基于简单规则的数据质量检查系统设计与应用

高 科[1],刁兴春[2],曹建军[2]
(1. 解放军理工大学 指挥信息系统学院,江苏 南京 210007;
2. 总参第六十三研究所,江苏 南京 210007)

摘 要:为了更加全面地对数据存在的质量问题进行检查,并找出其中的问题数据,分析了数据质量评估的一般性指标,从规则约束的角度对关系型数据字段的格式、语法、长度、取值范围,以及字段与字段之间的逻辑关系、函数依赖关系等进行分类描述,设计相应的数据质量检查算法并进行编码实现,形成一套完整的数据质量检查工具。对某单位的设备人员信息数据从完整性、规范性、一致性、有效性等方面进行检查。实验结果表明,这些规则能够有效检出关系型数据中存在的问题。

关键词:数据质量;评估指标;规则;关系型数据
中图分类号:TP311 **文献标识码**:A **文章编号**:1673-629X(2015)06-0176-05
doi:10.3969/j.issn.1673-629X.2015.06.039

Design and Application of Data Quality Detection System Based on Simple Rules

GAO Ke[1], DIAO Xing-chun[2], CAO Jian-jun[2]
(1. College of Command Information Systems, PLA University of Science and Technology, Nanjing 210007, China;
2. The 63rd Research Institute of PLA General Staff Headquarters, Nanjing 210007, China)

Abstract: In order to carry out the overall detection of data quality and locate the incorrect data, analyze the general indexes of data quality assessment. The rules of structured data, which takes the form, grammar, length, range of single field and the logical, functional dependency relationship of different fields into account, was classified and programmed by using corresponding algorithm. Design a corresponding data quality inspection tools and realize it by programming, forming a set of whole data quality inspection tool. Test on the data of facility and staff information from the angle of integrity, normalization, consistence, effectiveness. The result of experiment shows that such data rules can find out the errors of the data.

Key words: data quality; index assessment; rules; structured data

1 概 述

数据是信息的载体,数据质量的好坏对于其能否正确反映客观世界以及有效支持决策具有重要意义。对于数据质量的定义,文献[1]将其定义为数据适合使用的程度(Fitness for use),具体到某个系统,文献[2-3]将数据质量定义为该系统在多大程度上实现了模式和数据实例的一致性,以及模式和数据实例在多大程度上实现了正确性、一致性、完整性和最小性。

文献[4-5]总结了在质、量、形、时四个方面进行数据质量评估的一般性指标,包括精确性、完整性、一致性、有效性、唯一性、时效性等。同时选取完整性和有效性两个指标给出了一个六元组的数据质量检查评估模型。

数据规则,又称数据约束,是客观世界的数据所应遵循的语义限制,包括领域知识和业务规则[6]。自 1971 年美国 IBM 公司研究员 Codd E. F. 提出关系数据模型以来,这种规则约束便成了研究热点。评价数据是否满足一般性指标就是检查数据是否满足质、量、

收稿日期:2014-07-11 **修回日期**:2014-10-16 **网络出版时间**:2015-05-06
基金项目:国家自然科学基金资助项目(61371196);中国博士后科学基金特别资助项目(201003797)
作者简介:高 科(1989-),男,硕士研究生,研究方向为数据工程;刁兴春,研究员,博士生导师,研究方向为网络及信息技术;曹建军,通信作者,博士后,研究方向为数据质量、进化计算。
网络出版地址:http://www.cnki.net/kcms/detail/61.1450.TP.20150506.1627.014.html

形、时方面的具体规则。

文献[7-8]从完整性、准确性、一致性、时效性四个维度定义了数据质量的约束,给出了基于约束的数据质量评估算法。

文中在上述研究的基础上,着重对各维度下的数据质量规则进行细化,从单字段的格式、语法、长度、范围要求以及字段之间的逻辑关系和函数依赖关系等角度进行规则描述和算法设计[9],并开发出数据质量检查工具。结合某单位现有的设备和人员信息数据中,由于表结构设计不合理以及人员录入过程中出现失误导致的数据不完整、重复、格式逻辑错误等问题,应用检查工具进行检查,很好地发现了其中存在的数据质量问题。

