数据仓库环境下以用户为中心的数据清洗过程模型*)

鲍玉斌　孙焕良　冷芳玲　王大玲　于　戈
（东北大学信息学院　沈阳110004）

摘　要　数据清洗是数据仓库和数据挖掘中非常重要的一个环节。本文首先分析总结了数据清洗的有关概念，给出了数据清洗中需要解决的质量问题，并总结了解决这些问题的技术和方法。在此基础上提出了以人为中心的数据清洗过程模型。该模型集成了工作流技术、数据集成、数据转换和数据挖掘技术。给出了每个工具箱应该提供的基本功能。

关键词　数据清洗，过程模型，数据仓库，数据挖掘，数据质量

A Human-Centered Process Model for Data Cleansing under Data Warehousing

BAO Yu-Bin　SUN Huan-Liang　LENG Fang-Ling　WANG Da-Ling　YU Ge
(College of Information, Northeast University, Shenyang 110004)

Abstract　Data cleansing is an important step both in data warehousing and data mining. This paper reviews some concepts on data cleansing, lists the data quality issues needed to be resolved in data cleansing process, and presents the techniques and methods for data cleansing firstly. Then a human-centered process model for data cleansing is proposed. It combines with workflow, data integration, data transformation, and data mining techniques. It also presents the main functions of each toolkits.

Keywords　Data cleansing, Process model, Data warehousing, Data mining, Data quality

1　引言

数据的抽取、转换、清洗和装入(Extracting, Transformation, Loading, ETL)是建立数据仓库系统的重要环节之一，在一个数据仓库项目中，约80%的工作量都花费ETL阶段。ETL所完成的工作，就是要从事务处理系统中，将数据迁移到数据仓库系统。由于不同的事务处理系统必将用到不同的数据库系统，包括不同的关系型数据库，非关系型数据库，甚至文件系统等，因此数据ETL工具就一定要能够访问多种数据源。有报告显示数据库中的错误超过10%，在某些应用领域更高[1]。显然，作为决策支持系统的数据提供者，数据仓库必须提供高质量的数据和服务。如果在数据仓库工程开始的时候，忽视或淡化已有数据中的质量问题，则必将导致数据仓据项目的最终失败[2]。Gartner Group的调查同样指出，导致项目超支失败的主要原因是数据质量问题[3]。如果不采取严格的质量控制策略，字段错误率在5%左右[4]。如果业务人员发现数据仓库或数据集市中包含错误的数据，或者利用数据仓库或数据集市中的数据分析出的结论是错误的，业务人员就不信任数据仓库，进而放弃使用数据仓库。

数据仓库环境需要提供功能强大的数据清洗能力，因为数据仓库需要从各种不同的数据源连续不断地收集数据，而其中的某些数据源(高噪声环境下的数据或含有自由格式的数据)包含脏数据的可能性很高。另外数据仓库是用于决策支持，因此数据的准确性至关重要，否则可能得出错误的结论。例如，重复的或遗失数据将产生不正确的统计结果。由此，可以看出数据清洗是数据仓库项目中必不可少的工作。

数据清洗的目的就是检测和删除数据中的错误和不一致，以便提高数据的质量。数据质量问题是单个数据集中的问题，如数据项拼写错误、遗失数据、重复记录等。在联邦数据库、数据仓库或基于Web的全局信息系统等环境下，多数据源需要集成。因为多个数据源在数据表示等方面存在差异，集成后的数据存在冗余(如重复元组)，所以需要数据清洗，以便得到正确的、一致的、表示形式统一的并且消除重复的数据。

但是，由于存在各种可能的数据不一致，以及数据量的巨大和使用的技术众多等原因，使得数据清洗成为数据仓库建设中的一个很关键的繁琐问题。关于数据清洗中某些具体问题的解决，如重复元组的检测[5]、错误的自动发现[6]、异常值检测[7]等，都提出了许多方法，也有许多公司开发了针对具体应用的数据清洗工具，文[8]中给出了各种工具的综述。然而，只有很少的文献讨论数据清洗的过程模型。文[6]给出了一个数据清洗工具框架，其中提供了有关的统计分析和数据挖掘等分析工具。文[9]利用模式匹配技术发现数据中的错误，设计了数据转换函数，实现数据解析和属性复制、删除、分裂、合并等转换。其中提供了数据转换引擎，在数据转换的基础上，进行增量式错误检测。本文借鉴了文[10]中以人为中心的数据挖掘处理过程模型的思想，将其用于数据仓库的数据清洗过程中，并将数据挖掘技术、工作流技术、数据转换和模式集成技术集成于一体。

