基于文本挖掘的 PMI 日志分析研究

刘永富，陈永生

（同济大学 电子与信息工程学院，上海 200331）

摘 要：针对城轨交通的设备维护及检修，利用目前新兴的数据挖掘技术对日志进行关联分析，以期达到事故防患于未然的目的。介绍了关联规则的设计、算法的演示以及挖掘过程，并提出了一种改进算法，最后针对 PMI 日志做了相关分析研究。

关键词：文本挖掘；关联规则；Apriori PMI

中图分类号：TP311 **文献标识码**：A **文章编号**：1674-7720(2011)21-0083-03

Study on PMI log analysis based on text-mining

Liu Yongfu, Chen Yongsheng

(College of Electronic and Information Engineering, Tongji University, Shanghai 200331, China)

Abstract: With the increasing reliance on rail transport, the related equipment maintenance and repairing appears to be particularly important. This article is centered on the theme of this. While taking advantage of emerging data mining techniques to analysis them with association, to achieve the purpose of preventive measures. This paper briefly describes the design of association rules, algorithm demonstration, data mining process, and proposes an improved algorithm. And finally, made a analysis for the PMI log.

Key words: text-mining; association; Apriori PMI

当今社会飞速发展，城市轨道交通已经成为都市生活密不可分的部分。作为一种安全、可靠、清洁、快捷的公共交通运输方式，其为解决我国人口密集的大城市日益严重的交通拥挤问题的重要手段之一。至 2010 年底，上海已经投入运营 12 条线。其中新投入的 6、7、8、9、11 号线均采用基于通信的信号系统(CBTC)，如此便会产生大量的日志数据。为了使维护人员能从维修层面，根据日志中所记载错误，快速对系统故障定位，继而对设备进行相关修理，本文利用数据挖掘技术对日志进行关联分析并根据关联法则，增加错误诊断准确性，以防微杜渐，避免酿成事故。

1 关联算法设计

日志分析系统的总体结构设计思路：以日志资源库为基石，以分析平台为运行支撑，以关联分析功能为中心，建立系统架构，并在此系统架构的支持下开发软件。

关于关联算法的设计，选择目前比较成熟的 Apriori 算法并加以改进。

1.1 Apriori 算法介绍

Apriori 算法是一种挖掘布尔关联规则频繁项集的算法。它利用频繁项集性质，用逐层搜索的迭代方法来找出所有的频繁项集。首先，找出频繁 1-项集的集合，该集合记作 L_1。L_1 用于找频繁 2-项集的集合 L_2，而 L_2 用于找 L_3，如此下去，直到不能找到频繁 k-项集为止。在第 k 次循环中，先产生候选 k 项集的集合 C_k，C_k 的项集是用来产生频繁项集的候选集。C_k 中的每个元素在数据库中根据支持度计数进行验证，决定是否加入 L_k[1-2]。

设有初始项目集 L，包含 Z_k $(k=1\sim t)$，$Z_k=(z_1, z_2, z_3, \cdots\cdots)$，标志 TID$(1\sim t)$，设 min_sup$=a\%$，(即出现次数不少于 b 次，$b=a\%\times t$)。设置 K 为项目的元素数 (举例见表 1)，算法如下：

(1)重建事务集数据库：搜索包含 z_1(单个 item)的出现次数，生成 u_1；搜索包含 z_2 的出现次数，生成 u_2；以此类推，生成 u_n。

(2) 在 U 中去掉出现次数少于 b 的项，简化数据库(存在临时表中)。

(3)计算 $u_1 \cup u_2$，如交集中项数少于 b 则舍去，大于 b 则储存为新的集合。同理计算 $u_1 \cup u_3, u_1 \cup u_4, \cdots, u_1 \cup u_n, u_2 \cup u_3, \cdots, u_{n-1} \cup u_n$。新生成的频繁 2 项集合表示为 L_n。

(4)计算 $L_1 \cup L_2$。看 L_2 中第一个 item 是否存在于 L_1 中:若存在,忽略;若不存在,则将此 item 与 L_1 组成三项集,如交集中项数少于 b 则舍去,大于 b 则储存,并储存与之相应的项目集。同理类似处理 $L_1 \cup L_3, L_1 \cup L_4, \cdots\cdots L_1 \cup L_n, L_2 \cup L_3 \cdots\cdots L_{n-1} \cup L_n$。整理结果,去掉重复的项目集,得到频繁三项集。

(5)重复上述步骤,得到频繁 k 项集。

1.2 Apriori 算法改进

在求解频繁 $k+1$ 项集的时候,若其不在长度为 k 的频繁项集之间时,则必然不在长度为 $k+1$-频繁项集之中,而任意一个 k-项集的支持度与规模小于它的事务无关,故可以直接舍去,从而减少扫描的数据量。因此可以在遍历事务集时先遍历长度为 k 的频繁项集,若存在,再遍历事务表。在随后的过程中,及时删除其中不可能出现在候选项集中的记录,即字段长度不大于将要生成的 k-频繁项集 k 值,而且也不被包含在频繁项集中的记录。