2　数据质量的规则分类描述

数据在数据库中以记录形式存在,具有一定的结构和格式规范[10-11]。分析研究这些结构化的数据记录,结合各维度指标的评价要求,将数据规则作如下分类描述。

2.1　单字段规则

作为数据库中的最小组成单元,在进行数据表的创建时,会对每个字段进行取值类型、长度、是否为空或主键进行定义。这些最基本的定义在形态上有效约束了字段取值。

2.1.1　不完整字段检查

字段不完整分为未赋值和未定义两种情况。未赋值对应字段内容为空(即空字符),而未定义表示该字段值未知(即 NULL)。在数据库中,空字符实际上是有效字符,两个空字段值相等;而 NULL 则是一类特殊值,不同于零长度字符串。如果列定义中包含 NOT NULL 子句,则不能为该行插入含有 NULL 值的行。如果列定义中仅包含 NULL 关键字,则接受 NULL 值。通常,上述情况都作为记录不完整进行处理。

文献[12]对其有关分类和检测方法进行了详细介绍。通过定义记录的二进制表示,根据不完整记录样本生成各类记录的标准二进制表示集。通过位运算实现了记录的分类检测,根据不完整记录二进制表示确定记录的进一步处理。

2.1.2　完整字段检查

1)特殊格式。

字段值的格式一般由该字段所表示的内容含义决定,某些特有的数据类型在数据库存储时有其自己特有的格式。

(1)时间日期格式。

时间日期的格式表示种类多样,通常需要根据实际的需求选取最适合的表达格式。例如,不同精度下的时间日期可以表示成:

· 2013-10-13(年-月-日)

· 2013-10-13-23(年-月-日-时)

· 2013-10-13-23-59(年-月-日-时-分)

· 2013-10-13-23-59-00(年-月-日-时-分-秒)

同样的精度下,间隔符不同时也会产生不同的表达格式,例如:

· 20131013

· 2013-10-13

· 2013:10:13

以精度为年月日格式检查为例,算法描述如下:

```
valueInterface V=(valueInterface)list.get(0);
DATE=V.getString();
int type=3;
void dateCheck(string DATE, int type)
{
string[] dat=DATE.split("-");
if(dat.length()==type) return true;
else return false;
}
```

(2)身份证格式。

二代身份证的号码按照国家的标准编制,长度为 18 位,分别由 18 个数字或者 17 个数字加上字母 X 组成。其中,每个数字位表示的含义如下:$a_1 \sim a_6$ 位是行政区划代码;$a_7 \sim a_{14}$ 位是出生日期代码;$a_{15} \sim a_{17}$ 位是顺序码;第 a_{18} 位是校验码。其中校验码是由号码编制单位按公式(1)和表 1 计算出来的。

$$S=\sum(a_i * w_i)(\mathrm{mod}11)(i=1,2,\cdots,17) \quad (1)$$

表 1　S 与 a_{18} 对应表

S	0	1	2	3	4	5	6	7	8	9	10
a_{18}	1	0	X	9	8	7	6	5	4	3	2

式中,a_i 代表身份证各位上的数字;w_i 代表各位上的权重,分别为:7、9、10、5、8、4、2、16、3、7、9、10、5、8、4、2、1;S 代表计算结果。根据表 1 查找 S 所对应的校验码 a_{18} 的值,判断身份证是否合法。算法描述如下:

```
valueInterface V=(valueInterface)list.get(0);
SFZH=V. getString();
void sfzhCheck(string SFZH)
{
int w[18];
char a[18];
for(int i = 18;i > 0;i--)
{
a[i] =V.toCharArray();//字符串转存为字符
}
```

```
if((∑(Integer.parseInt(a[i]) * w[i]))mod11) = =2
&&a[18] = = 'X')  return true;
else if((∑(Integer.parseInt(a[i]) * w[i]))mod11) +
Integer.parseInt(a[18]))%11 = =1)
return true;
else return false;
}
```

(3)数值与字符串格式。

数值格式的不同主要分为整数格式和浮点数格式。整数包括正整数、负整数和零,浮点数是指带有有限位小数的有理数。计算机中使用有限的连续字节保存浮点数,Java 平台上的浮点数类型 float 和 double 采纳了 IEEE754 标准中所定义的单精度 32 位浮点数和双精度 64 位浮点数的格式。Java 自带的 getMetaData()接口中,getColumnTypeName()方法可以直接获取字段的类型名。算法描述如下:

```
ResultSetMetaData data = rs.getMetaData();
columnTypeName[i] = data.getColumnTypeName(i);
//获取指定字段的类型名
if(columnTypeName[i].equals("指定格式"))return true;
else return false;
```

