

# 基于 Web 的实时数据监测系统的研究与实践

贺超波<sup>1</sup>, 陈启买<sup>2</sup>

(1. 仲恺农业工程学院 计算机科学与工程学院, 广东 广州 510225;

2. 华南师范大学 计算机学院, 广东 广州 510631)

**摘 要:**针对目前各类工业生产监测系统要求实时性强、不受区域范围限制、维护方便等的需要,提出一种基于 Web 的,以 Ajax、SVG 为核心技术的实时数据监测系统设计方案。该方案应用 Ajax 开源框架 DWR 提高了实时监测数据获取的速度,并利用 SVG 可动态显示图形以及支持用户交互的特点解决了在 Web 上实时显示监测数据变化趋势曲线、单击数据点获取数据值以及数据变化趋势曲线历史回放等关键问题。基于该方案设计开发的金属腐蚀监测系统的应用结果表明该方案可提高基于 Web 的实时数据监测系统的实时性与灵活性。

**关键词:**实时数据监测; Ajax; DWR; SVG

**中图分类号:** TP39; TP393

**文献标识码:** A

**文章编号:** 1673-629X(2011)03-0200-05

## Research and Practice of Real-Time Data Monitoring System Based on Web

HE Chao-bo<sup>1</sup>, CHEN Qi-mai<sup>2</sup>

(1. Department of Computer Science and Engineering, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China;

2. Department of Computer, South China Normal University, Guangzhou 510631, China)

**Abstract:** Various kinds of industry production monitoring systems have the requirements of strong real-time, no limitation of accessing region and convenient maintenance. Therefore one design scheme of real-time data monitoring system based on Web was put forward, whose core technologies were Ajax and SVG. The scheme applied DWR, which was the open source framework supporting Ajax, to increase the speed of acquiring real-time monitoring data. By using the SVG, which had the ability of supporting dynamic graphic display and interaction with users, some key problems were solved, such as displaying the curve of real-time monitoring data change tendency, acquiring the data value by clicking the data point and playing back the history monitoring graphic. The application results of metal corrosion monitoring system based on this scheme showed that the scheme could improve the real-time and flexibility of real-time monitoring systems based on Web.

**Key words:** real-time data monitoring; Ajax; DWR; SVG

## 0 引言

如何提高监测系统的实时性,实现远距离实时监测以及提高系统的维护灵活性是各类工业实时数据监测系统设计者所关注的问题。传统的基于 Web 的实时数据监测系统虽然解决了基于 C/S 结构的监测系统维护成本高的问题,但在显示实时变化的监测数据处理上,需要不断地刷新浏览器页面才能获得最新的监测数据<sup>[1]</sup>,这种处理方式存在效率低、效果差以及操作不方便、不友好等问题,同时该类系统在监测数据图

表化实时展示以及用户与变化趋势图形的交互操作处理上也存在一定的局限性。基于以上问题,文中提出了一种新的基于 Web 的实时数据监测系统的设计方案,使用 Ajax 技术提高了系统的实时性,采用 SVG 动态图形展示技术实现了监测数据图形的实时展示以及同用户的交互处理。文中结合具体的系统实现实例对该方案所涉及的相关技术及关键功能的实现做了详细阐述。

## 1 相关技术概述

### 1.1 Ajax 异步通信模式

Ajax (Asynchronous Java and XML) 是一种基于浏览器的新的 Web 应用程序开发模式,利用多种技术的组合为系统提供了更多的交互性。通过运行在浏览器

收稿日期:2010-07-22;修回日期:2010-10-29

基金项目:广东省科技计划项目(2009B010800036)

作者简介:贺超波(1981-),男,广东河源人,硕士,讲师,CCF 会员,研究方向为数据库与软件设计。

中的以 JavaScript 为基础,以 XMLHttpRequest 为核心的 Ajax 引擎获取用户的输入,显示请求的数据并处理客户端的交互行为<sup>[2,3]</sup>。这相当于在用户和服务器之间加了一个中间层,使用户操作与服务器响应异步化。Ajax 具有的优势主要表现在:

1) 增强了 Web 应用程序同用户间的交互性。传统模式下,浏览器提交请求时客户只能等待。采用异步通信的模式,使服务器在处理用户请求的同时,客户端依然可以接收用户的指令。