2　数据清洗定义

数据清洗的目的就是检测和删除数据中的错误和不一致，以便提高数据质量[11]。数据清洗是较新的研究领域，对大数据集的清洗是很费时的工作。数据清洗可以作为数据仓库、数据挖掘和数据/信息质量管理的重要步骤。然而，关于数据清洗没有公认的定义，不同的应用领域有不同的定义。在数据仓库领域中，当几个数据库合并的时候使用数据清洗，例如表

*)本文的工作得到国家自然科学基金项目资助(项目编号:60173051)。

示相同实体的记录具有不同的表示格式，这就产生了重复元组，这就是重复元组检测和清除问题，称之为合并/纯净问题[12]，也称为实例辨识或对象辨识问题。因此，在数据仓库领域，数据清洗定义为清除错误和不一致数据的过程，并且解决元组重复问题。在数据挖掘的处理过程中，数据清洗是第一个步骤[10]。在数据挖掘的文献中称数据清洗是指用计算机实现数据的检查、遗失值和不正确数据检测和校正数据的过程[13]。

联邦数据库和基于 Web 的信息系统需要解决与数据仓库中相同的数据转换问题。其中每个数据源有一个包装器(Wrapper)用于抽取数据和一个集成器(Mediator)用于集成数据[14,15]。它们仅提供有限的数据清洗功能，主要是进行数据转换和模式集成。

与数据清洗密切相关的是数据质量问题。文[16]中提出了数据质量的4个维：准确性、及时性、完整性和一致性。而数据的正确性是由上述4个方面定义的。数据清洗过程就是评价数据的正确性并提高数据的质量。

3 主要数据质量问题及解决方法

数据清洗的目标是为了清除数据中的错误，提高数据的质量。因此，首先需要了解数据中存在哪些质量问题。对于不同的质量问题有不同的解决方法。根据数据质量的4个重要方面，将数据质量问题分为：

·不完整数据。如缺失记录和字段，或者设计中有的记录和字段没有被记录；

·不正确数据。例如错误代码(如将沈阳的区号写为021)、错误的聚合运算、重复元组等；

·不可理解数据。例如，一个字段中包含多种信息，或者为节省磁盘空间而使用怪异的数据格式，或者未知的代码等；

·不一致数据。例如，不同代码的不一致使用，一个代码有不一致的含义，相同的意义不同的代码，不一致的名字和地址等；

·模式冲突。模式冲突包括命名冲突和结构冲突。命名冲突包括同名异义和异名同义；结构冲突包括类型冲突、依赖冲突、关键字冲突和行为冲突等。

上述质量问题，有的可以使用工具辅助进行辨识和校正，有的只能靠手工处理，有的是可以使用工具检测出来，但是需要交互式地进行校正或处理。对于遗失值、异常值、不一致代码、模式集成、重复元组检测和拼写错误等可以使用工具处理。

对于遗失数据可以通过属性之间的依赖关系预测属性遗失值。例如，可以利用属性依赖分析发现属性之间的依赖关系，或者通过数据挖掘的方法发现数据之间满足某种模式，根据这种模式或依赖关系预测元组中相应字段的值，进而替代遗失值。

数据中的异常(Outliers)数据包括字段值异常和记录异常两种。字段值异常可以使用统计学的方法计算某个字段值的平均值、标准差 σ、取值范围、空值出现的数量和频率、最大值、最小值等。根据这些统计值和相关的启发式规则可以发现数据中的异常。例如，对于在[均值－nσ，均值＋nσ]之外的取值，认为是异常，一般 n 取3到6的整数。或者位于字段置信区间之外的数据值，认为是异常值。可以使用重复元组检测技术检测异常的元组，还可以使用数据挖掘技术发现数据中的异常。例如，通过比率关联规则[17]发现数值属性间的比率关系，如果两个属性间有95%的元组满足某种比率关系，则剩余的5%就很可能是异常数据。

