第16卷 第4期
2006年4月

计算机技术与发展
COMPUTER TECHNOLOGY AND DEVELOPMENT

Vol. 16 No. 4
Apr. 2006

C4.5算法在列车轨道故障检测上的应用研究

肖秋根,王成友,梁 华,刘云辉
(国防科学技术大学 电子科学与工程学院,湖南 长沙 410073)

摘 要:列车轨道故障检测的实现需要对大量的数据进行分析来判定检测结果,决策树是进行数据挖掘与分类分析的常用工具。文中主要讨论如何应用C4.5算法构造列车轨道故障检测的决策树以及根据生成的决策树实现轨道故障的判决。

关键词:C4.5算法;决策树;轨道故障检测
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1005-3751(2006)04-0076-03

Application of C4.5 Algorithm in Train's Rail Deformation Detection

XIAO Qiu-gen, WANG Cheng-you, LIANG Hua, LIU Yun-hui
(College of Electronic Sci. and Eng., National Univ. of Defense Techn., Changsha 410073, China)

Abstract: To realize the train's rail deformation detection, need deal with volumes of information to predict the detection result. And the decision tree is the most popular tool to data mining and classification. This paper mainly discusses how to build a decision tree of rail deformation detection by using C4.5 algorithm and how to make decision of the rail deformation by building decision tree.

Key words: C4.5 algorithm; decision tree; rail deformation detection

0 引 言

轨道是铁路运输的基础,轨道状态的好坏直接影响列车运行的安全和稳定。为使轨道长期处于良好的平顺状态,必须经常对轨道状态进行检查和测量,查找各种轨道病害和伤损部件,并及时实施养护维修和设备更新。随着铁路事业的发展,从事轨道故障检测研究的单位也越来越多,方法多种多样。有基于图像的轨道故障检测方法,以及基于超声波的轨道故障检测方法等。文中借鉴文献[1]提到的方法,讨论的是在采集加速度信号的基础上应用C4.5算法[2-5]构造轨道故障检测的决策树,对轨道故障进行判决。

利用决策树对列车轨道故障进行判决,首先通过分析处理现有的列车轨道故障数据资料来得到生成决策树的训练数据;然后通过训练数据生成的决策树对列车运行实时数据进行分析得到列车运行轨道的实时状况。

试验数据是列车在沈大线上运行采集,采集了不同位置的数据,及不同时刻相同位置的数据等。在试验过程中,可以增加或减少生成决策树的属性个数来获取最佳的效果。

1 应用C4.5算法构造列车轨道故障检测决策树

决策树学习是以实例为基础的归纳学习算法,它着眼于从一组无次序、无规则的事例中推理出决策树表示的分类规则。构造决策树的目的是找出属性和类别间的关系,用它来预测将来未知类别的记录的类别。C4.5算法是用于构造决策树的经典算法之一。

1.1 C4.5算法

C4.5算法是Quinlan于1993年提出的,是在ID3算法基础上发展起来的决策树生成算法。

较之于ID3算法,C4.5算法克服了它在应用中的不足,主要体现在以下几个方面:

(1) 用信息增益率来选择属性,克服了用信息增益选择属性时偏向选择取值多属性的不足;

(2) 在树构造过程中或者构造完之后进行剪枝;

(3) 能够完成对连续属性的离散化处理;

(4) 能够对不完整数据进行处理;

(5) 能够生成规则。

C4.5算法主要步骤:

设 T 为数据集,类别集合为 $\{C_1, C_2, \cdots, C_k\}$,选择一个属性 V 把 T 分为多个子集。V 有互不重合的 n 个取值 $\{v_1, v_2, \cdots, v_n\}$,则 T 被分为 n 个子集 $T_1, T_2, \cdots, T_n$,其中 T_i 中所有实例的取值均为 v_i。