2) 加快了响应速度,缩短了用户的等待时间。在传统的浏览器与服务器的交互访问中,即使是只需从服务器端得到一个简单的数据,都要返回一个完整的 Web 页。而 Ajax 采取的是异步交互模式,不需要刷新页面,只需从服务器返回需要的数据,这样就降低了网络信息传输量,缩短了响应时间。Ajax 具备的以上优势使得在 Web 上不间断、实时、“无抖动”地显示监测数据成为可能,同时系统与用户的交互操作将更加方便友好。

## 1.2 DWR 开源框架

DWR(Direct Web Remoting)是一个支持 Ajax 技术的 JAVA 开源框架,提供了一种简单易用的方式使用 Ajax 和 XMLHttpRequest。DWR 包含服务器端 Java 类库、一个 DWR Servlet 以及 JavaScript 库,它可以让浏览器上的 JavaScript 方法调用运行在 Web 服务器上的 Java 类的方法。DWR 很好地解决了 Java 方法调用的同步特性与 Ajax 异步特性之间的矛盾,它把回调函数(callback method)作为参数传递给 Java 方法,响应返回后自动调用回调函数,从而实现了 Ajax 技术应用的异步通信机制。应用 DWR 开源框架进行监测系统的开发,实时数据的获取与处理方法可以应用 Java 进行设计实现,相应的 JAVA 类部署在服务端,Web 用户程序中可应用 JavaScript 编写程序远程调用该方法从而可以获得到实时监测数据<sup>[4,5]</sup>。

## 1.3 SVG 动态图形展示技术

SVG(Scalable Vector Graphics)即可伸缩向量图形,是一种基于 XML 的、开放的矢量图形格式,具有动态、可缩放和平台无关等优点。使用 SVG 可以在 Web 上显示出高质量的矢量图形,支持几何图形、动画、渐变色、滤镜效果等,并可嵌入到现有基于浏览器的 Web 应用程序中,可以对图形进行缩放,而不会有质量损失;同时 SVG 具有其它图形格式如 JPEG、GIF、PNG 等所没有的特性,可支持 JavaScript 的 Document Object Model(DOM)编程,支持与用户的交互操作,支持编写脚本代码进行各类触发事件的处理,从而可实现图形的动态展示。SVG 具备的以上特性可以实现在 Web 上显示实时监测数据变化趋势图形,并可对监测图形

进行交互操作,如单击获取数据点数值以及图形放大缩小等功能<sup>[6-11]</sup>。

## 2 系统体系结构设计

工业生产中的实时数据监测系统往往通过各类可靠的监测设备进行数据采集存储,形成监测数据库,然后以监测数据库为基础通过不同的监测数据展示程序显示监测数据变化状况。结合 Ajax、DWR 以及 SVG 技术的特点,文中设计了如图 1 所示的基于 Web 的实时数据监测系统的体系结构。

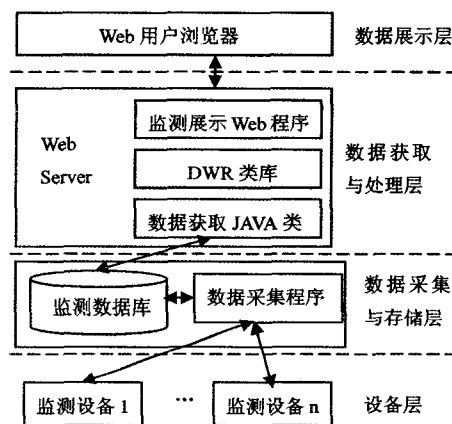


图1 基于 Web 的实时数据监测系统体系结构

图1所示的基于 Web 的实时数据监测系统体系结构基于四层架构,最底层为设备层,设备层包含各类监测设备用于监测各类工业生产状态数据,如石油化工行业输油管道腐蚀监测设备,可以实时获得管道金属损失(Metal Loss)、温度(Temperature)等值。与这些监测设备的通讯往往要借助遵从该设备通讯协议设计的数据采集程序,数据采集程序可通过计算机串口或并口通信方式获取监测设备的实时监测数据并存储于监测数据库,它与监测数据库共同组成了系统体系结构的数据采集与存储层。数据获取与处理层包含根据指定参数从监测数据库中获取数据的 JAVA 类,用于实现 Ajax 异步通信机制的 DWR 开源类库以及用于展示实时监测数据动态变化图形的 Web 界面程序,Web 界面程序主要包含 SVG 图形格式描述代码以及动态操作 SVG 图形实现绘制实时数据点、曲线等的 JavaScript 代码。数据获取与处理层所需要的程序均部署在 Web Server 中,如 Tomcat、Weblogic 等,而 Web Server 可处理来自数据展示层 Web 用户浏览器的连接请求。整个系统体系结构的设计遵从从开放、与平台无关的特点,并充分考虑到了基于 Web 的实时数据监测系统的实时性要求。

