Analysis of Data Cleansing Approaches regarding Dirty Data – A Comparative Study

Kofi Adu-Manu Sarpong
Institute of Computer Science
Valley View University, Accra-Ghana
P.O. Box VV 44, Oyibi-Accra

John Kingsley Arthur
Institute of Computer Science
Valley View University, Accra-Ghana
P.O. Box VV 44, Oyibi-Accra

ABSTRACT

Data Cleansing is an activity involving a process of detecting and correcting the errors and inconsistencies in data warehouse. It deals with identification of corrupt and duplicate data inherent in the data sets of a data warehouse to enhance the quality of data. The research was directed at investigating some existing approaches and frameworks to data cleansing. That attempted to solve the data cleansing problem and came up with their strengths and weaknesses which led to the identification of gabs in those frameworks and approaches. A comparative analysis of the four frameworks was conducted and by using standard testing parameters a proposed feature was discussed to fit in the gaps.

General Terms

Data warehousing, data cleansing, data sets

Keywords:

Framework, Strengths and weaknesses, gap analysis

1. INTRODUCTION

Poor data quality (DQ) continues to be an important issue for companies. Cleansing data of errors in structure and content is important for data warehousing and integration. There is a high need of data cleansing since it is centerd on the quality of data and to make the data “fit for use” by its user’s data through the removal of errors and having proper documentation. Current solutions for data cleaning involve many iterations of data “auditing”

to find errors, and long-running transformations to fix them. There are various approaches used in cleaning data in manufacturing industries, schools/colleges/universities, organizations and many more. Users need to endure long waits, and often write complex transformation scripts.

Muller et al., outlined several DC frameworks among which the most popular ones are reviewed in this paper in order to know their strengths and weakness. These include Potter’s Wheel, IntelliClean, AJAX and ARKTOS [1].

2. DATA CLEANSING APROACHES

2.1Potter’s Wheel: An Interactive Data Cleaning System

Potter’s Wheel is an interactive data cleaning system that tightly integrates transformation and discrepancy detection. Users gradually build transformations to clean the data by adding or undoing transforms on a spreadsheet-like interface; the effect of a transform is shown at once on records visible on screen [1, 2].

These transforms are specified either through simple graphical operations, or by showing the desired effects on example data values. In the background, Potter’s Wheel automatically infers structures for data values in terms of user-defined domains, and accordingly checks for constraint violations. Thus users can gradually build a transformation as discrepancies are found, and clean the data without writing complex programs or enduring long delays although the data cleansing process is not documented [3, 4]. Data is displayed on a scalable spreadsheet interface [10].

2.1.1 Potter’s Wheel: Design Goals

- Eliminate wait time during each step
- Eliminate programming, but keep user “in the loop”
- Unify detection and transformation
- Extensibility

2.1.2 Critique of Porter’s Wheel framework

From the above review, it was realised that Potter’s Wheel had some strength as well as some weaknesses. In this section, some of the strengths and weaknesses in Potter’s Wheel are critically looked at.

2.1.2.1 Strengths

i. Potter’s Wheel is an interactive system for data transformation and cleaning. By integrating discrepancy detection and transformation, Potter’s Wheel allows users to gradually build a transformation to clean the data by adding transforms as discrepancies are detected. Users can specify transforms through graphical operations or through examples, and see the effect instantaneously, thereby allowing easy experimentation with different transforms.

i. Parsing strings using structures of user defined domains result in a general and extensible discrepancy detection mechanism. Such domains provide a powerful basis for specifying Split transformations through example values.

ii. End user power: there are graphical representations of transformations on data such that users do have a chance to add or even undo effects on transformations.

2.1.2.2 Weaknesses

i. Scrolling by the end-user is directly related to the scope of the automatic discrepancy detection. Hence if the end user does not scroll up to that record, a particular discrepancy may be available but will not be detected.

ii. Data to be tested for errors is fetched in samples of the source data; however, the sequence of the fetch remains unknown. That is whether the fetch is done sequentially or randomly or by this implies that if there are errors in the data. The situation of random may call for the third situation as described in point 'a'.

iii. Delay of time – as stated in the second concept, it is possible that same sampled data could be fetched and retested over and over again. When this situation occurs, greater time will be used in detecting discrepancies within the data warehouse.

iv. Although the user interface is said to be interactive, however, its effectiveness was not measured. Therefore, an appropriate tool such as an interactive querying system needs to be used to measure the effectiveness of the user interface.

