International Journal of Academic Research and Development
ISSN: 2455-4197, Impact Factor: RJIF 5.22
www.academicsjournal.com
Volume 2; Issue 3; May 2017; Page No. 84-88

A novel approach of data cleaning/cleansing detecting, editing

Swati Kamble, Vaishali Kohle
D.Y. Patil College of Engineering Akurdi, Pune, Maharashtra, India

Abstract
We classify data quality problems that are addressed by data cleaning and provide an overview of the main solution approaches. Data cleaning is especially required when integrating heterogeneous data sources and should be addressed together with schema-related data transformations. We also discuss current tool support for data cleaning.

Keywords: data cleaning, heterogeneous data, schema-related data transformation, data warehousing, bi tool, data analysis tool, specialized cleaning tool

1. Introduction
Data cleaning, also called data cleansing or scrubbing, deals with detecting and removing errors and inconsistencies from data in order to improve the quality of data. Data quality problems are present in single data collections, such as files and databases, e.g. like spelling mistake, missing information or data, invalid data. Data cleaning intends to identify and correct these errors or at least to minimize their impact on study results. Data cleansing is hard to do, hard to maintain, hard to know where to start. There seem to always be errors, dupes, or format inconsistencies. One of the most challenging aspects of data cleansing has got to be maintaining a clean list of data, whether it's sourced from multiple vendors or manually entered by your hard-working interns, or a combination of both. One mistype could create a whole myriad of problems within your database, and can lead to hours upon hours of manual cleansing that could so easily have been avoided. So what is the solution to these frustrating, time consuming problems? Data scrubbing, also called data cleansing, is the process of amending or removing data in database that is incorrect, incomplete, improperly formatted, or duplicated. An organization in a data-intensive field like insurance, retailing, banking, tele-communications, or transportation might use a data scrubbing tool to systematically examine data for flaws by using rules, algorithms, and look-up tables. Data cleansing, data cleaning or data scrubbing is the process of detecting and correcting (or removing) corrupt or inaccurate records from a record set, table, or database. Used mainly in databases, the term refers to identifying incomplete, incorrect, inaccurate, irrelevant, etc. parts of the data and then replacing, modifying, or deleting this dirty data.

2. Process
2.1 Data auditing
The data is audited with the use of statistical and database methods to detect anomalies and contradictions: this eventually gives an indication of the characteristics of the anomalies and their locations. Several commercial software packages will let you specify constraints of various kinds (using a grammar that conforms to that of a standard programming language, e.g., JavaScript or Visual Basic) and then generate code that checks the data for violation of these constraints. This process is referred to below in the bullets "workflow specification" and "workflow execution." For users who lack access to high-end cleansing software, Microcomputer database packages such as Microsoft Access or File Maker Pro will also let you perform such checks, on a constraint-by-constraint basis, interactively with little or no programming required in many cases.

2.2 Workflow specification
The detection and removal of anomalies is performed by a sequence of operations on the data known as the workflow. It is specified after the process of auditing the data and is crucial in achieving the end product of high-quality data. In order to achieve a proper workflow, the causes of the anomalies and errors in the data have to be closely considered.

2.3 Workflow execution
In this stage, the workflow is executed after its specification is complete and its correctness is verified. The implementation of the workflow should be efficient, even on large sets of data, which inevitably poses a trade-off because the execution of a data-cleansing operation can be computationally expensive.