## 3 关键功能实现

基于以上系统体系结构进行了工业实时数据监测

系统的实际应用开发,设计开发了石油化工输油管道金属腐蚀实时监测系统,该系统的监测设备以及数据采集程序能对金属腐蚀损失、温度等参数值进行采集与存储<sup>[12]</sup>,监测数据的图表化展示是该系统的核心组成部分,下面主要对该系统如何基于 Web 进行实时数据获取、显示实时数据表格、显示变化趋势图形、对图形进行交互操作以及历史图形回放等关键功能的实现进行详细阐述。

### (1) 实时数据获取。

DWR 开源框架能够支持 Web 程序中的 JavaScript 方法调用 JAVA 类中的方法,所以实时数据的获取程序可由 JAVA 类负责实现,主要设计一方法负责从腐蚀监测数据库中抽取数据。该金属腐蚀监测数据库中的主要数据包含有设备号(slave\_number)、腐蚀损失(metalloss)、温度(temperature)以及记录时间(record\_time)等值,则可设计一 JAVA 类 Corrosiondata 封装这些参数以及操作这些参数的相应方法,设计数据获取类 Corrosiondatalist,该类主要包含一方法 getdataList,该方法用于获取某设备指定起始监测时间的数据,返回的是 List 类型列表数据。Corrosiondata 与 Corrosiondatalist 类的结构如图 2 所示。

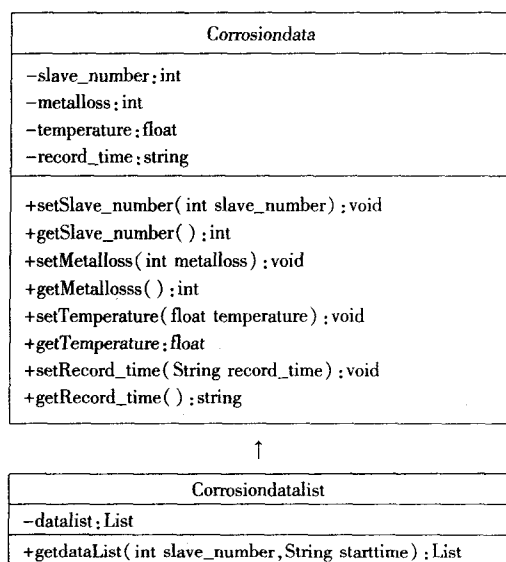


图 2 实时数据获取 JAVA 类结构

Web Server 部署以上的 JAVA 类后,为可以在 Web 中嵌入 JavaScript 代码远程调用 Corrosiondatalist 类中的 getdataList 方法,Web Server 端需对 DWR 框架核心文件 dwr.xml 文件进行正确配置,主要配置如下:

```

<create javascript="jsdatalist" creator="new" scope="application">
  <param name="class" value="Corrosiondatalist"></param>
  </create>
  <convert match="Corrosiondata" converter="bean">

```

```

  <param name="include" value="slave_number, metalloss,
  temperature, record_time"/></convert>

```

### (2) 显示实时数据表格。

首先在 Web 程序中通过 JavaScript 设计相应函数远程调用 getdataList 方法获取指定参数的实时监测数据,然后可通过 DWR 类库中的工具类 DWRUtil 所提供的表格操作函数生成实时数据表格,具体处理代码如下所示:

```

function GetrealtimeData( )
{
  jsdatalist.getdataList( slavenumber, starttime, callBackMethod );
}

function callBackMethod( dataList )
{
  for( var i in dataList ) {
    DWRUtil.removeAllRows( "datatable" );
    DWRUtil.addRows( "datatable",
      dataList, [ function( item ) { return item.slave_number; },
        function( item ) { return item.metalloss; },
        function( item ) { return item.temperature; },
        function( item ) { return item.record_time; } ],
        { escapeHtml: false } );
  }
}

