JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Exterminate Inconsistencies and Errors from Data Sets with Data Sanitization

[1]B.Yasaswi, [2]B.Hanu Krishna, [3]B.Vamsi Krishna, [4]B V Subba Rao, [5]S.Sai Kumar
[1]III B.Tech CSE, PVP Siddhartha Institute of Technology, Vijayawada-7, India,
[2]IV B.Tech CSE, VIT University, Vijayawada, Andhra Pradesh, India,
[3]MS in Cyber Security, The University of Tampa, Florida, USA,
[4]Dept of IT, PVP Siddhartha Institute of Technology, A.P, India,
[5]Dept of IT, PVP Siddhartha Institute of Technology, A.P, India,

Abstract : The problem of data cleaning, which consists of removing inconsistencies and errors from original data sets, is well known in the area of decision support systems and data warehouses. This holds regardless of the application - relational database joining, web-related, or scientific. In all cases, existing ETL (Extraction Transformation Loading) and data cleaning tools for writing data cleaning programs are insufficient. The main reason for using the computers is to organize the data in an efficient and effective manner .In early days for valuable data can be organization sake we have to use the tools like Queries. In these some problems are arises. That is why these languages are called as Data Management systems. There were so many limitations in the management system like data inconsistency, inconvenience in retrieval of data etc. Because of all these limitations we have to face the problems like memory inefficiency and heavy in consumption of time and also lack of quality. To overcome all these problems we have designed a software(what I mean is ETL tool) which organizes the data in a very efficient manner with respect to redundant data. Our project deals with the data organization by giving all data oriented features and by solving the problems like data inconsistency and data redundancy.
Data from different data sources are usually first transformed and cleaned before being loaded into the data warehouse Data Cleaning (cleansing or scrubbing) is the process of detecting and removing errors, inconsistencies and data redundancies from data in order to improve the quality of data.

IndexTerms- **data cleansing, data warehouse, ETL, redundancy**.

I. INTRODUCTION

A Data Warehouse integrates data from a number of data sources for purposes of end user querying and analysis.
Data warehouses, which are repositories of data collected from several data sources, form the backbone of most current CRM and decision support applications. Since data sources are independent, they may adopt independent and potentially inconsistent conventions. For example, one source may adopt the use of standard while another source adopts the use of fully expanded descriptions. Moreover, data entry mistakes at any of these sources introduce more errors. Since high quality data is essential for gaining the confidence of users of CRM and decision support applications developed over data warehouses, ensuring high data quality is critical to the success of data warehouse implementations. Therefore, significant amount of time and money are spent on the process of detecting and correcting errors and inconsistencies.
The process of cleaning dirty data is often referred to as data cleaning. Since the types of errors and inconsistencies can be domain-specific, it is important and challenging to develop generic domain-independent data cleansing solutions. Our goal in the data cleaning project is to develop a set of domain-independent tools which can be used for developing effective and efficient data cleaning solutions.

II. RESEARCH METHODOLOGY

Data Cleansing:

Source systems contain “dirty data” that must be cleansed. ETL software contains rudimentary data cleansing capabilities. Specialized data cleansing software is often used. Important for performing name and address correction and house holding functions. Leading data cleansing vendors include Vality (Integrity), Harte-Hanks (Trillium), and First logic (i.d. Centric).

We have mainly two algorithms for classify the data at the source systems by means of cleaning.
The Token-Based Data Cleaning Algorithm (TB-Cleaner)
The Sorted Token-Based Data Cleaning Algorithm (STB-Cleaner)
The cleaning tasks consist of:
1. Record Duplicate detection (starting with dimension tables)
2. Record Duplicate Elimination (only one copy of duplicates in dimension tables)

3. Record unification (assigning same warehouse id to duplicates in the fact table).

The Token-Based Data Cleaning Algorithm (TB-Cleaner):

In the TB-Cleaner algorithm the Names and important fields like Date of Birth and Address are converted into tokens (short format) for the purpose of comparison.