对于不一致的代码可以通过查表或哈希表的方法进行检测。一般一个字段的编码值是有限的，因此可以使用哈希表检测代码的不一致。例如建立邮编同地址的对应哈希表，用来检测地址和邮编的不一致问题。

对于模式冲突问题可以通过模式集成技术解决。模式集成需要针对不同的问题采取不同的集成策略，文[18]给出了模式集成的综述。

当多个数据进行合并时会产生重复元组问题。重复元组检测的方法很多，如排序近邻算法[19]、N-gram 滑动窗口算法[20]，领域无关的优先队列算法[21]等。

拼写错误可以使用 N-gram 滑动窗口算法，或者编辑距离等方法检测并校正。

4 数据清洗框架

数据仓库中 ETL 过程如图1所示。数据源的数据通过包装器和监视器抽取到数据仓库的数据预处理环境中。其中监视器是监视并收集数据源数据的变化，包装器是将数据从数据源抽取出来。数据清洗是对包装器和监视器抽取的数据进行集成、转换、清洗形成“清洁”的数据，供后续的数据聚合和过滤处理，形成符合数据仓库格式要求的数据。

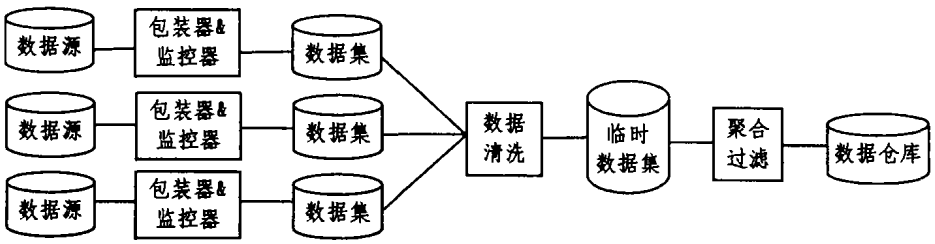

图1 数据仓库中的数据 ETL 过程

针对图1中的数据清洗阶段，本文提出如图2所示的以人为中心的数据清洗处理模型。其中，包括数据集成、转换、清洗、验证和报告5个处理过程，以及完整性分析、数据转换工具、统计分析工具、可视化工具和数据挖掘算法5个辅助支持工具。集成、转换和清洗这个过程是可以反复进行的。模型中涉及三种流，即数据流、元数据流和处理流。每一步处理都需要使用数据集的元数据信息，并且会产生新的技术元数据。下面对各个主要的部件进行详细讨论。

(1)集成

集成是使用模式集成方法将从多个数据源获取的数据合并和集成为单一数据集。集成主要解决模式冲突问题。进行数据集成时，根据设计的目标模式将源数据集中的不符合要求的设计进行转换。例如，量纲不统一的需要统一，类型不统一的进行类型统一等。当数据集中的属性数目很多时，完全靠手

工比对不一致的模式问题是很繁琐的工作，因此，在进行集成操作时需要借助有关的完整性分析、可视化分析和统计分析工具辅助发现源数据集中存在的模式不一致问题。比如通过同义词词典发现异名同义的属性，利用统计信息发现同名异义的情况等。

(2)转换

转换过程是将集成后的数据集转换成清洗所需的格式。这里所说的转换不是指进行量纲的统一变换，而是指对数据中的属性列和行进行有关的变换，以便于进行数据清洗。例如，对于自由格式的字段，如用户地址等，需要解析出其中的基本元素，然后再进行重复元组检测、异常检测等分析。转换过程需要借助数据转换工具和可视化分析工具的支持。