改进算法通过先遍历长度为 k 的频繁项集,减少访问事务表中的无效记录,从而使访问次数减少而提高了运行效率。

1.3 算法示例

(1)初始事务表,取 min_sup=50%(=2),初始项目集表如表 1 所示。

(2)项目集 Z5={1,2,3,4,5}经过变换,如表 2 所示。项目 4 去掉。表 2 中为新得到的频繁一项集,将这些数据插入到临时表 Temp 中。

表 1 初始项目集表

TID	项目
1	1 3 4
2	2 3 5
3	1 2 3 5
4	2 5

表 2 Z5={1,2,3,4,5}经过变换

项目	Sup	K
1	2	1
2	3	1
3	3	1
4	1	1
5	3	1

(3)从 Temp 表中得到 K 值为 1 的事务集,根据算法叙述生成二项集,如表 3 所示。

根据支持度值,将事务集 1 2、1 5 去掉,得到频繁二项集,将这些数据插入到临时表 Temp 中。

表 3 二项集

项目集	Sup	K
1 2	1	2
1 3	2	2
1 5	1	2
2 3	2	2
2 5	3	2
3 5	3	2

(4)求出频繁三项集

从 Temp 表中得到 K 值为 2 的事务集,根据算法叙述生成三项集,如表 4 所示。

表 4 三项集

项目集	Sup	K
1,2,3	0	3
1,3,5	1	3
2,3,5	2	3

根据支持度值,将事务集 1 2 3、1 3 5 去掉得到频繁三项集,将这些数据插入到临时表 Temp 中。

(5)使用类似的方法可以求出频繁 K 项集。

2 PMI 日志分析

2.1 PMI 简介

Alcatel-Thales CBTC 信号系统中实现联锁功能的设备称为 PMI,完整 CBTC 调试完成后 PMI 将属于 ZC(区域控制器)的一部分,仍然实现联锁功能。主要组成:计算机联锁模块(MEI),看门狗机架(CDG),通信模块(SCOM)和维护辅助系统(SAM)。

ZC-PMI 联锁结构如图 1 所示。

图 1 ZC-PMI 联锁结构

PMI 接收到 ATS 的进路请求/进路取消命令,排列/取消进路。PMI 会按进路定义中的顺序预留进路元素。如果收到的进路有区域重叠,PMI 按接收的顺序排列重叠区域的进路。一旦进路使用完毕,PMI 就会逐段解锁进路。PMI 不会激活敌对进路(反向进路、交叉进路)。数据库中含有一个敌对进路表,PMI 通过检查该表来确保从 ATS“进路请求”命令中接收到的进路,不会与现有的已排列进路相冲突。PMI 通过 ATS 的“道岔动作”或“进路请求”命令接收道岔动作请求。如果道岔已经移动,那么 PMI 就拒绝 ATS 的道岔动作请求。PMI 会评估道岔动作规则从而决定请求的道岔是否可以动作。如果道岔是故障的,那么 PMI 会拒绝道岔动作请求[3]。

2.2 PMI 日志

PMI 采用 2*2 取 1 的工作方式,如此每天便会产生大量的日志记录,其中包括正常工作信息以及出错信息。现主要对报错、警告信息进行相关分析研究。PMI 日志格式如图 2 所示。

(1)日期时间:该记录产生时间;

(2)行号:该记录唯一标识;

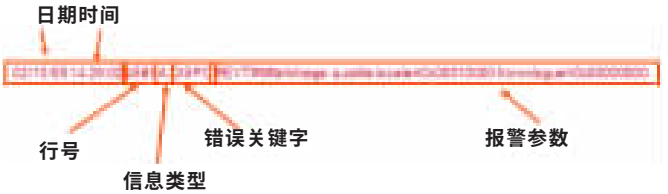

图 2 PMI 日志格式

(3)信息类型:A(报警或意见)、C(故障诊断或指示器)、E(消息发送)、P(状态图转变)等;

(4)错误关键字:表示该记录具体错误,例如:TAZ、COMP、DISPO 等;

(5)报警参数:表示该记录产生的具体位置[2]。

2.3 事务集提取及关联分析

将系统重启一次表示为一个事务,日志中会记录该次重启的原因。重启可能有多个错误,这就有必要分析其关联性,从而做到防微杜渐。

分析步骤如下:

(1)建立如表 5 所示的 ItemDB 表,用于存储事务集;

表 5 ItemDB 表结构

字段名	类型	长度	是否主键	是否为空	字段描述
ID	int	4	Yes	No	主键,自增
ITEM	nvarchar	4 000	No	No	一次重启所对应的所有错误
TIME	DATE		No	No	重启时间

(2)建立如表 6 所示的 Temp 表,用于存储频繁项目集;

(3)遍历日志,根据错误关键字进行文字匹配,将所有错误存入到 ItemDB 中作为事务集;

(4)利用关联规则对事务集进行分析,得出错误之间的关联性;

表 6 Temp 表结构

字段名	类型	长度	是否主键	是否为空	字段描述
ITEM	nvarchar	4 000	NO	NO	频繁 K 项集的具体项
SUP	int	4	NO	NO	改记录的支持度
K	Int	4	NO	NO	K

(5)分析关联性,得出终结。

该系统主要完成了对 PMI 日志的研究,同时利用关联规则进行了相关分析。借于此,对地铁维护、检修人员有如下好处:(1)快速对系统故障定位,修理相关设备;(2)增加错误诊断准确性,减少没有必要的排查;(3)若错误 A 与错误 B 关联,则 A 发生的情况下,须预防 B 发生,做到防微杜渐,避免酿成事故。

参考文献

[1] 朱辉生.关联规则挖掘的两种改进算法[J].计算机应用与软件,2006,23(8):117-119.

[2] 毛国君,段立娟.数据挖掘原理与算法[M].北京:清华大学出版社,2005.

[3] 曹锦磊.简析地铁 PMI 系统[EB/OL].2009-8.

[4] 法国 THALES 集团.PMI 维护与安装技术文档[Z].2010.

[5] Judith Bishop. C# 3.0 Design Patterns [M]. O'Reilly Media, Inc.2007.

(收稿日期:2011-06-13)

作者简介:

刘永富,男,1986 年生,硕士研究生,主要研究方向:计算机仿真。

陈永生,1966 年生,博导,主要研究方向:列车控制系统以及模式识别。