Ex: Suppose considar the following example as consists the *Savings Account (SA) Customer table* and *Checking Account (CA) Customers table.*

SAVINGS ACCOUNT (SA) CUSTOMER TABLE

cid	Cname	Cbirth	Csex	Cphone	caddress
S001	John Smith O	25-Dec-70	M	(519)111-1234	Sunset #995 N9B3P4
S002	Tim E. Ohanekwu	10-Jan-1975	M	2566416	Roundup St. No. 695 n962t7
S003	Colette Jones	08/Aug/64	M	123-4567	600 XYZ apt 5a5 N7C4K4
S004	Ambrose A. Diana	Nov/11/72	F	5196669999	4 Church Rd N8k6T6
S005	Smith John	30-Oct-78	F	519 560 3626	182 Haven Ave M9B3T7

Fig1. Checking Account (CA) Customers table

cid	Name	Bday	Sex	phone	address
1001	S. John	25-12-70	M	1111234	995 Sunset N9B3P4
1002	John Cole	08-08-1964	M	Null	XYZ No. 600 apt 585 n7c4k4
1003	Ambo D. Dian	10-11-1972	M	566-5555	Church St. #4 n8k 6t6
1004	Ohanekwu T. E.	1-1-75	M	5192566416	#695 Randolph avenue N9B2T7
1005	Edema Tom Obi	23-Mar-1967	M	977-5950	98 Haven Rd M8C 8S4

Fig2. Checking Account (CA) Customers table

Consider the integration of two banking data sources for savings account (SA) in Table 1 and checking account (CA) in Table 2 to obtain the data warehouse fact table (Table 3) and dimension tables including Customer CDT (Table 4)

Fact table (yet to be cleaned)

Row	WID	Transtype	Account	Transtime	Amount
1	S005	D	SA	570A	525.25
2	1005	D	CA	585A	1015.99
3	1004	W	CA	1020P	150
4	S005	W	SA	825P	125.44
5	1001	D	CA	720P	650.33
6	S004	W	SA	660A	325.50
7	1002	W	CA	600A	250.16
8	S001	D	SA	990P	1005.53
9	1003	D	CA	1140P	450.50

Customer Dimension Table (Yet to be cleaned)

Row	WID	Sex	Phone	Birth	Address
S001	John Smith O	M	(519) 111-1234	25-Dec-70	Sunset #995 N9B3P4
S002	Tim E. Ohanekwu	M	2566416	10-Jan-1975	Roundup St. No. 695 n962t7
S003	Colette Jones	M	123-4567	08/Aug/64	600 XYZ apt 5a5 N7C4K4
S004	Ambrose A. Diana	F	5196669999	Nov/11/72	4 Church Rd N8k6T6
S005	Smith John	F	519 560 3626	30-Oct-78	182 Haven Ave M9B3T7

III.EXPERIMENT

Defining a cleaning algorithm that is less dependent on external interventions (like interactive user input or external data source which might not be available), and also less dependent on match score thresholds.
Improving the quality of token keys from dirty fields such that already formed short token keys can also be used for record comparisons and not just for sorting in order to improve on accuracy of result as well as on the processing time.