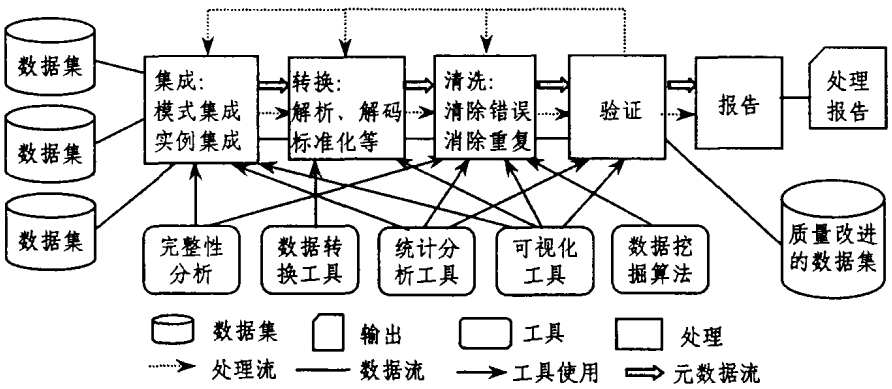

图2 以人为中心的数据清洗模型

(3)清洗

清洗是指消除重复元组、清除数据中的主要错误等。这是数据清洗过程的关键步骤。数据清洗首先需要发现可能的数据错误，这需要借助相关的分析工具和方法的支持，如完整性分析、统计分析、数据挖掘等；然后，用户使用可视化工具分析检测出的可能的错误；最后，根据不同方法检测出的不同类型的错误设置相应的规则清除或纠正错误。

发现数据中潜在错误的方法很多，如聚类分析、关联规则发现、统计分析或编辑距离等。

(4)验证

验证是通过数据质量评价措施，检测经过处理后的数据是否还存在主要的错误。有些错误是嵌套的，即隐藏在另外的一些错误的后面，当将显露在前端的错误清除后，隐藏在其中的错误才会出现。这也是数据清洗的一个难点所在，因此，在进行了一个循环的数据清理之后需要检查数据质量是否达到要求，即利用有关的统计分析工具、可视化分析工具和数据质量评价方法评价数据质量，如果满足了质量要求，则输出分析报告，生成处理脚本程序。否则，在此基础上，进行下一轮的清理工作直到满足要求为止。

(5)报告

报告是将经过验证可以清除数据中主要错误的数据处理方法和过程整理出来，以一种通用的描述语言(如扩展的SQL语言，或某种支持4GL语言的宿主语言)表示，以便真正执行数据清洗过程，尤其是在数据仓库环境中，数据的清洗工作是在数据仓库的初始加载和运行中的更新维护时执行的，因此需要将数据清洗过程程序化。

(6)完整性分析工具

可以使用完整性分析发现数据中的错误。对于给定的关系模式的数据集，完整性分析可以用于进行简单的数据清洗操作。关系数据完整性包括实体完整性、属性完整性和参照完整性。对于各种完整性可以定义相应的完整性规则，例如一个实体的标识符不能为空，并且必须唯一。因此，完整性分析针对关系数据库的完整性约束的类型，检查数据集的相应的部分，再根据完整性规则，确定是否违反完整性约束。

(7)数据转换工具

利用数据转换工具将集成后的数据集转换成便于清洗的格式。文[9]中将数据转换分成三种类型：单个数据值的转换、一行到一行的转换(称为垂直转换)和多行间的映射(称为水平转换)。具体地，其中提供了几个转换函数，包括格式化一列(将一个属性列转换成一种指定的格式，例如将英文名字格式化成名在前、姓在后，并且以空格分隔)、删除一列、复制一列、添加一列、合并列(将两列连接成一个属性列，例如将处理后的单个姓和名字字段合并成一个姓名字段)、拆分列(将一个属性列按照某种正则表达式或者按照某个位置拆分成两个列)、折叠列(是将一行转换成多行，例如原来的一行中包含了一个学生的多门课程的成绩，折叠后每行中只包含一门课程的成绩，这样将原来的一行变成多行)、展开列(与折叠列是互逆操作)等。