令 $|T|$ 为数据集 T 的例子数,$|T_i|$ 为 $V = v_i$ 的例子数,$|C_j| = \mathrm{freq}(C_j, T)$ 为 C_j 的例子数,$|C_{jv}|$ 是 $V = v_i$ 例子中,具有类别 C_j 的例子数。

则有:

(1) 类别 C_j 的发生概率为

$$P(C_j) = |C_j| / |T| = \mathrm{freq}(C_j, T)$$

收稿日期:2005-09-11
作者简介:肖秋根(1981-),女,江西泰和人,硕士研究生,研究方向为智能信息系统技术;刘云辉,教授,研究方向是智能信息系统技术。

(2) 属性 $V = v_i$ 的发生概率为

$P(v_i) = |T_i| / |T|$

(3) 属性 $V = v_i$ 的例子中,具有类别 C_j 的条件概率为

$P(C_j | v_i) = |C_{jv}| / |T_i|$

类别信息熵计算:

$$H(C) = -\sum_j P(C_j) * \log P(C_j) = -\sum_j \frac{|C_j|}{|T|}\log\frac{|C_j|}{|T|} = -\sum_{j=1}^{k}\frac{\text{freq}(C_j,T)}{|T|}\log\frac{\text{freq}(C_j,T)}{|T|} = \text{info}(T) \tag{1}$$

类别条件熵计算:

$$H(C|V) = -\sum_j P(v_j)\sum_i P(C_j|v_i)\log P(C_j|v_i) = -\sum_j \frac{|T_j|}{|T|}\sum_i \frac{|C_{jv}|}{|T_i|} * \log\frac{|C_{jv}|}{|T_i|} = \sum_{i=1}^{n}\frac{|T_i|}{|T|} * \text{info}(T_i) = \text{Info}_v(T) \tag{2}$$

信息增益计算:

$$I(C,V) = H(C) - H(C|V) = \text{info}(T) - \text{Info}_v(T) = \text{gain}(V) \tag{3}$$

属性 V 的信息熵计算:

$$H(V) = -\sum_i P(v_i) * \log P(v_i) = -\sum_{i=1}^{n}\frac{|T_i|}{|T|} * \log\frac{|T_i|}{|T|} = \text{split} - \text{info}(V) \tag{4}$$

信息增益率计算:

$$\text{gain} - \text{ratio}(V) = I(C,V)/H(V) = \text{gain}(V)/\text{split} - \text{info}(V) \tag{5}$$

1.2 列车轨道故障检测属性选取以及决策树的生成

1.2.1 属性选取

文中讨论的方法是在采集的加速度信号基础上应用 C4.5 算法生成决策树。采集的加速度包括车体的水平加速度、车体的垂直加速度,以及转向架的水平加速度。生成决策树选取属性如表 1 所示。

表 1　类别以及属性

判决类别:轨道故障,无故障……	
属性名称	属性值
里程	continuous
速度	continuous
车体垂直加速度方差	continuous
车体水平加速度方差	continuous
转向架水平加速度方差	continuous
道岔信息	true, false
……	……

注:表 1 中 continuous 表示属性值是连续的数值,省略号表示类别以及属性都可以根据需要来进行添加。

类别 C 的集合为{轨道故障,无故障},属性 T 的集合为{里程,速度,车体垂直加速度方差,车体水平加速度方差,转向架水平加速度方差,道岔信息}。其中里程和速度属性是为了标明故障地点以及列车速度,对判决没有影响,所以在决策树剪枝时会被除去;一般在有道岔的地方火车的振动就会比较大,所以可以首先通过道岔信息属性来排除非故障点;加速度可以反映轨道的振动的大小,而其方差可以反映加速度值的波动大小,通过加速度值的方差这个属性可以判定某一地点是否有轨道故障引起的振动。文中方法还可以根据应用加入需要的属性,例如弯道、天气、桥梁、司机驾驶水平等;以及把故障分类细化,例如一级故障、二级故障等。