```

其中 callBackMethod 为回调函数用于实现 JAVA 的同步机制与 Ajax 的异步机制的协同,其参数 dataList 用于存储 getdataList 方法所返回的腐蚀数据列表,removeAllRows 与 addRows 为操作表格的函数,参数值“datatable”为 Web 中一 HTML 表格 ID 值。

### (3) 显示实时数据变化趋势图形。

实时数据图形化展示涉及到数据点以及变化趋势曲线的动态绘制,SVG 图形格式提供了圆点、折线等图形元素,所以在获取到实时数据的基础上,根据坐标刻度计算出各数据点的位置后,应用 JavaScript 操作 SVG 的 DOM 方式动态地为 SVG 图形添加相应的圆点图形元素进行表示,后绘制经过这些圆点的折线从而可以实现动态显示实时监测数据变化趋势曲线的效果。具体处理步骤如下所示:

首先在 Web 页面中嵌入 SVG 图形,代码如下:

```

<embed id="monitorsvg" src="monitor.svg" width="1020"
height="570"
type="image/svg+xml" />

```

为获取到 SVG 图形的 DOM,在 SVG 图形格式文件中加入 onload 事件处理函数 main( evt ),该函数主要代码如下:

```

function main( evt ) {
  svg_doc = evt.target.ownerDocument; //初始化 SVG 文档对象
  curvearea = svg_doc.getElementById( "curvearea" ); //获得曲线绘制区域
  draw_Y_axis(); //绘制 Y 轴刻度值函数, Y 轴代表 metalloss
  draw_X_axis(); //绘制 X 轴刻度值函数, X 轴代表 record_

```

time

... }

在已确定坐标刻度值( $x\_step$  与  $y\_step$ )的基础上,通过计算各数据的金属损失、温度以及时间值与坐标 0 点( $x\_zero, y\_zero$ )的相对位置可以确定数据点的绘制位置,如要确定某数据(金属损失,时间)点的绘制位置,可采用如下代码进行处理:

```
function draw_point(p_metalloss, p_time, i) // i 为数据点编号
{
    node = svg_doc.createElement("circle"); // 创建 circle 图形元素用于表示数据点
    x = x_zero + ((p_time - initialtime) / x_step); // 确定 X 坐标位置, initialtime 为初始时间值
    y = y_zero - ((p_metalloss - initialmetalloss) / y_step); // 确定 Y 坐标位置, initialmetalloss 为初始金属损失值
    node.setAttribute("cx", x); node.setAttribute("cy", y); // 设置数据点位置
    node.setAttribute("r", 3); node.setAttribute("id", "p" + i); // 设置数据点编号
    node.setAttribute("metalloss", p_metalloss); // 设置 metalloss 值
    node.setAttribute("record_time", " + p_time); // 设置 record_time 值
    ...
    curvearea.appendChild(node); // 绘制数据点
}
```

在确定数据点的位置后,绘制数据点的同时可以绘制经过该实时监测数据点的折线图形元素从而实现监测数据曲线实时变化的效果,相应处理代码如下:

```
function draw_curve(p_x, p_y) // 绘制曲线
{
    path = path + p_x + " " + p_y + " "; // 设置折线路径数据点, p_x, p_y 为数据点的位置
    curve = svg_doc.getElementById("curve"); // curve 为 SVG 图形中的折线图形元素
    curve.setAttribute("points", path); // 绘制折线图形元素
}
```

#### (4) 监测图形交互操作处理。

实时监测数据的变化趋势图形常常需要接受用户的交互操作,如单击数据点获取数据值、放大缩小区域图形等。SVG 支持与用户的交互,通过 JavaScript 可动态处理用户对 SVG 图形的操作。以单击数据

点获取数据值为例,如金属腐蚀监测系统中需计算两选定数据点之间的腐蚀速率,需支持鼠标对数据点的选取操作,该操作可以按如下方式进行处理:

在绘制各实时数据点时增加单击数据点事件处理函数,具体处理代码如下所示:

```
node.setAttribute("onclick", "calculate(" + i + ")"); // i 代表数据点的编号值
calculate(i) 的处理如下:
function calculate(i)
{
    node = svg_doc.getElementById("p" + i); // 获取编号为 "pi" 的数据点
    p_metalloss = node.getAttribute("metalloss"); // 获取数据点 metalloss 值
    p_time = node.getAttribute("record_time"); // 获取数据点 record_time 值
    ...
}
```