As shown in table 1, Potter's Wheel has shown to be very interactive. Due to its interactivity it makes it easy to use. Although it has this capability, the research has shown clearly that this framework only deals with text data and does not consider any other database. Its operation is such that it involves much human assistance hence the reason for the high human dependency for exceptional errors if they occur. One critical aspect of data cleansing was not considered that is how the cleansed data is maintained.

2.2 IntelliClean: A Knowledge-Based Intelligent Data Cleaner

Intelliclean is a knowledge-based Intelligent Data cleaner. The system proposes a generic knowledge-based framework for effective data cleaning that implements existing cleaning strategies. Intelliclean employs a new method to compute transitive closure under uncertainty which handles the merging of groups of inexact duplicate records. Experimental results show that this framework can identify duplicates and anomalies with high recall and precision [5].

2.2.2 Conceptual model and benchmarking metrics for data cleaning

The model starts by receiving "dirty" datasets with variety of errors. Cleaning strategies are applied to the dataset with the objective of obtaining consistent and correct data as the output. The effectiveness of the cleaning strategies will thus be the degree by which data quality is improved through cleaning.

2.2.3 Critique of IntelliClean Framework

A thorough study was conducted in reviewing Intelliclean. Since there is no perfect system, Intelliclean also has its own strengths and weaknesses. This section considers some of these strengths and weaknesses.

2.2.3.1 Strengths

i. The introduction of an expert system within the framework makes it much more generic because the system is able to learn. Hence, changes made in the textual databases will be detected and identified as a transformed data automatically by the framework. This makes it more efficient than other frameworks.

ii. It provides effective rules for resolving the recall-precision dilemma which is a short coming in other frameworks; hence, it enhances the effectiveness of the framework.

2.2.3.2 Weaknesses

i. The IntelliClean considered only textual forms of data in the data warehouse, implying that duplicates of data of other formats will not be identified. For example, image, video, graphics and other file formats are not applicable with the IntelliClean.

ii. Data source from the web is not applicable to the Intelliclean system. It was not considered as a source. Therefore, a generic and data source independent system when developed can resolve this problem.

2.3 ARKTOS: A Tool for Data Cleaning and Transformation in Data Warehouse Environments

ARKTOS as cited by [6] is a framework capable of modelling and executing the Extraction-Transformation-Load process (ETL process) for data warehouse creation. The authors consider data cleansing as an integral part of this ETL process which consists of single steps that extract relevant data from the sources, transform it to the target format and cleanse it, then loading it into the data warehouse.

A meta-model is specified allowing the modelling of the complete ETL process. The single steps (cleansing operations) within the process are called activities. Each activity is linked to input and output relations [7].

The logic performed by an activity is declaratively described by a SQL-statement. Each statement is associated with a particular error type and a policy which specifies the behaviour (the action to be performed) in case of error occurrence.

Six types of errors can be considered within an ETL process specified and executed in the ARKTOS framework. PRIMARY KEY VIOLATION, UNIQUENESS VIOLATION and REFERENCE VIOLATION are special cases of integrity constraint violations. The error type NULL EXISTENCE is concerned with the elimination of missing values. The remaining error types are DOMAIN MISMATCH and FORMAT MISMATCH referring to lexical and domain format errors as cited by [1].

2.3.2 Critique of ARKTOS framework

From thorough review and analysis of ARKTOS framework, some strengths as well as weakness were discovered. Below are the strengths and weaknesses.

2.3.2.1 Strengths

i. The Arktos, is capable of modelling and executing practical ETL scenarios by providing explicit primitives for the capturing of common tasks (like data leaning, scheduling and data transformations).

ii. The system makes it easier for authoring by providing three ways to describe an ETL scenario: a graphical point-and-click front end and two declarative languages: XADL (an XML variant), which is more verbose and easy to read and SADL (an SQL-like language) which has a quite compact syntax.

2.3.2.2 Weakness

i. Although the research could identify the optimization problems (Identification of a small set of algebraic operators, Local optimization of ETL activities and, Global (multiple) optimization of ETL activities) of ETL processes but was silent on the solution to the identified problems.

2.1 AJAX

According to [1, 8, and 9], AJAX is an extensible and flexible framework attempting to separate the logical and physical levels of data cleansing. The logical level supports the design of the data cleansing workflow and specification of cleansing operations performed, while the physical level regards their implementation.

The main goal of AJAX is to facilitate the specification and execution of data cleaning programs either for a single source or for integrating multiple data sources [8]. In this paper the main issue is transforming existing data from several data collections into a target schema and eliminating duplicates within this process.

