Stream');
    ado_stream.Type = 'adTypeBinary';
    ado_stream.Open
    file = [ tempname, '.mat']; //构建一个临时文件 file
    save (file, 'matrix'); //将矩阵 matrix 存进临时文件 file
    ado_stream.LoadFromFile (file);
    ado_recordset.Fields.Item(column).Value = ado_stream.Read;
    ado_recordset.Update
    ado_stream.Close;
    ado_recordset.Close;
```

4) 从 Oracle 数据库提取矩阵 matrix 的函数 myblob_from_db 代码如下:

```
function myblob_from_db (ado_connection, table, ...
    column, where, file)
    ado_command = actxserver ('ADODB.Command');
    command_text = ['SELECT ', column, ' FROM ', table, ' ',
        where];
    ado_command.CommandText = command_text;
    try
        ado_command.ActiveConnection = ado_connection;
    catch err
    end
    ado_recordset = actxserver ('ADODB.Recordset');
    ado_recordset.Open (ado_command);
    file = [ tempname, '.mat']; //构建一个临时文件 file
    file_handle = fopen (file, 'w');
    fwrite (file_handle, ...
        ado_recordset.Fields.Item(column).Value);
    fclose (file_handle);
    ado_recordset.Close;
    load (file);
    matrix
```

3.4 注意事项

在 Windows XP 系统下对 BLOB 型数据的存储查询时出现了一个问题:

执行 `ado_command.ActiveConnection = connection_str` 命令时, Matlab 系统报错 `Error: Object returned error code: 0XFFFFFFFE`。

经过反复测试验证,发现报错是由于 Windows XP 与 COM. ADODB 组件的不兼容,解决的方法是将 `ado_command.ActiveConnection = connection_str` 置于 `try/catch` 的 `try` 里,忽略执行 `ado_command.ActiveConnection = connection_str` 命令产生的错误。

Windows7 对 BLOB 型数据的存储查询时没有出现任何问题,说明 Windows7 对 COM. ADODB 组件的兼容性更好。

Windows XP 的 COM. ADODB 组件版本是 `Microsoft_ActiveX_Data_Objects_2.7_Library`。

Windows7 的 COM. ADODB 组件版本是 `Microsoft_ActiveX_Data_Objects_6.1_Library`。

4 3 种数据库访问技术的比较

在 Matlab 开发环境下, ODBC、JDBC、ADO 都能较好地与 Oracle 数据库进行交互访问,三者之间也有着紧密的联系。

ODBC 相比于 JDBC 和 ADO 出现得较早,已得到广泛的应用,几乎支持所有的关系型数据库,而且提供了可访问所有数据库的驱动程序。在 Matlab 开发环境下,通过 SUN 公司的 JDBC/ODBC 桥调用 Oracle 公司的 ODBC 驱动或者微软的 ODBC 驱动便可创建数据源连接 Oracle 数据库。但在两种驱动同时都存在的情况下,建议选用对于 DATE 和 TIMESTAMP 日期型数据的兼容性更好的 Oracle 公司的驱动。简而言之,

ODBC 使用简单;但是使用 ODBC 访问数据库速度较慢,而且 ODBC 的使用需要很多相应的部件支持,在不同系统间移植起来比较麻烦。

从功能上看,JDBC 与 ODBC 都不支持 BLOB 型数据。但相比于 ODBC, Matlab 的 Database Toolbox 和 VQB 对于 JDBC 的兼容性更好。且 ODBC、JDBC 和 ADO 3 种数据库访问方式中,JDBC 具有跨平台性,兼容性好。