2.4 Post-processing and controlling
After executing the cleansing workflow, the results are inspected to verify correctness. Data that could not be corrected during execution of the workflow is manually corrected, if possible. The result is a new cycle in the data-cleansing process where the data is audited again to allow the specification of an additional workflow to further cleanse the data by automatic processing.
Good quality source data has to do with "Data Quality Culture" and must be initiated at the top of the organization. It is not just a matter of implementing strong validation checks on input screens, because almost no matter how strong these checks are, they can often

still be circumvented by the users. There is a nine-step guide for organizations that wish to improve data quality:

- Declare a high level commitment to a data quality culture
- Drive process reengineering at the executive level
- Spend money to improve the data entry environment
- Spend money to improve application integration
- Spend money to change how processes work
- Promote end-to-end team awareness
- Promote interdepartmental cooperation
- Publicly celebrate data quality excellence
- Continuously measure and improve data quality
- Others include:

2.5 Parsing

For the detection of syntax errors. A parser decides whether a string of data is acceptable within the allowed data specification. This is similar to the way a parser works with grammars and languages.

2.6 Data transformation

Data transformation allows the mapping of the data from its given format into the format expected by the appropriate application. This includes value conversions or translation functions, as well as normalizing numeric values to conform to minimum and maximum values.

2.7 Duplicate elimination

Duplicate detection requires an algorithm for determining whether data contains duplicate representations of the same entity. Usually, data is sorted by a key that would bring duplicate entries closer together for faster identification.

2.8 Statistical methods

By analyzing the data using the values of mean, standard deviation, range, or clustering algorithms, it is possible for an expert to find values that are unexpected and thus erroneous. Although the correction of such data is difficult since the true value is not known, it can be resolved by setting the values to an average or other statistical value. Statistical methods can also be used to handle missing values which can be replaced by one or more plausible values, which are usually obtained by extensive data augmentation algorithms.

3. Tool Support

A large variety of tools is available on the market to support data transformation and data cleaning tasks, in particular for data warehousing. Some tools concentrate on a specific domain, such as cleaning name and address data, or a specific cleaning phase, such as data analysis or duplicate elimination. Due to their restricted domain, specialized tools typically perform very well but must be complemented by other tools to address the broad spectrum of transformation and cleaning problems. Other tools, e.g., ETL tools, provide comprehensive transformation and workflow capabilities to cover a large part of the data transformation and cleaning process. For comprehensive vendor and tool listings, see commercial websites, e.g., Data Warehouse Information Center (www.dwinfocenter.org), Data Management Review (www.dmreview.com), Data Warehousing Institute (www.dw-institute.com) 10 A general problem of ETL tools is their limited interoperability due to proprietary application programming interfaces (API) and proprietary metadata formats making it difficult to combine the functionality of several tools. We first discuss tools for data analysis and data reengineering which process instance data to identify data errors and inconsistencies, and to derive corresponding cleaning transformations. We then present specialized cleaning tools and ETL tools, respectively.

3.1 Data analysis and reengineering

Tools According to our classification in 3.1, data analysis tools can be divided into data profiling and data mining tools. MIGRATIONARCHITECT (Evoke Software) is one of the few commercial data profiling tools. For each attribute, it determines the following real metadata: data type, length, cardinality, discrete values and their percentage, minimum and maximum values, missing values, and uniqueness. MIGRATIONARCHITECT also assists in developing the target schema for data migration. Data mining tools, such as WIZRULE (Wiz Soft) and DATAMININGSUITE (Information Discovery), infer relationships among attributes and their values and compute a confidence rate indicating the number of qualifying rows. In particular, WIZRULE can reveal three kinds of rules: mathematical formula, if-then rules, and spelling-based rules indicating misspelled names, e.g. Value Edinburgh appears 52 times in field Customer; 2 case(s) contain similar value(s)". WIZRULE also automatically points to the deviations from the set of the discovered rules as suspected errors. Data reengineering tools, e.g., INTEGRITY (Varity), utilize discovered patterns and rules to specify and perform cleaning transformations, i.e., they reengineer legacy data. In INTEGRITY, data instances undergo several analysis steps, such as parsing, data typing, and pattern and frequency analysis. The result of these steps is a tabular representation of field contents, their patterns and frequencies, based on which the pattern for standardizing data can be selected. For specifying cleaning transformations, INTEGRITY provides a language including a set of operators for column transformations (e.g., move, split, delete) and row transformation (e.g., merge, split). INTEGRITY identifies and consolidates records using a statistical matching technique. Automated weighting factors are used to compute scores for ranking matches based on which the user can select the real duplicates.