1.2.2 样本数据获取

样本数据需要全面客观地反映列车轨道的情况,这样训练生成的决策树才能够正确地对采集数据做出判决。文中训练样本数据是通过列车在沈大线上采集的大量数据进行分析处理后得到,基本能够反映沈大线列车轨道情况。抽样训练样本数据如表 2 所示。

表 2　抽样训练样本数据

里程 (m)	速度 (km/h)	车体垂直加速度方差	车体水平加速度方差	转向架水平加速度方差	道岔信息	…	类别
88743	116	0.057970	0.061645	0.415085	false	…	轨道故障
969	30	0.053623	0.037376	0.164557	true	…	无故障
1650	43	0.122037	0.079937	0.248043	false	…	轨道故障
7757	84	0.060141	0.059801	0.200410	false	…	轨道故障
251996	137	0.085514	0.086312	0.374930	false	…	无故障
252040	137	0.072434	0.094163	0.354962	false	…	无故障
…	…	…	…	…	…	…	…

表 2 中车体垂直、车体水平以及转向架水平加速度方差是列车上 3 个不同位置加速度传感器采集的加速度值的方差。

1.2.3 生成决策树

应用 C4.5 算法的列车轨道故障检测决策树生成步骤如图 1 所示。

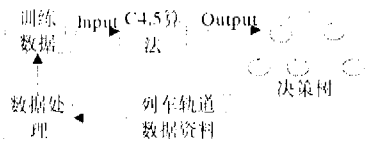

图 1　决策树生成流程

根据图 1 给出的流程图以及 C4.5 算法步骤以及事后决策树的剪枝法,用 VC++ 平台加以实现,运行可以得到如下的决策树,如图 2 所示。

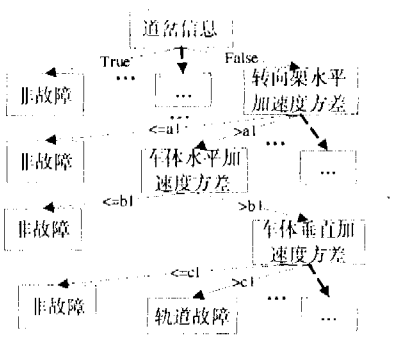

图 2　决策树结构

为了更清楚地理解决策树的知识表示，可以把它转化成规则的形式，如下：

If(道岔信息＝True) then (非故障)

If(道岔信息＝False) and(转向架水平加速度方差≤a1) then (非故障)

If(道岔信息＝False) and (转向架水平加速度方差＞a1) and (车体水平加速度方差≤b1) then (非故障)

If(道岔信息＝False) and (转向架水平加速度方差＞a1) and (车体水平加速度方差＞b1) and (车体垂直加速度方差≤c1) then (非故障)

If(道岔信息＝False) and (转向架水平加速度方差＞a1) and (车体水平加速度方差＞b1) and (车体垂直加速度方差＞c1) then (轨道故障)

图 2 中 a1，b1，c1 分别表示转向架水平加速度方差的分类门限、车体水平加速度方差的分类门限以及车体垂直加速度方差的分类门限；省略号表示在试验过程中，不断完善加入新的属性以及类别时的决策树结构。

应用决策树进行轨道故障判决流程如图 3 所示。

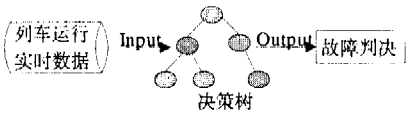

图 3 决策树判决流程

1.3 实例分析

现以列车在沈大线上运行采集的数据为例，对算法进行验证，试验结果如表 3 所示，通过统计判决正确概率基本达到预期效果。本文方法中属性主要还只是考虑了 3 个加速度的方差值，类别也只是分了轨道故障和无故障两类。在以后的试验以及工作中根据实际遇到的情况，可以加入更多的相关属性以及可以更加详细地对故障分类，使得故障类别更加清楚，判决更趋于合理化，同时判决的正确概率也得到提高。