ADO 是微软公司数据库访问应用程序接口的一种高层访问,与 ODBC、JDBC 相比有 4 个优点:1) ODBC、JDBC 所访问的数据源只能是关系数据库,可拓展性较差;而 ADO 是专门为关系型数据库或 ISAM 数据库设计的,它使用一种简单的编程方式获得任何地方的任何数据,具有很强的可扩展性。2) ADO 能在脚本语言中访问数据库,但是 ODBC、JDBC 不能。3) Mathworks 公司提供的 Database Toolbox 和 VQB 只支持 JDBC 和 ODBC 方式访问数据库,不支持 ADO 方式访问数据库;但是在 Matlab 开发环境下可以结合 Windows 下的 COM. ADODB 组件和 OLE DB 驱动以 ADO 方式访问数据库,并且完成 JDBC 和 ODBC 都不能完成的 LOB 型大对象数据的存储访问。4) ADO 作为高层数据库访问技术,具有比 ODBC、JDBC 更易于使用、访问速度更快、扩展性更好等性能,是目前最为流行的数据库访问技术。

但是,ADO 的这些优势并不意味着它目前能取代 ODBC 和 JDBC,因为 ODBC 和 JDBC 都有各自独特的优势:ODBC 使用简单;JDBC 是结合 Java 语言编写的数据库接口,具有跨平台性。在 Matlab 开发环境下,Database Toolbox 和 VQB 只支持 ODBC 和 JDBC。

表 1 给出了在 Matlab 开发环境下的三种数据库访问方式的综合性能比较结果。

表 1 JDBC、ODBC 与 ADO 综合性能比较

访问技术	易用性	访问速度	日期数据	BLOB 型数据	跨平台性	主要优势
JDBC	好	较快	支持	不支持	支持	与 Matlab 兼容性好
ODBC	好	较慢	不支持	不支持	不支持	使用简单
ADO	较好	快	支持	支持	不支持	支持 BLOB 大对象数据存储

对于远程访问 Oracle 数据库服务器, ODBC、JDBC 和 ADO 均可,设置好网络连接, ping 通 Oracle 数据库服务器后,在数据源(对于 ODBC)或者连接字符串(对于 JDBC 和 ADO)里加入要访问的数据库服务器的 IP 地址,其他同访问本机上的 Oracle 数据库一样。

5 结语

在 Matlab 开发环境下, ODBC、JDBC 和 ADO 三种数据库访问方式均可与 Oracle 数据库进行数据交互。在与 Oracle 数据库的数据交互中, JDBC 和 ADO 两种数据库访问技术的结合使用解决了在 Matlab 环境下不能存储图像数据这一难题。

这三种访问技术在 Matlab 开发环境下各有其优缺点。因此,在开发应用程序时应根据具体情况选择合适的数据库访问方式。值得注意的是,ADO 已作为目前最流行的数据库访问技术,其在各方面都具有特别突出的性能,如果未来的 Matlab 新版本的 Database Toolbox 部分能对 ADO 提供函数支持,将会使 Matlab 的数据库功能得到提高。

参考文献:

[1] MathWorks. Database toolbox user's guide[M]. [S. l.]: Math-

Works, 2013: 2-4.
[2] ZHU X, WU X. Discovering relational patterns without candidate generation[C]// Proceeding of International Conference on Data Engineering. Washington, DC: IEEE Computer Society, 2007: 726-735.
[3] JUNEAU J, BAKER J, NG V, *et al.* Databases and Jython: object relational mapping and using JDBC[M]// The Definitive Guide to Jython. New York: Apress, 2010: 231-261.
[4] 魏祖宽,江利娟,金在弘. 数据库访问技术比较研究[J]. 计算机与现代化,2009(12): 46-50.
[5] 肖飞,王东辉,陈敏,等. 面向多学科语义的运载火箭集成设计数据管理方法[J]. 国防科技大学学报, 2011, 33(4): 19-23.
[6] 俞建,张辽军. 数据库访问技术研究[J] 计算机与现代化,2004(10): 29-32.
[7] JUNEAU J, ARENA M. Accessing PL/SQL from JDBC, HTTP, Groovy, and Jython[M]// Oracle and PL/SQL Recipes. New York: Apress, 2010: 345-359.
[8] PALM W J. Introduction to Matlab 7 for engineers[M]. 3rd ed. [S. l.]: McGraw-Hill, 2007: 31-39.