2)语法要求。

语法要求是指在数据表中,某些字段的取值遵循一定的语法格式。例如学生信息表中的学号字段(XH),要求取值由开头字母加上 9 位数字组成。且硕士研究生学号以字母"S"开头。以学号语法检查为例,算法描述如下:

```
valueInterface V = (valueInterface)list.get(0);
XH = V.getString();
char C = 'S';
void grammaCheck(string XH, char C)
{
char X = XH.charAt(0);
if(X.equals(C))return true;
else return false;
}
```

3)字段长度要求。

字段长度要求分为定长和不定长两种类型。所谓定长即字段值长度唯一,例如学生信息表中的身份证号(SFZH)字段,要求所有取值长度为 18。不定长的字段长度设定在用户需求的范围之内,一方面长度最大值不超过建表时定义的字段值长度上限;另一方面对于某些特殊含义的字段,例如学生信息表中的姓名(XM)字段,当长度小于 2 时即视为无效值。以身份证号字段长度检查为例,算法描述如下:

```
valueInterface V = (valueInterface)list.get(0);
//获取字段值
SFZH = V.getString();//转换成字符串型
int length = 18;
void lengthCheck(string SFZH, int length)
{
if(SFZH.length() = =length)  return true;
else return false;
}
```

4)范围限制。

一些数据因为受到其类型的限制,取值范围固定。例如某学校的学生基本信息表中,学生性别字段(SEX)取值范围只能为集合{男,女}中的其中一个;学生年龄字段(AGE)取值范围通常在 18 至 26 之间。两种情况算法描述分别如下:

(1)取值范围参数为有限集。

```
valueInterface V = (valueInterface)list.get(0);
SEX = V.getString();
string range[] = {"男","女"}
void rangeCheck1(string SEX, string range[])
{
if(SEX = ="男"||SEX = ="女") return true;
else return false;
}
```

(2)取值范围参数为区间。

```
valueInterface W = (valueInterface)list.get(1);
AGE = W.getInt();
int left = 18; int right = 26;
void rangeCheck2(string XH, int left, int right)
{
if(AGE>=left&& AGE<=right)return true;
else return false;
}
```

2.2 字段间关联关系规则

2.2.1 逻辑关系

逻辑关系是指同一张表的不同字段取值之间存在由常识或领域知识确定的约束关系。例如学生信息表中的入团时间(RTSJ)和入党时间(RDSJ)字段,应当满足入团时间字段取值在入党时间字段之前。算法描述如下:

```
valueInterface V = (valueInterface)list.get(0);
valueInterface W = (valueInterface)list.get(1);
RTSJ = V.getInt();
RDSJ = W.getInt();
int logicalNum = 1;(每种逻辑设定相应编号)
void logicalCheck(int RTSJ, int RDSJ, int logicalNum)
{
if(RTSJ<RDSJ)return true;
else return false;
}
```

2.2.2 函数依赖关系

函数依赖是指在同一张数据表中的不同字段之间

取值所具有的函数关系,这种函数关系使得彼此的取值相互制约。例如学生信息表中,某一门课程的合格率(HGL)取值应当是合格人数(HGRS)与参考人数(CKRS)的比值。算法描述如下:

```
valueInterface U=(valueInterface)list.get(0);
valueInterface V=(valueInterface)list.get(1);
valueInterface W=(valueInterface)list.get(2);
HGL=U.getInt();
HGRS=V.getInt();
CKRS=W.getInt();
int DependencyNum=1;(每种依赖设定相应编号)
void fucDependencyCheck(string HGL, string HGRS, string CKRS,int DependencyNum)
{
if(HGL==HGRS/CKRS)return true;
else return false;
}
```

2.3　多规则逻辑组合

对于比较复杂的数据检查,单一的检查规则不能满足检查要求,可以通过设置多条检查规则,选取适当的逻辑组合方法。例如对人员身份证号进行检查时,可以同时设置字段长度是否为 18 以及格式规范两项检查规则,然后用逻辑连接词“AND”进行联合。算法描述如下:

```
if(sfzhCheck(SFZH)&&lengthCheck(SFZH,18))
return true;
else return false;
```