(8)统计分析工具

利用统计分析工具可以获得数据的描述信息，如可以获得某个字段的数据类型、长度、取值范围、不同取值数和各个取值出现的频率、唯一性、NULL值的出现次数、特殊的字符串模式、字段的平均值、标准差σ、最大值、最小值以及字段的置信区间等。对于统计分析获得的分析结果，利用领域知识建立的启发式规则可以发现数据中的错误。例如，If 基数(性别)>2 then error；If max(年龄)> 200 then error 等。

(9)可视化工具

可视化分析工具可以帮助数据质量分析人员分析数据的分布情况。另外，还可以利用它查看数据清洗过程每一步的结果，以便用户可以交互式地进行错误验证、确认和校正。根据数据源的数目、异构的程度和数据脏的程度，需要不同的数据转换和清洗方法，因此需要建立数据清洗的工作流和有关的转换清洗规则。所以需要可视化的数据清洗工作流建立工具。利用该工具可以可视化地建立数据清洗的流程，以交互式的方式选择所需的方法以及设定有关参数，并且可以利用流程调度和监视功能方便地执行流程，并查看每一阶段的结果。

(10)数据挖掘工具

统计分析工具只能发现有限的相对简单的数据错误，并且需要用户对分析结果进行再分析，才能发现某些错误。而利用数据挖掘工具可以自动发现数据中的潜在错误。目前有许

多利用数据挖掘发现数据中错误的方法，如聚类分析算法(如K-means、层次聚类等)、关联规则算法(如定量关联规则、比率关联规则和有序关联规则等)等。有些数据挖掘算法需要很长的时间，因此对于不同的应用场合，需要选择合适的算法。对于数据仓库数据源数据的清洗需要运行速度快的挖掘算法，而对于实时性要求较低的场合，如数据挖掘的数据清洗过程，可以选择运行时间相对较长的算法。

文[11]中提出数据仓库数据清洗方法应该满足：应该能够检测和删除所有主要的错误和不一致的情况，包括单数据源和多数据源集成的情况；应该有工具的支持，需要有限的手工查看和编程；应该很容易地扩展到其它数据源；数据清洗不应该孤立地进行，应该同数据转换结合起来，并利用元数据信息；数据清洗和数据转换的映射函数应该使用声明性语言描述；在数据仓库环境下，应该提供工作流支持，以便执行多数据源、大数据量的可靠有效的数据转换。

本文提出的模型提供了一个将数据集成、转换和清洗集成于一体的数据清洗框架，一方面指出了数据清洗的流程，另一方面给出了数据清洗需要的支持工具。该模型是以数据清洗的操作者为中心，交互式地进行工作流建模、自动地工作流调度和监视清洗工作。

结语 本文首先回顾了数据清洗的相关概念和技术。然后，提出了一个集成的以人为中心的数据清洗过程模型。该模型利用了工作流思想，将数据清洗的流程看作是一个工作流，流程中的每个任务或数据清洗的每一步都可以选择适当的技术和方法。并指出了数据清洗所需的工具集。该模型不仅适合于数据仓库环境下的数据清洗，也适用于数据挖掘过程的数据清洗。本文的进一步工作是实现数据清洗的可视化工作流定义工具、工作流调度和监视工具，设计实现各个工具箱，定义有关的数据清洗规则和策略。