MLR 方法,得到的结果不管是在小字典还是大字典,得到的分类结果都比前面的方法要好。这说明我们提出的 MLR 方法在字典大小的选择上更具紧凑性和鲁棒性。

3 结语

本文提出了一个 M 近邻的判别性低秩字典学习方法。学习的字典经过低秩表示去除噪声后增强了同类字典原子之间的线性相关性,再用 KSVD 算法更新字典,使字典原子更容易表示其测试样本。在分类中用 M 近邻方法,去除非同类的字典原子对该测试样本的影响,增强了分类器的精度,提高了分类性能,从而得到的更优的分类器。因此,MLR 方法在保持字典的表示性能的基础上,既增加了字典的紧凑性和纯粹性,又增强了字典的判别性,使字典的质量更优。该方法应用到扩展的 YaleB 和 AR 人脸数据集中,得到了较好的分类结果,甚至用更小的字典也能增强其分类性能。

参考文献:

- [1] YANG M, ZHANG L, YANG J, *et al.* Metaface learning for sparse representation based face recognition[C]// Proceedings of the 2010 17th IEEE International Conference on Image Processing. Piscataway: IEEE, 2010: 1601–1604.
- [2] ELAD M, AHARON M. Image denoising via sparse and redundant representations over learned dictionaries[J]. IEEE Transactions on Image Processing, 2006, 15(12): 3736–3745.
- [3] YANG J, YANG M. Top-down visual saliency via joint CRF and dictionary learning[C]// Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2012: 2296–2303.
- [4] WRIGHT J, YANG A Y, GANESH A, *et al.* Robust face recognition via sparse representation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2009, 31(2): 210–227.
- [5] AHARON M, ELAD M, BRUCKSTEIN A. K-SVD: an algorithm for designing overcomplete dictionaries for sparse representation[J]. IEEE Transactions on Signal Processing, 2006, 54(11): 4311–4322.
- [6] ZHANG G, JIANG Z, DAVIS L S. Online semi-supervised discriminative dictionary learning for sparse representation[C]// Computer Vision—ACCV 2012. Berlin: Springer, 2013: 259–273.
- [7] QIU Q, JIANG Z, CHELLAPPA R. Sparse dictionary-based representation and recognition of action attributes[C]// Proceedings of the 2011 IEEE International Conference on Computer Vision. Piscataway: IEEE, 2011: 707–714.
- [8] JIANG Z, ZHANG G, DAVIS L S. Submodular dictionary learning for sparse coding[C]// Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2012: 3418–3425.
- [9] YANG M, ZHANG D, FENG X. Fisher discrimination dictionary learning for sparse representation[C]// Proceedings of the 2011 IEEE International Conference on Computer Vision. Piscataway: IEEE, 2011: 543–550.
- [10] ZHANG Q, LI B. Discriminative K-SVD for dictionary learning in face recognition[C]// Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2010: 2691–2698.
- [11] JIANG Z, LIN Z, DAVIS L. Label consistent K-SVD: learning a discriminative dictionary for recognition[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(11): 2651–2664.
- [12] CHEN C-F, WEI C-O, WANG Y-C F. Low-rank matrix recovery with structural incoherence for robust face recognition[C]// Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2012: 2618–2625.
- [13] SHEN X, WU Y. A unified approach to salient object detection via low rank matrix recovery[C]// Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2012: 853–860.
- [14] CUI X, HUANG J, ZHANG S, *et al.* Background subtraction using low rank and group sparsity constraints[C]// Computer Vision—ECCV 2012. Berlin: Springer, 2012: 612–625.
- [15] MA L, WANG C, XIAO B, *et al.* Sparse representation for face recognition based on discriminative low-rank dictionary learning [C]// Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2012: 2586–2593.
- [16] LIN Z, CHEN M, MA Y. The augmented lagrange multiplier method for exact recovery of corrupted low-rank matrices [EB/OL]. [2014-08-01]. ftp://ftp.umi.acs.umd.edu/.snapshot/hourly.5/pub/swamiviv/662_project/papers/Lin09-MP.pdf.
- [17] LIU G, LIN Z, YU Y. Robust subspace segmentation by low-rank representation[C/OL]// Proceedings of the 27th International Conference on Machine Learning. Haifa: [s. n.], 2010 [2014-08-01]. <http://www.icml2010.org/papers/521.pdf>.