2.4　专用算法

2.4.1　相似重复记录检测

相似重复记录的检查过程比较复杂。其针对的是一类特殊的“问题”数据,这些数据在“形”上面可能完全符合要求,但是在语义方面因为存在较高的冗余使得数据失去了自身价值而需要进一步处理。

文献[13]针对不同字段类型的相似度检测方法进行了详细的分类描述。将相似重复记录检测看成二分类问题,定义了字符串型、枚举型和日期型三种典型属性类型的相似特征和归一化算法。建立了特征选择模型,利用蚁群算法进行实现并验证了该方法的有效性。

2.4.2　经纬度检查

地理信息数据是一类比较特殊的数据,地理位置信息通过经纬度表示。数据记录至少包括地名代码、地名、经度值、纬度值等字段信息,同时需要有一张标准的行政区划信息表作为检查参照。当经纬度偏差超过设定阈值时,则认为是问题数据。

算法描述如下:

```
valueInterface U=(valueInterface)list.get(0);
addID=U.getInt();//获得检查对象代码
find(int addID);//标准表中寻找经纬度信息
computeDistance();//计算记录值与标准值之间距离
compare();
//比较偏差与阈值关系,返回值为 true 或者 false
```

3　系统设计与实现

通过对上述规则进行整理归纳,设计合理的算法,利用 Java 语言开发环境编程实现,开发出一套数据质量检查工具[14]。基于规则的数据质量检查工具的系统架构如图 1 所示。

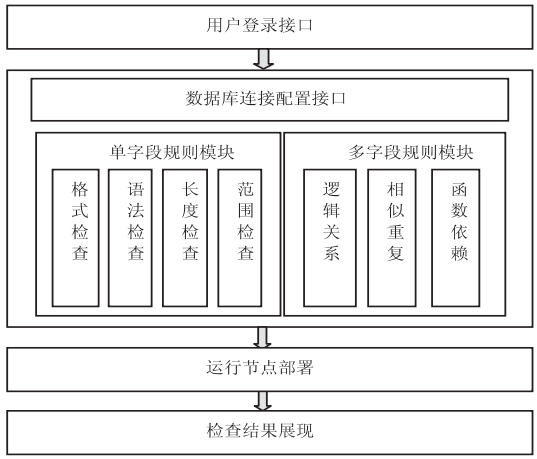

图 1　数据质量检查软件系统架构图

系统由用户登录接口、数据质量检查配置模块、运行节点以及浏览器显示模块构成。在数据质量检查配置模块中,包括基本的数据库连接配置接口以及可支持二次开发的字段规则检查模块。系统支持 MySQL、Oracle 等数据库连接。在运行节点部署部分,可以根据检查任务工作量的大小选择一个或多个节点进行处理。任务执行完毕后,用户可以在浏览器中进行检查结果的查看。

软件在对本地数据库中的目的表进行检查的同时,将结果以临时表的形式进行存储,同时通过浏览器展现检查结果。检查人员可直接对问题数据选择性进行修改并返回至源方。

4　实例分析

某单位每年都会对其人员和设备信息情况进行统计汇总,汇总数据存储在名为“设备动态情况”(t_SBDTQK)和“关键岗位人员”(t_GJGWRY)的数据表中。然而由于录入过程不规范、标准不严格以及人为失误等因素,使得每年的汇总数据都存在不少问题,严重影响了数据的整体质量。2013 年 11 月,笔者对当年汇总完毕的数据利用上述规则进行检查,实验过程和检查内容如图 2 所示。

为了进一步评估该单位的人员设备数据质量,笔者对检查发现的问题数据进行了统计,如表 2 所示,进

行观察分析。

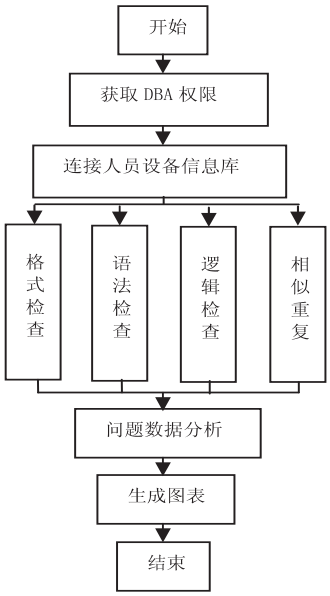

图 2 数据检查实验过程

表 2 问题数据统计情况记录

字段名	存在的问题	问题记录数	总数	百分比/%	影响维度
维修能力	空值	1 320	4 770	27.7	完整性
身份证号	长度不规范	9 838	32 180	33.7	规范性
正常存储最大存储	正常存储>最大存储	146	2 809	5.2	一致性
姓名	重复记录	36	32 180	0.1	有效性