3.2 Specialized cleaning tools

Specialized cleaning tools typically deal with a particular domain, mostly name and address data, or concentrate on duplicate elimination. The transformations are to be provided either in advance in the form of a rule library or interactively by the user. Alternatively, data transformations can automatically be derived from schema matching tools such as described in [21]. Special domain cleaning: Names and addresses are recorded in many sources and typically have high cardinality. For example, finding customer matches is very important for

customer relationship management. A number of commercial tools, e.g., Idcentric (First Logic), Pureintegrate (Oracle), Quickaddress (Qas Systems), Reunion (Pitney Bowes), and TRILLIUM (Trillium Software), focus on cleaning this kind of data. They provide techniques such as extracting and transforming name and address information into individual standard elements, validating street names, cities, and zip codes, in combination with a matching facility based on the cleaned data. They incorporate a huge library of pre-specified rules dealing with the problems commonly found in processing this data. For example, TRILLIUM's extraction (parser) and matcher module contains over 200,000 business rules. The tools also provide facilities to customize or extend the rule library with user-defined rules for specific needs. Duplicate elimination: Sample tools for duplicate identification and elimination include Datacleanser (Edd), Merge/Purgelibrary (Qm Software), Matchit (Help IT Systems), and MASTERMERGE (Pitney Bowes). Usually, they require the data sources already be cleaned for matching. Several approaches for matching attribute values are supported; tools such as Datacleanser and Merge/Purgelibrary also allow user-specified matching rules to be integrated.

3.3 Business intelligence (BI) tool

Is regarding to create a value for the organizations depends on data or, more precisely, on facts. While it looks like another buzzword to describe what successful entrepreneurs have been doing for years, that is, using business common way. From a modern business-value perspective, corporations use BI to develop decision-making capabilities for managerial processes (e.g., planning, budgeting, controlling, assessing, measuring, and monitoring) and to ensure vital information is explored in a appropriate manner. Computer systems are the equipment that help us do work better, faster, and with more reliability and effectivity. Business intelligence systems, is also known as EIS[Executive Information Systems], or Decision Support Systems, are a non-transactional IT system used to support business decision making and resolve management issues, generally used by executives and managers. Almost all people agrees that OLAP and data warehouse systems are a vital and essential part of business intelligence systems. Many business intelligence systems were in the structure of a data warehouse systems.

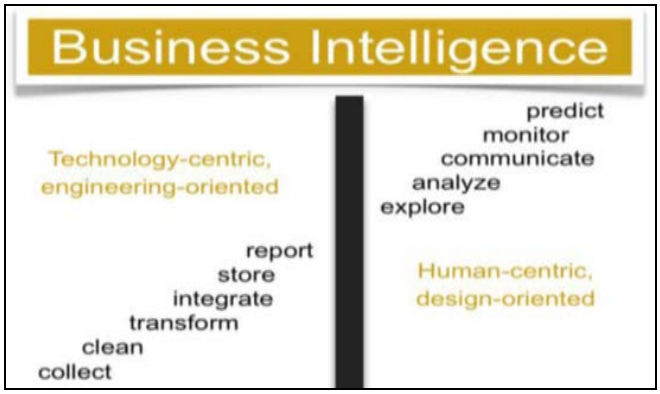

Fig 1: Concept of Business Intelligence

4. Data Quality

High-quality data needs to pass a set of quality criteria. Those include:

4.1 Validity

The degree to which the measures conform to defined business rules or constraints (see also Validity (statistics)). When modern database technology is used to design data-capture systems, validity is fairly easy to ensure: invalid data arises mainly in legacy contexts (where constraints were not implemented in software) or where inappropriate data-capture technology was used (e.g., spreadsheets, where it is very hard to limit what a user chooses to enter into a cell, if cell validation is not used). Data constraints fall into the following categories:

- **Data-Type Constraints** – e.g., values in a particular column must be of a particular datatype, e.g., Boolean, numeric (integer or real), date, etc.
- **Range Constraints**: typically, numbers or dates should fall within a certain range. That is, they have minimum and/or maximum permissible values.
- **Mandatory Constraints**: Certain columns cannot be empty.
- **Unique Constraints**: A field, or a combination of fields, must be unique across a dataset. For example, no two persons can have the same social security number.
- **Set-Membership constraints**: The values for a column come from a set of discrete values or codes. For example, a person's gender may be Female, Male or unknown (not recorded).
- **Foreign-key constraints**: This is the more general case of set membership. The set of values in a column is defined in a column of another table that contains unique values.

For example, in a US taxpayer database, the "state" column is required to belong to one of the US's defined states or territories: the set of permissible states/territories is recorded in a separate States table. The term foreign key is borrowed from relational database terminology.

4.2 Accuracy

The degree of conformity of a measure to a standard or a true value - see also Accuracy and precision. Accuracy is very hard to achieve through data-cleansing in the general case, because it requires accessing an external source of data that contains the true value: such "gold standard" data is often unavailable. Accuracy has been achieved in some cleansing contexts, notably customer contact data, by using external databases that match up zip codes to geographical locations (city and state), and also help verify that street addresses within these zip codes actually exist.

4.3 Completeness

The degree to which all required measures are known. Incompleteness is almost impossible to fix with data cleansing methodology: one cannot infer facts that were not captured when the data in question was initially recorded. (In some contexts, e.g., interview data, it may be possible to fix incompleteness by going back to the

original source of data, i.e., re-interviewing the subject, but even this does not guarantee success because of problems of recall - e.g., in an interview to gather data on food consumption, no one is likely to remember exactly what one ate six months ago. In the case of systems that insist certain columns should not be empty, one may work around the problem by designating a value that indicates "unknown" or "missing", but supplying of default values does not imply that the data has been made complete.

4.4 Consistency

The degree to which a set of measures are equivalent in across systems (see also Consistency). Inconsistency occurs when two data items in the data set contradict each other: e.g., a customer is recorded in two different systems as having two different current addresses, and only one of them can be correct. Fixing inconsistency is not always possible: it requires a variety of strategies - e.g., deciding which data were recorded more recently, which data source is likely to be most reliable (the latter knowledge may be specific to a given organization), or simply trying to find the truth by testing both data items (e.g., calling up the customer).

4.5 Uniformity

The degree to which a set data measures are specified using the same units of measure in all systems (see also Unit of measure). In datasets pooled from different locales, weight may be recorded either in pounds or kilos, and must be converted to a single measure using an arithmetic transformation.

The term integrity encompasses accuracy, consistency and some aspects of validation (see also data integrity) but is rarely used by itself in data-cleansing contexts because it is insufficiently specific. (For example, "referential integrity" is a term used to refer to the enforcement of foreign-key constraints above.)

5. Challenges and Problems

5.1 Error correction and loss of information

The most challenging problem within data cleansing remains the correction of values to remove duplicates and invalid entries. In many cases, the available information on such anomalies is limited and insufficient to determine the necessary transformations or corrections, leaving the deletion of such entries as a primary solution. The deletion of data, though, leads to loss of information; this loss can be particularly costly if there is a large amount of deleted data.

5.2 Maintenance of cleansed data

Data cleansing is an expensive and time-consuming process. So after having performed data cleansing and achieving a data collection free of errors, one would want to avoid the re-cleansing of data in its entirety after some values in data collection change. The process should only be repeated on values that have changed; this means that a cleansing lineage would need to be kept, which would require efficient data collection and management techniques.