#### (5) 监测图形历史回放。

实际应用中实时监测数据变化趋势曲线绘制位置在越过当前时间轴最大值刻度时需能够自动后移,同时自动变化时间轴的起始时间以及重新绘制数据点以及曲线,这样才能够连续地在 Web 上显示实时数据的变化趋势曲线;另外为能够查看历史变化趋势曲线,了解实时监测数据的整体变化状况,应提供曲线的前移与后移等监测图形历史回放操作。

考虑到实时监测系统存在数据采集时间间隔短、数据量大的情况,为提高数据点以及曲线的绘制效率以及节省 Web 用户程序所占用的系统资源,实时数据变化趋势曲线的前移与后移操作处理在获取监测数据

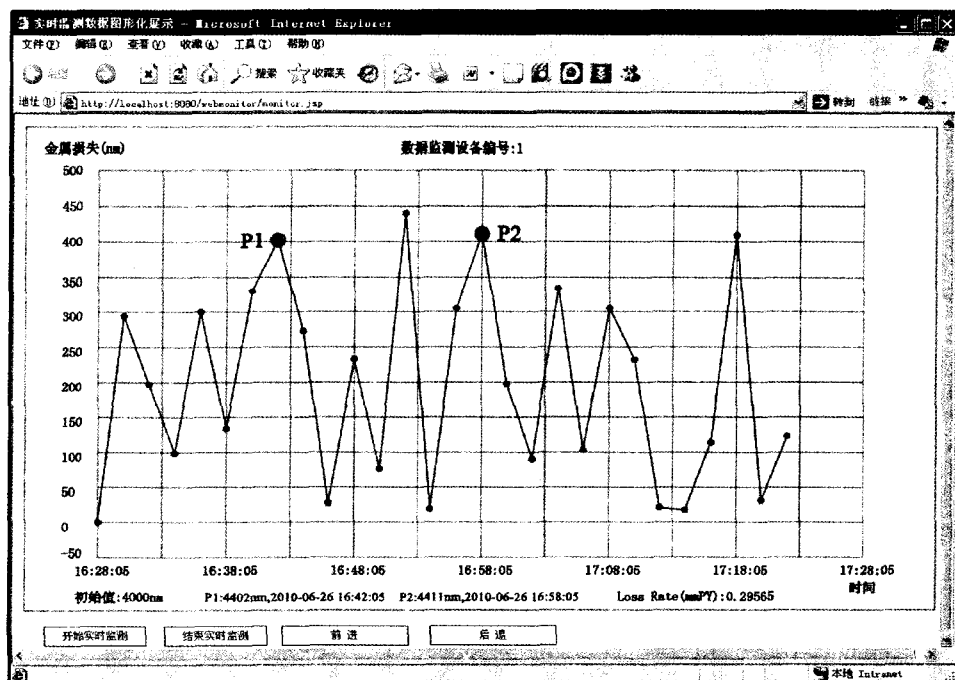


图3 基于 Web 的金属腐蚀实时监测系统

中的数据时不应一次性选取全部数据,比较优化的方法是只选取当前时间轴的最小刻度值与最大刻度值之间的数据。然后重新绘制该时间范围内的数据点以及曲线,相应的处理步骤如下所述:

1) 删除当前表示数据点以及曲线的 SVG 图形元素。

2) 计算当前时间轴的最小刻度值 (Min\_time) 与最大刻度值 (Max\_time)。

3) 调用 getdataList 方法获取 [Min\_time, Max\_time] 范围内的数据。

4) 调用 draw\_point 与 draw\_curve 方法重新绘制数据点以及变化趋势曲线。

最终所设计开发的基于 Web 的金属腐蚀实时监测系统的测试运行效果如图 3 所示。

#### 4 结束语

文中在对 Ajax 异步通信机制进行分析的基础上,同时对 DWR 开源框架以及 SVG 动态图形展示技术做了深入研究,设计了基于 Web 的实时数据监测系统的体系结构,并对其中关键的实时数据图表化展示功能的实现做了详细说明。

以此为基础设计开发的输油管道金属腐蚀监测系统在现实的应用中也体现出了在 Web 环境下进行实时监测的有效性和灵活性,在不需刷新 Web 页面的条件下,实现了实时数据的动态图表化展示,该系统的具体实现对其余类似的工业生产实时数据监测系统的开