通过实验记录发现,表 2 的问题数据统计情况记录中,该单位采集的人员和设备信息在完整性和规范性等方面存在较为严重的问题。

设备维修能力为空以及字段值不能传达任何有用的信息;身份证号码不满 18 位则可以认为是错误数据;仓库存储的逻辑错误以及人员信息重复记录需要进一步进行确认。

通过检查分析问题记录,发现该单位存在设备命名不规范、仓库使用不规范,人员调动较多且信息更新不及时等问题。同时为以后进行数据采集工作时应当注意的问题提供了有效参照。

5 结束语

随着信息技术的蓬勃发展,数据已经成为一种全新的资源,人们对于数据质量的要求也越来越高。然而,由于数据的海量以及种类繁多等特征,使得对于数据的统一管理缺乏有效的方法机制。

文中根据关系型数据之间存在的规则关系,对不同种类的规则进行归纳分析,并生成相应的算法应用到软件中对数据进行了检查,也得到了理想的实验结果。笔者可在今后工作中将规则进一步扩充完善,使检查更加全面。

参考文献:

[1] Huang K T, Lee Y W, Wang R Y. Quality information and knowledge management [M]. New Jersey: Prentice Hall, 1998.

[2] Aebi D, Perrochon L. Towards improving data quality [C]// Proc of the international conference on information systems and management of data. [s. l.]: [s. n.], 1993: 273-281.

[3] Improving data warehouse and business information quality: methods for reducing costs and increasing profits [M]. [s. l.]: John Wiley & Sons, 1999.

[4] McGilvray D. Executing data quality projects: ten steps to quality data and trusted information (TM) [M]. [s. l.]: Morgan Kaufmann, 2010.

[5] 韩京宇,徐立臻,董逸生. 数据质量研究综述[J]. 计算机科学,2008,35(2):1-5.

[6] 杨青云,赵培英,杨冬青,等. 数据质量评估方法研究[J]. 计算机工程与应用,2004,40(9):3-4.

[7] 丁海龙,徐宏炳. 数据质量分析及应用[J]. 计算机技术与发展,2007,17(3):236-238.

[8] 雷天武. 基于规则的数据质量管理系统架构与关键问题研究[D]. 济南:山东大学,2009.

[9] 宋 敏,覃 正. 国外数据质量管理研究综述[J]. 情报杂志,2007,26(2):7-9.

[10] 程录庆. 数据约束对数据质量的影响研究[J]. 长江大学学报:自然科学版,2011,8(5):100-102.

[11] 梁吉胜,李天阳,王惠霞,等. 基于约束的数据质量评估算法研究[J]. 科学技术与工程,2012,12(3):551-554.

[12] 曹建军,刁兴春,吴建明,等. 基于位运算的不完整记录分类检测方法[J]. 系统工程与电子技术,2010,32(11):2489-2492.

[13] 曹建军,刁兴春,杜 鹢,等. 基于蚁群特征选择的相似重复记录分类检测[J]. 兵工学报,2010,31(9):1222-1227.

[14] 房 强. 面向半结构化数据的数据质量控制系统的研究与实现[D]. 沈阳:东北大学,2008.

基于简单规则的数据质量检查系统设计与应用

作者： 高科， 刁兴春， 曹建军， GAO Ke， DIAO Xing-chun， CAO Jian-jun
作者单位： 高科,GAO Ke(解放军理工大学 指挥信息系统学院,江苏 南京,210007)， 刁兴春,曹建军,DIAO Xing-chun,CAO Jian-jun(总参第六十三研究所,江苏 南京,210007)

刊名： 计算机技术与发展 ISTIC
英文刊名： Computer Technology and Development
年，卷(期)： 2015(6)