Given the dirty fact table and dimension table, CDT, the TB cleaner algorithm aims to produce two corresponding clean tables, starting with dimension table, CDT, by going through the following sequence of steps:
Step 1: Select and rank 2 or 3 fields based on their record identifying abilities. Selected fields from CDT table 4 are "Birth", "Name" and "Address" in the given order.
Step 2: Extract smart token for each selected field as follows. Form either numeric, alphabetic or alphanumeric tokens after removing stop words and unimportant characters like "#", "(".
A) *Numeric Tokens*: After necessary format conversions like for date 19-Dec-1978 to 19-12-1978, field content is decomposed into indivisible important members (e.g., to obtain 19 12 78), which when sorted
B) Alphanumeric Tokens: Only alphabets (aA –zZ) are in these tokens. The first character of each work is obtained and the defined token consists of all such in an order. E.g., Dr. Christie I. Ezeife and Ezeife Ije C. will both yield the same token CEI.
C) *Alphanumeric tokens:* After obtaining indivisible important members, both alphabetic and numeric tokens are defined as detailed above to get the desired token. E.g., 600 XYZ blvd apt 585 N7C4K4 is decomposed into 600 585 744 NCK and the defined token is 585600744NCK. The result of this step is a table of tokens.
Step 3: The table of tokens from step 2 is sorted separately on two most important fields, e.g., "birth" and "Name" token fields to obtain two sorted token tables.
Step 4: Duplicate Detection, Elimination and WID generation: Using each of the 2 token tables, identify all pairs of records as duplicates if they are (1) perfect match because their similarity match count (smc) is 1.0 or (2) near perfect match because their SMC is between 0.67 and 0.99.
The records are (3) no match if their SMC < 0.33, but they are (4)maybe a match if their SMC is between 0.33 and 0.66
SMC = number of corresponding token fields that match /number of token fields used
If records are maybe a match, their similarity match ratio (SMR) is computed as (2*number of common characters in the two tokens) /total number of characters in the two tokens.
The two tokens are a match if their **SMR** is greater or equal 0.67
The results from the two token tables are combined to obtain all duplicates as (S001, 1001), (S002, 1004), (S003, 1002) and (S004. 1003)
Wid is obtained as a concatenation of the first token with the second token of only one record in the duplicated lists (e.g., JOS122570 for both records 1 and 6).
While all duplicates in dimension table CDT are deleted but only the first kept with the new WID, all duplicates in the fact table are kept, but with the same new WID.

IV.RESULTS AND DISCUSSIONS

Experiments show that the TB cleaner has a recall close to 100%, which is always higher than the recall for Lee's and Basic's algorithm. Recall is a measure of cleaning accuracy equivalent to number of identified duplicates/number of actual duplicates
It can also be seen that as the size of data increases, the performance gain gap between the TB cleaner the others widens.

This method should produce faster response time with huge data because of use of short tokens in record comparisons and a limit of only 2 parses at data due to choice of two token tables.

V. CONCLUSION

The requirements of the end users are getting increased day to day. Large databases, huge amounts of data have to be processed in the real life from various data sources. More facilities and features are required to organize and analyze entire data efficiently.

Data storage, data manipulations and finding relation ship is the important tasks in the present situation. The existing systems were may be insufficient and inefficient for the particular organization.

Current application, only my project DATA WAREHOUSE CLEANSER, achieves this Extraction, Transformation and Loading Strategies almost all.

REFERENCES

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

[3] Bernstein, P.A.; Bergstraesser, T.: Metadata Support for Data Transformation Using Microsoft Repository. In 9-14, 1999
[4] Bernstein, P.A.; 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. In:47-67.
[6] Chaudhuri, S., Dayal, U.: An Overview of Data Warehousing and OLAP Technology. ACM SIGMOD Record 26(1), 1997.
[7] Cohen, W.: Integration of Heterogeneous Databases without Common Domains Using Queries Based Textual
[8] Similarity. Proc. ACM SIGMOD Conf. on Data Management, 1998.
[9] Li, W.S.; Clifton, S.: SEMINT: A Tool for Identifying Attribute Correspondences in Heterogeneous Databases
[10] Using Neural Networks. In Data and Knowledge Engineering 33(1):49-84, 2000.
[11] Milo, T.; Zohar, S.: Using Schema Matching to Simplify Heterogeneous Data Translation. Proc. 24th VLDB, 1998.
[12] Monge, A. E. Matching Algorithm within a Duplicate Detection System. IEEE Techn. Bulletin Data Engineering (4), 2000 (this issue).
[13] Monge, A. E.; Elkan, P.C.: The Field Matching Problem: Algorithms and Applications. Proc. 2nd Intl. Conf.
[14] Knowledge Discovery and Data Mining (KDD), 1996.