发将有较好的借鉴作用。

#### 参考文献:

- [1] 李秀学,何存富. 基于 B/S 结构的嵌入式实时监测系统研究[J]. 测控技术,2006(7):70-72.
- [2] Palson L D. Building rich Web application with Ajax[J]. Computer,2005(10):14-17.
- [3] 刘迎春. 利用 Ajax 技术实现 WebGIS 中异步数据交互的研究[J]. 计算机测量与控制,2007(5):698-700.
- [4] 张 栗,张凤元,危胜军. 基于 DWR 框架的 WEB 应用的设计与实现[J]. 计算机技术与发展,2008,18(8):84-87.
- [5] DWR: Easy AJAX for JAVA[EB/OL]. [2009-06-20]. <http://directwebremoting.org/dwr/introduction/index.html>.
- [6] 黄凯伟. SVG 开发实践[M]. 北京:电子工业出版社,2008.
- [7] 陈传波,王 菁,邓 凯. 基于 SVG 的实时数据动态发布技术的研究[J]. 小型微型计算机系统,2002(5):609-612.
- [8] 周文生. 基于 SVG 的 WebGIS 研究[J]. 中国图象图形学报,2002(7):70-75.
- [9] 杨 斌,张利欣,章立军. 基于 SVG 的 Web 远程实时监测客户端研究[J]. 计算机应用研究,2010(6):2144-2146.
- [10] Adobe on SVG[EB/OL]. [2008-08-15]. <http://www.adobe.com/svg>.
- [11] 叶 锐,任 静. SVG 图像技术在 WebGIS 中的应用研究[J]. 计算机技术与发展,2007,17(1):179-182.
- [12] 贺超波,陈启买,罗金群,等. 基于 Modbus 协议的金属腐蚀监测系统的设计与实现[J]. 现代计算机,2006(8):49-52.

(上接第 199 页)

营销,然而由于电信行业的特殊性,导致数据存在大量的噪音,甚至某些用户的自然信息等数据为空值等,这导致单分类器的预测精确度无法达到满意的效果。文中以此为切入点,将 AdaBoost 算法引入客户消费行为分析研究中,提出基于 BP 神经网络的集成分类器,不仅提高了模型的预测精确度,还为以后的深入研究做好了很好的铺垫。

#### 参考文献:

- [1] 于红蕾,华庆一,刘燕玲,等. 数据仓库在电信统计分析中的应用[J]. 计算机技术与发展,2007,17(8):201-204.
- [2] 王纯麟,何建敏. 基于 AdaBoost 的电信客户流失预测模型[J]. 价值工程,2007(2):107-111.
- [3] 郭 明,郑惠莉,卢毓伟. 基于贝叶斯网络的客户流失分析[J]. 南京邮电学院学报,2005(5):84-88.
- [4] Freund Y, Schapire R E. A Decision-theoretic Generalization of Online Learning and an Application to Boosting[J]. Journal of computer and System Sciences, 1997(1): 119-139.
- [5] Breiman L. Bagging Predictors[J]. Machine Learning,1996:

24(2):123-140.

- [6] 姜 远,周志华,谢 琪,等. 神经网络集成在肺癌细胞识别中的应用[J]. 南京大学学报(自然科学版),2001,37(5):529-533.
- [7] Valiant L G. A Theory of Learnable[J]. Communication of ACM,1984,27:1134-1142.
- [8] 林存炜. 基于 Boosting 的人工神经网络集成及其模式分类[D]. 西安:西安电子科技大学,2005.
- [9] Freund Y, Schapire R E. A Short Introduction to Boosting[J]. Journal of Japanese Society for Artificial Intelligence, 1999, 14(5):2-3.
- [10] 于 玲,吴铁军. 集成学习:Boosting 算法综述[J]. 模式识别与人工智能,2004,17(1):52-59.
- [11] 陈春玲,商子豪. 基于 AdaBoost 和概率神经网络的人侵检测算法[J]. 南京师范大学学报(工程技术版),2008,8(4):22-27.
- [12] Drummond C, Holte R C. C4.5, Class Imbalance, and Cost Sensitivity: Why Under-sampling Beats Over-sampling[C]//In ICML 2003 Workshop on Learning from Imbalanced Data Sets, International Conference on Machine Learning. Washington, DC:[s. n.], 2003.