参考文献

1 Wang R Y, Reddy M P, Kon H B. Towards quality data: an attribute-based approach. decision support systems, 1995, 13
2 Celko J, McDonald J. Don't warehouse dirty data. Datamation, 1995, 41(19): 42～45
3 Forino R, Data e. Quality: the data quality assessment, part 2. DM Review Online in September 2000. http://www.DMReview.com
4 Redman T. The impact of poor data quality on the typical enterprise. Communications of the ACM, 1998, 41(2): 79～82
5 Monge A E. Matching algorithm within a duplicate detection system. IEEE Techn. Bulletin Data Engineering, 2000, 23(4)
6 Marcus A, Maletic J I. Utilizing association rules for identification of possible errors in data sets: [The University of Mernphis' Technical Report CS-00-02]. 2000
7 Knorr E M, Ng R T. A unified notion of outliers. properties and computation. In: Proc. of KDD 97, 1997. 219～222
8 Orli R J. Data extraction, transformation, and migration tools part II. Available at: http://www.kismeta.com/ex2.html, 1996
9 Raman V, Hellerstein J M. Potter's Wheel: an interactive data cleaning system. In: Proc. of the 27th VLDB Conf. Roma, Italy, 2001
10 Brachman R J, Anand T. The process of knowledge discovery in databases: a human-centered approach. In Advances in Knowledge Discovery and Data Mining, MIT Press/AAAI Press, 1996
11 Rahm E, Do H H. Data cleansing: problems and current approaches. IEEE Techn. Bull. Data Engineering, 2000, 23(4)
12 Hernandez M A, Stolfo J S. Real-world data is dirty: data cleansing and the merge/purge problem. Journal of Data Mining and Knowledge Discovery, 1998, 2: 9～37
13 Simoudis E, Livezey B, Kerber R. Using recon for data cleaning. In: Proc. of KDD, 1995. 282～287
14 Tork R M, Schwarz P M. Don't scrap it, wrap it! a wrapper architecture for legacy data sources. In: Proc. 23rd VLDB, 1997
15 Wiederhold G. Mediators in the architecture of future information systems. IEEE Computer, 1992, 25(3): 38～49
16 Fox C, Levitin A, Redman T. The notion of data and its quality dimensions. Information Processing and Management, 1994, 30(1): 9～19
17 Korn F, Labrinidis A, Kotidis Y. Ratio Rules: A new paradigm for fast, quantifiable data mining. In: Proc. of the 24th VLDB Conf. New York, USA, 1998
18 Stephen H, Sudha R. Multiuser view integration (MUVIS): an expert system for view integration. ICDE, 1990. 402～409
19 Hernandez M, Stolfo S. The Merge/purge problem for large databases. In: Proc. of ACM-SIGMOD Conf., May 1995
20 Kukich K. Techniques for automatically correcting words in text. ACM Computing Surveys, 1992, 24(3): 377～439
21 Monge A E, Elkan C P. An efficient domain-independent algorithm for detecting approximately duplicate database records. Department of computer Science and Engineering, University of California, 1997

(上接第23页)

定是否部署这个应用。当一个 sar 文件放入 Scanner 的扫描目录以后，MainDeployer 会调用 ServiceDeployer 来部署这个应用。ServiceDeployer 根据 MainDeployer 传入的 DeploymentInfo创建一个子类 ServiceDeploymentInfo 的实例，以描述这个组件部署的信息。随后 ServiceController 实现服务的创建和控制，它的 ServiceCreator 属性创建服务，ServiceConfigurator属性进行服务创建后的配置。ServiceCreator 和 ServiceConfigurator 直接和 MbeanServer 进行交互，将待部署应用的服务动态地装载到 MbeanServer 提供的运行环境中。

结束语 WebFrame2.0是中科院软件所自主开发的遵循 J2EE 规范的 Web 应用服务器。基于本文的可扩展热部署模型，WebFrame 中实现了 J2EE 规范所规定的基本应用类型的热部署，增加了 Mbean 服务的热部署。此外，实现了远程部署的功能，将热部署系统设计为客户端和服务器端两部分，用户可以通过客户端将应用热部署到远端的服务器上。与传统的部署机制相比，本文提出的可扩展热部署机制具有如下特点：提高可用性，当部署一个组件时，系统不需要被中断；提高系统中组件的可维护性；增强 Web 应用服务器的可扩展性，有助于灵活的支持新的应用扩展；提高系统的灵活性，使可部署的服务富于变化。下一步的研究工作包括对模块结构的形式化检查，以减少动态装载时错误的发生。

参考文献

1 陈宁江，金蓓弘，范国闯. 多层企业应用的关键：J2EE Web 应用服务器. 计算机科学，2003，30(1)
2 Sun microsystems. Java (tm) 2 Platform Enterprise Edition Specification, v1.4, 2002.8
3 Gamma E, Helm R, Johnson R, Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software. Addison Wsely, 1994
4 Sun Microsystems. Java™ Management Extensions Instrumentation and Agent Specification, v1.0, 2000.6
5 Gosling J, et al. Java Language Specification, Second Version. Addison-Wesley Pub Co; Sep. 1996