5.3 Data cleansing in virtually integrated environments

In virtually integrated sources like IBM's Discovery Link, the cleansing of data has to be performed every time the data is accessed, which considerably increases the response time and lowers efficiency.

5.4 Data-cleansing framework

In many cases, it will not be possible to derive a complete data-cleansing graph to guide the process in advance. This makes data cleansing an iterative process involving significant exploration and interaction, which may require a framework in the form of a collection of methods for error detection and elimination in addition to data auditing. This can be integrated with other data-processing stages like integration and maintenance.

6. Limitation

The limitations of the study are as follows:

1. The study has been done on narrow basis.
2. This research has been conducted on one or two IT organizations.

7. Conclusion

We further outlined the major steps for data transformation and data cleaning and emphasized the need to cover schema- and instance-related data transformations in an integrated way. Furthermore, we provided an overview of commercial data cleaning tools. While the state-of-the-art in these tools is quite advanced, they do typically cover only part of the problem and still require substantial manual effort or self-programming. So far only a little research has appeared on data cleaning, although the large number of tools indicates both the importance and difficulty of the cleaning problem. We see several topics deserving further research. First of all, more work is needed on the design and implementation of the best language approach for supporting both schema and data transformations. For instance, operators such as Match, Merge or Mapping Composition have either been studied at the instance (data) or schema (metadata) level but may be built on similar implementation techniques. Data cleaning is not only needed for data warehousing but also for query processing on heterogeneous data sources, e.g., in web-based information systems. This environment poses much more restrictive performance constraints for data cleaning that need to be considered in the design of suitable approaches. Furthermore, data cleaning for semi-structured data, e.g., based on XML, is likely to be of great importance given the reduced structural constraints and the rapidly increasing amount of XML data.

8. References

1. Abiteboul S, Clue S, Milo T, Mogilevsky P, Simeon J. Tools for Data Translation and Integration. In [26]:3-8, 1999.
2. Batini C, Lenzerini M, Navathe SB. A Comparative Analysis of Methodologies for Database Schema Integration. In Computing Surveys. 1986; 18(4):323-364.
3. Bernstein PA, Bergstraesser T. Metadata Support for

Data Transformation Using Microsoft Repository. 1999; 26:9-14.
4. Bernstein PA, Dayal U. An Overview of Repository Technology. Proc. 20th VLDB, 1994.
5. Bouzeghoub M, Fabret F, Galhardas H, Pereira J. Simon E, Matulovic M. Data Warehouse Refreshment. 16:47-67.
6. Chaudhuri S, Dayak U. An Overview of Data Warehousing and OLAP Technology. ACM SIGMOD Record. 1997, 26(1).
7. Cohen W. Integration of Heterogeneous Databases without Common Domains Using Queries Based Textual Similarity. Proc. ACM SIGMOD Conf. on Data Management, 1998.
8. Do HH, Rahm E. On Metadata Interoperability in Data Warehouses. Techno. Report 1-2000, Department of Computer Science, University of Leipzig. http://dol.uni-leipzig.de/pub/2000-13.
9. Doan AH, Domingo's P, Levy AY. Learning Source Description for Data Integration. Proc. 3rd Intl. Workshop the Web and Databases (Web DB), 2000.
10. Lee ML, Lu H, Ling TW, Kio YT, Cleansing Data for Mining and Warehousing. Proc. 10th DEXA, 1999.
11. Li WS, Clifton S, Semint. A Tool for Identifying Attribute Correspondences in Heterogeneous Databases Using Neural Networks. In Data and Knowledge Engineering. 2000; 33(1):49-84.
12. Milo T, Zohar S. Using Schema Matching to Simplify Heterogeneous Data Translation. Proc. 24th VLDB, 1998.
13. Jump up^ Wu S. "A review on coarse warranty data and analysis", Reliability Engineering and System. 2013; 114:1–11. doi:10.1016/j.ress.2012. 12.021