Analysis and Implementation of a Data Pre-processing System

Snigdha Petluru
Department of Information Technology
G. Narayanamma Institute of Technology and Science
Hyderabad, India
petlurus@gmail.com

Renu Nishitha Salver
Department of Information Technology
G. Narayanamma Institute of Technology and Science
Hyderabad, India
renu.nishitha@gmail.com

L Smitha
Assistant Professor,Department of Information Technology
G. Narayanamma Institute of Technology and Science
Hyderabad, India
smitha2005sri@gmail.com

Abstract—Today, we generate vast amounts of data each day, most of which is unstructured, incomplete, and more importantly inconsistent. In order to overcome the shortcomings of manually analyzing data, we have designed a data pre-processing system that cleans, integrates and transforms a data set. User specified files are integrated and stored in an automatically generated output file. A data table is also generated and the values are updated into their corresponding locations. In order to clean the missing values, we perform mean, median and mode on the complete data tuples in order to replace the missing data with these values. Transformation of our data is done by normalizing from wide ranges to narrow ranges [-1, 1] by implementing decimal scaling normalization, min-max normalization and z-score normalization .The processed data is stored in a .ARFF file which can be used for business requirements in a productive way.

Keywords—*data pre-processing; cleaning; transformation; mining;*

*****

I. Introduction

In today’s world, every gadget that we generate is bound to generate some data on a regular basis. In older days, we used traditional forms of data entry into registers. Since the rise of the computing age, we have come to a juncture where almost 90% of the data that we store has been generated in the last two years[1]. Much of this data is semi-structured, and most is inconsistent, and incomplete.

With the onset of big data mining, many governments are encouraging the public sector offices to publish data online in order to create transparency. This has opened opportunities for journalists and statisticians and data analysts to identify patterns by analyzing this data. But even so, this data is not fit to be analysed on the go. This can be attributed to the fact that the various e-forms, mobile data, catalogs and excel sheets are often incomplete. This brings forth the issue of the veracity of data, given that analysis of wrong data can lead to potentially disastrous results to the firm.

The rest of this paper describes the prototype of a data pre-processing system that aims to address the growing concerns of completeness and readability of data, by cleaning and transforming the given data files, using a simple, interactive interface that is structured around servlets in a Java environment. A responsive HTML template is used to take input from the user, thus ensuring that there is less complication on the user’s end.

II. Overview of the System

A. *Objective*

The objective is to examine the possibility of performing data pre-processing on any data set that is provided by the user. In the process of doing so, we try to ensure that there is minimum effort that is put on behalf of the user. Most of the process is done in the background. Integration is done, wherein the output file is generated automatically at the same location as the input files. Similarly, table creation and updating is done automatically. We perform cleaning and transformation on the various tuples containing numerical data and thus provide a more reliable output for the user to start working with, as depicted in Fig .1.

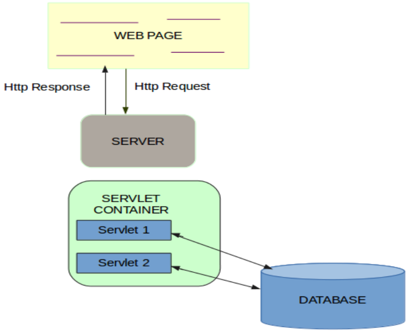


Fig .1 : Proposed system's default architecture

B. *Prerequisites*

The development of this code occurs on a Java platform. It is essential to use a web server in order to enable user interactions by means of a flat responsive web template that thereby communicates messages to servlets. These servlets are used to connect to the database and perform operations. We have used Oracle 10G to perform all data storing and modification related operations. Some of the automated data handling functions include:

- Data table creation
- Data modification
- Result Set generation and
- Data output generation

We use Java script to add dynamism to our web pages and the entire access to the user is taken care by installing Apache Tomcat server.

III. Cleaning

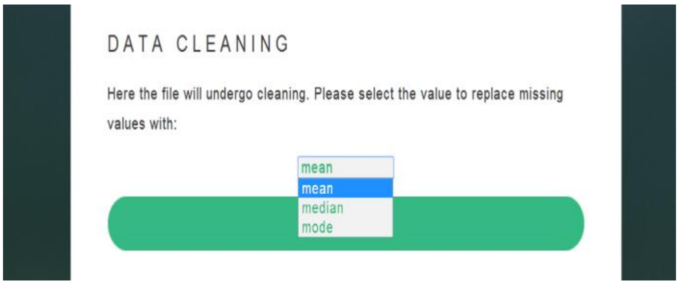
A. *Background*

In order to provide access to accurate and consistent data, consolidation of different data representations and elimination of duplicate information becomes necessary. Usually, the cleaning process filters the requests concerning non-analyzed resources such as images, multimedia files. For example, requests for graphical page content (*.jpg & *.gif images). By removing the unwanted data, we can easily reduce the log file size . For example, by filtering out image requests,more than 50% of the Web log files may be reduced when compared to the original ones.

When the multiple data files are to be integrated in a single file, there arises a necessity of cleaning. In the business world, incorrect data can be costly. Many companies use customer information that record data like contact information, addresses, and preferences. For instance, if the addresses are inconsistent, the company will suffer the cost of resending mail or even losing customers. So, there is an emergent need to make the data clean, consistent, accurate, relevant etc for better business analysis.

B. *Implementation*

We consider the data table that needs to be cleaned. All integral values that are missing are stored as -1. Firstly, we retrieve all the tuples of a column that are not assigned as -1. We perform a mean, median or mode on this result set. Then we update the entire column in the database by substituting all the missing values with one of the three options. The choice of which value to use for substitution is made by the user, as depicted in Fig. 2.