- [18] GOLUB G H, HANSEN P C, O'LEARY D P. Tikhonov regularization and total least squares[J]. SIAM Journal on Matrix Analysis and Applications, 1999, 21(1): 185–194.
- [19] XU Y, ZHANG D, YANG J, *et al.* A two-phase test sample sparse representation method for use with face recognition[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2011, 21(9): 1255–1262.

(上接第 82 页)

- [9] 王海亮, 张立民, 王海凤, 等. 精通 Oracle10g SQL 和 PL/SQL [M]. 北京: 中国水利水电出版社, 2007: 356–362.
- [10] 刘增军, 向为, 孙广富. 基于 ADO 的数据库开发技术研究[J]. 科学技术与工程, 2007, 7(5): 747–752.
- [11] 范冠雄. 基于 Visual ++ 的数据库访问技术比较研究[J]. 计算机与数字工程, 2011, 38(1): 64–67.
- [12] WADA Y, WATANABE Y, SYOUBU K, *et al.* Virtual database technology for distributed database in ubiquitous computing environment[J]. American Journal of Database Theory and Application, 2012, 1(2): 13–25.
- [13] IHLE M, FELDWSCH H, TEIXEIRA C A, *et al.* EPILEPSIAE—a European epilepsy database[J]. Computer Methods and Programs in Bio-medicine, 2012, 106(3): 127–138.
- [14] MAKINOCHI A. A consideration on normal form of not necessarily-normalized relation in the relational data model[C]// Proceedings of the Third International Conference on Very Large Data Bases. [S. l.]: VLDB Endowment, 1977, 3: 447–453.

作者：[陈静](#)，[范乃吉](#)，[袁晓东](#)，[蒋一岚](#)，[CHEN Jing](#)，[FAN Naiji](#)，[YUAN Xiaodong](#)，[JIANG Yilan?](#)
作者单位：[中国工程物理研究院 激光聚变研究中心, 四川 绵阳, 621900](#)
刊名：[计算机应用](#) 
英文刊名：[Journal of Computer Applications](#)
年，卷(期)：2015(z1)

参考文献(14条)

1. [MathWorks Database toolbox user' s guide](#) 2013
2. [ZHU X;WU X Discovering relational patterns without candidate generation](#) 2007
3. [JUNEAU J;BAKER J;NG V Databases and Jython:object relational mapping and using JDBC](#) 2010
4. [魏祖宽, 江利娟, 金在弘 数据库访问技术比较研究\[期刊论文\]-计算机与现代化](#) 2009(12)
5. [肖飞, 王东辉, 陈敏, 张为华 面向多学科语义的运载火箭集成设计数据管理方法\[期刊论文\]-国防科技大学学报](#) 2011(4)
6. [俞建, 张燎军 数据库访问技术研究\[期刊论文\]-计算机与现代化](#) 2004(10)
7. [JUNEAU J;ARENA M Accessing PL/SQL from JDBC, HTTP, Groovy, and Jython](#) 2010
8. [PALM W J Introduction to Matlab 7 for engineers](#) 2007
9. [王海亮;张立民;王海凤 精通Oraclele 10g SQL和PL/SQL](#) 2007
10. [刘增军, 向为, 孙广富 基于ADO的数据库开发技术研究\[期刊论文\]-科学技术与工程](#) 2007(5)
11. [范冠雄 基于Visual ++的数据库访问技术比较研究](#) 2011(01)
12. [WADA Y;WATANABE Y;SYOUBU K Virtual database technology for distributed database in ubiquitous computing envi-ronment](#) 2012(02)
13. [IHLE M;FELDWISCH H;TEIXEIRA C A EPILEPSI-AE-a European epilepsy database](#) 2012(03)
14. [MAKINOUCHI A A consideration on normal form of not necessari-ly-normalized relation in the relational data model](#) 1977

引用本文格式：[陈静](#)·[范乃吉](#)·[袁晓东](#)·[蒋一岚](#)·[CHEN Jing](#)·[FAN Naiji](#)·[YUAN Xiaodong](#)·[JIANG Yilan?](#) [Matlab环境下的Oracle数据库访问技术](#) [期刊论文]-[计算机应用](#) 2015(z1)