In order to achieve this, five major transformations are discussed in the paper. These transformations are mapping, viewing, matching, clustering, and merging. According to [8], the mapping transformation standardizes all data formats and simply produces a more appropriate data format by applying two major operations such as splitting and merging. The matching compares several records and finds pairs that match specified criteria. Another transformation called clustering groups together matching pairs with high similarity by applying a given group criteria. The merging transformation is then applied to each of these clusters in order to eliminate duplicates or produce a new record based on the integrated results. AJAX tends to be more complex as compared to other approaches.

2.3.2.3 Critique of AJAX

2.3.2.3.1 Strengths

i. It allows customization because it is extensible. For example extension of functions from other libraries, combination with primitives with SQL etc. all of this adds dynamism to the functionality of the AJAX.

ii. Interactivity of system is high.

2.3.2.3.2 Weaknesses

i. There is higher level of dependency on human expert for the resolution of exceptional cases that arise as a result of executing a micro-operator. This is a demanding situation as compared with the IntelliClean system which has a an expert module which caters for such kinds of exception

ii. The approach used in creating quality data (cleaned data) is very complex as compared to other frameworks. This will supposedly suggest that, it will take greater time to clean data as compared with other frameworks.

After the thorough criticizing of the data cleansing frameworks, a comparative analysis was conducted to compare the four data cleansing frameworks basing on four parameters as shown in table 1 below. It was realized that none of the frameworks considered the maintenance aspect of the data cleansing process.

Table 1: Comparative analysis of Data Cleaning Frameworks

Parameter	Porters Wheel	AJAX	IntelliClean	ARKTOS
Interactivity	It is very interactive; hence easy to use	Complex interface and hence not friendly to non-technical persons.	Interactive with end user. However, requires little input from end-users	Highly interactive; it has graphical interfaces for loading and executing validations on loaded files.
Data format/Structure	Text	Text	Text	Text
Human dependency	High human dependency for exceptional errors.	High human dependency. Example; evaluation and validation of errors are fully dependent on human expert.	Very minimal because of the expert module embedded in the system.	Although the system has complex modules for dealing with duplicates, however, there is a high dependency on human expects for error correction.
Maintenance	Not considered	Not considered	Not considered	Not considered

3 GAP ANALYSIS

In this section, the various weaknesses are discussed and possible solutions are provided to fill the gap identified in the frameworks and concepts of cleaning data in the data warehouse as identified above.

In the review of the Porter's wheel framework, IntelliClean, ARKTOS, AJAX, the following defects were identified

i. Scrolling by the end-user is directly related to the scope of the automatic discrepancy detection. Hence if the end-user does not scroll up to that record, a particular discrepancy might be available but will not be detected.

ii. Data to be tested for errors is fetched in samples of the source data; however, the sequence of the fetch remains unknown. That is whether the fetch is done sequentially or randomly or by this implies that if there are errors in the data. The situation of random may call for the third situation as described in point ‘a’.

iii. Delay of time – as stated in the second concept, it is possible that same sampled data could be fetched and retested over and over again. When this situation occurs, greater time will be used in detecting discrepancies within the data warehouse

iv. Although the user interface is said to be interactive, however, its effectiveness was not measured. Therefore, an appropriate tool such as an interactive querying system needs to be used to measure the effectiveness of the user interface.

v. The IntelliClean considered only textual forms of data in the data warehouse, implying that duplicates of data of other formats will not be identified. For example, image, video, graphics and other file formats are not applicable with the IntelliClean. Data source from the web is not applicable to the Intelliclean system. It was not considered as a source. Therefore, a generic and data source independent system when developed can resolve this problem.

vi. Although the research could identify the optimization problems (Identification of a small set of algebraic operators, Local optimization of ETL activities and, Global (multiple) optimization of ETL activities) of ETL processes but were silent on the solution to the identified problems.

vii. There is higher level of dependency on human expert for the resolution of exceptional cases that arise as a result of executing a micro-operator. This is a demanding situation as compared with the IntelliClean system which has an expert module which caters for such kinds of exception.

viii. The approach used in creating quality data (cleaned data) is very complex as compared to other frameworks. This will supposedly suggest that it will take greater time to clean data as compared with other frameworks.

ix. Maintenance is not considered in the framework for cleaning of data. Hence after cleaning of data, the question is what next? Data needs to be maintained.

x. Although several cleaning tools were compared in terms of the file formats that they can clean, however, how effective they do the cleaning was swept under the carpet. The effectiveness of these tools should be clearly known.

xi. The research was able to measure the effectiveness of the developed algorithm against the standard algorithm, however, with very minimal iterations. It is also identified that minimal iterations could result in inability to identify duplicate records. Therefore, there is doubt on the outcome of this research.

xii. The data warehouse that was used for the testing of results was small. Huge data warehouse will bring out the exact outcome of the research.

xiii. There is an extensive involvement of users. For example, after detection of duplicates (errors), the end user is asked to validate. Continues human validation may result in other human errors in the data warehouse.