表 3 试验结果

里程(m)	速度(km/h)	车体垂直加速度方差	车体水平加速度方差	转向架水平加速度方差	道岔信息	…	判决结果
5793	85	0.079934	0.077401	0.520618	false	…	轨道故障
7152	91	0.099115	0.080663	0.392309	false	…	轨道故障
8636	83	0.113714	0.106451	0.362360	true	…	无故障
9266	84	0.076295	0.059881	0.282721	false	…	无故障
10100	86	0.064951	0.076098	0.440309	false	…	无故障
31458	88	0.092874	0.084715	0.379678	false	…	轨道故障
…	…	…	…	…	…	…	…

2 结束语

决策树在市场划分、金融风险、产品开发以及故障诊断中已经得到了比较广泛的应用。C4.5 算法是决策树的一个经典算法，文中把 C4.5 算法应用到列车轨道故障的判决中，通过对样本数据的学习训练生成决策树，根据生成的决策树来对未知的输入数据进行决策，实现故障检测的自动化和智能化，具有广阔的应用前景。

参考文献：

[1] 唐桂峰，李 宁，陈世福. 高速公路路面破损智能识别系统的设计与实现[J]. 计算机科学，2004，31：12－15.

[2] Quinlan J R. C4.5 Programs for Machine Learning[M]. San Mateo：Morgan Kaufmann Publishers，Inc，1993.

[3] 陈文伟，黄金才. 数据挖掘技术[M]. 北京：北京工业大学出版社，2002.23－25.

[4] 唐海兵，秦怀青. 利用决策树改进基于特征的入侵检测系统[J]. 微机发展，2005，15(4)：102－105.

[5] 朱 明. 数据挖掘[M]. 合肥：中国科学技术大学出版社，2002.67－72.

(上接第 75 页)

表 1 列出了在高精度定时器内核上和普通 2.6.10 内核上睡眠不同时间(从 100ns 到 4s)的平均延迟，可以看出平均延迟时间有效地从毫秒级减小到 10 微秒级。

表 1 平均睡眠延迟对比

高精度定时器内核(精度＝10 微秒)	普通内核(精度＝1 毫秒＝1000 微秒)
24934 纳秒	1996892 纳秒

图 2、图 3 分别绘制了在普通内核和高精度定时器内核上睡眠 50μs10000 次所测得的实际睡眠时间。与表 1 结果相似，50μs 的实际睡眠时间从(2.008～2.112)ms 降低到(67～83)μs。

4 结 论

高精度定时器用(jiffies＋sub_jiffie)表示定时器超时值，用本地 APIC 或 PIT 作为高精度定时器中断源，有效地提高了内核动态定时器的精度，加强了 Linux 在电信级应用的软实时能力，符合 CGL3.0[3]的要求，有利于将电信应用平台向 Linux 移植。

参考文献：

[1] Love R. Linux Kernel Development[M]. USA：Sams Publishing，2004.

[2] Bovet D P，Cesati M. Understanding the Linux Kernel(2nd edition)[M]. [s.l.]：O'Reilly Publishing，2002.

[3] Abeni L，Goel A，Krasic C，et al. A Measurement－Based Analysis of the Real－Time Performance of Linux[A]. Proceedings of the Eighth IEEE Real－Time and Embedded Technology and Applications Symposium (RTAS'02)[C]. Washington，DC，USA：IEEE Computer Society，2002.133－142.

[4] IEEE Std 1003.1－2001，System Interfaces，Issue 6 － for descriptions of interfaces of POSIX clocks and timers[EB/OL]. http://standards. ieee. org/catalog/olis/posix. html.，2001.

[5] Open Source Development Labs. Carrier grade Linux requirements definition (Version 3.0)[EB/OL]. http://www.osdl.org/docs/cgl-perf-req-def-30.pdf，2005.