3.1 Summary of gap analysis

Majority of the frameworks do have a common problem of interactivity with the end-user, consideration of only textual contents of databases and hence multimedia databases, procedure for testing is undefined (no standardization), higher human dependency, hidden details of cleaning processes and lastly, maintenance lost in all the framework.

From the above discussion the following will be proposed to fill the gaps identified from the review.

- **User friendly interface:** A customizable user interface is developed which makes it easier for both technical and non technical users to load and clean data. Data cleaning here does not depend on scrolling area, it picks the file directly.
- **Multimedia databases:** The proposed framework will consider multimedia databases. Problems of databases do not always arise from text formats of data. Therefore, in this framework, databases consider text, image, videos and graphics.
- **Showcase of detailed cleaning process:** in the existing systems the details of the cleaning processes are not showed. However, in the proposed system, a comprehensive detail is provided.
- **Maintenance:** Periodically, the cleaned data will be automatically reloaded and cleaned. This will make it possible such that most of the time there will be a clean data in the warehouse.

3.2 Parameters for standard testing

- Interactivity
- File format
- Loading rate
- Data size
- Processor speed
- Memory size

The parameters listed above in section 3.2 were used for standard testing especially when it comes to the tools usability. These parameters where used to test the efficiency of the four frameworks discussed in this paper. The interactivity takes into consideration the kind of interface (graphical or command line) used in each of the frameworks. File format has to do with the type of file (databases, text, etc), the size of the data was considered and the rate of loading the data (that is the time it takes to load a particular data). To also test, the speed of the processor and the size of the testing machine were considered.

4 CONCLUSIONS AND FUTURE WORK

Data is important to every organization and for that matter no organization can survive without data. There are several algorithms and frameworks that attempt to eliminate and clean the “dirty data”. Among these approaches, this research focused on the most popular data cleansing frameworks (Potter’s Wheel, AJAX, Intelliclean and ARKTOS) and outlined their strengths and weakness. A comparative analysis based on the gaps was conducted using a standard parameter for testing each of the frameworks. Future work can involve researching into how to overcome the weaknesses of the four frameworks studied and implementing the four possible solutions proposed in this research work to enhance the throughput of the any future development.

5 REFERENCES

[1] Heiko Muller, Johann-Christoph Freytag. (2003). Problems, Methods, and Challenges in Comprehensive Data Cleansing, pp. 21.

[2] Raman V and Hellerstein J.M, Potter's Wheel: An Interactive Data Cleaning System, Proceedings of the 27th VLDB Conference, Roma, Italy, 2001, pp. 1-10.

[3] F. Naumann, Quality-Driven Query Answering for Integrated Information Systems, Lecture Notes in Computer Science, LNCS 2261, Springer, 2002. pp. 34

[4] Louardi BRADJI, Mahmoud BOUFAIDA. (2011). Open User Involvement in Data Cleaning for Data Warehouse Quality. International Journal of Digital Information and Wireless Communications (IJDIWC) 1(2), pp. 573.

[5] Mong L.L, Tok W.L and Wai L.L.(2000). IntelliClean : A Knowledge-Based Intelligent Data Cleaner, ACM, pp. 290-294

[6] P. Vassiliadis, Z.a Vagena, S. Skiadopoulos, N. Karayannidis, T. Sellis. (2001). ARKTOS: towards the modeling, design, control and execution of ETL Processes . Information Systems, Vol.26 , pp. 537-556.

[7] Panos V., Zografoula V, Spiros S., and Nikos K.(2000). ARKTOS: A Tool For Data Cleaning and Transformation in Data Warehouse Environments. Bulletin of the IEEE Computer Society Technical Committee on Data Engineering, pp.1-6

[8] H. Galhards, D. Florescu, D. Shasha, E. Simon. (May 2000). AJAX: An extensible data cleaning tool. Proceedings of the ACM SIGMOD on Management of data, Dallas, TX USA, pp. 21-22.

[9] Herbert, K.G., Wang, J.T.L. (2007). Biological data cleaning: a case study. In Int. J. of Information Quality, vol. 1, number. 1, pp. 60–82

[10] S. Chaudhuri and U. Dayal. (1997). An overview of data warehousing and OLAP technology. In SIGMOD Record. pp. 65-74