DATA CLEANING

Here the file will undergo cleaning. Please select the value to replace missing values with:

Fig .2. Cleaning as a part of the transformation process

Pseudocode

```
1.Begin:
2. retrieve choice of user;
3. switch(choice)
    case mean:
        4.1 retrieve result set by executing query
                "select avg(col) from table where
                col!=-1";
        4.2 replace -1 in datatable by mean
                calculated above;
        4.3 break;
    case median:
        4.1 retrieve result set by executing query
                "select col from table where col!=-1";
        4.2 store elements of result set in array
                and sort;
        4.3 calculate median by finding mid point
                of the array;
        4.4 replace -1 in datatable by median
                calculated above;
        4.5 break;
    case mode:
        4.1 retrieve result set by executing query
                "select col from table where col!=-1";
        4.2 store elements of result set in array
                and sort;
        4.3 calculate mode by finding most
                frequently occurring value;
        4.4 replace -1 in data table by mode
                calculated above;
        4.5 break;
5. If more columns exist, goto 4.1 of corresponding
        case;
6. Commit changes;
```

IV. Transformation

A. *Background*

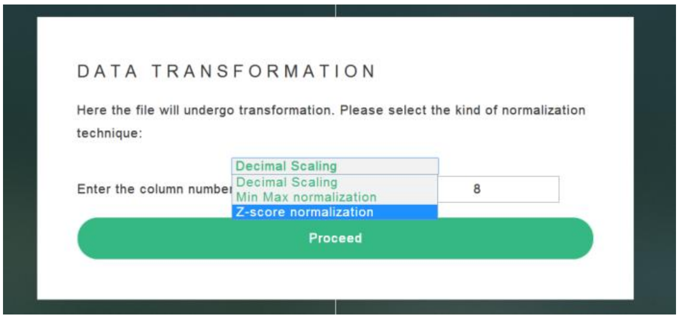
It is often important to able to comprehend the data at a glance. If there is a diverse range of values to be analysed, it often makes the job of analysts more difficult. Also, its highly difficult to observe such a trend without specifying some boundary conditions. This is where transformation comes into play. It helps make sense of these values, while grouping them or normalizing them to fall into a specified range. This establishes a sense of uniformity that is easy to read and convenient to analyze.

Essentially, transformation can fall into one of these categories:
a) Normalization - where data is transformed to a specified range or scale.
b) Data aggregation - where data is converted based on differential levels of hierarchy, into a data cube. This can enable drill down and roll up operations
c) Smoothing - that works on reducing the noise of the data, by performing binning or regression tests.
d) Generalization - where a form of abstraction produces higher level details of lower level classes.
e) Attribute construction - where we generate new attributes from an existing pool of attributes and their corresponding values, to adder newer dimensions to the analysis.

B. *Implementation*

This paper emphasizes on incorporating normalization techniques to perform transformation on the data set. The user must choose a particular column to perform the normalization on. This, unlike cleaning, is not done uniformly on the database because of the fact that there can be primary or foreign keys which must be present to uniquely identify each tuple. Modifying such data can lead to a lot of errors and defeat the purpose of having distinct, irreplaceable data entries. The user is given three choices to perform normalization as shown in Fig .3.

(a) Min - max normalization - where data is scaled based on a newer maximum and minimum. We have assumed a scale of [1-10]
(b) Decimal scaling - where data is scaled to values that do not exceed, and are decimals
(c) Z-score normalization - where data is transformed based on the mean and standard deviation values



DATA TRANSFORMATION

Here the file will undergo transformation. Please select the kind of normalization technique:

Enter the column number 8

Fig .3. Transformation of a specified column

Pseudocode

```
1.Begin:
2. retrieve choice of user;
3. Retrieve specified column number as *col;*
4. switch(choice)
    case *min-max* :
        4.1  retrieve result set by executing query
                 “select *col* from *table* ”;
        4.2  Read result set into array;
        4.3  Sort the array;
        4.4  Find min and max of array;
        4.5  Assume newmin and newmax;
        4.6  For each entry x, perform
                 transformation as
                 $x = ( (x-min)/(max-min) ) *(newmax-newmin) )+newmin$;
        4.7  Update database;
        4.8  break;
    case *decimal-scaling*:
        4.1  retrieve result set by executing query
                 “select *col* from *table*”;
        4.2  store elements of result set in array
                 and sort;
        4.3  Find min and max of array;
        4.4  while(max>=1)
                 4.4.1  max = max/10;
                 4.4.2  den=den*10;
        4.5  For each entry x, perform transformation
                 as $x = x/den$;
        4.6  Update database;
        4.7  break;
    case *z-score*:
        4.1  retrieve result set by executing query
                 “select *col* from *table* ”;
        4.2  Read result set into array;
        4.3  Sort the array;
        4.4  Calculate average as avg;
        4.5  Evaluate standard deviation as
                 deviation;
        4.6  For each entry x, perform
                 transformation as
                 $x = ((x-avg)/deviation)$;
        4.7  Update database;
        4.8  break;
5. *Commit changes;*
```

V. CASE STUDIES AND RESULTS

A) University database

We have considered a university database of United States of America which comprises of around 1000 records that include 35 attributes of both numeric and string data types. It is a database which has few missing values represented by asterisk symbol(*). As our paper deals with how the numeric data is cleaned, the program checks for the data in the corresponding column and substitutes the missing value with the user’s choice of mean, median and mode. However, the string values are just substituted by the previous ones in the column.

This cleaned data is now to be transformed by performing normalization ie., decimal scaling normalization,z-score normalization,and min-max normalization. In case of transformation, user must explicitly specify the column number to be transformed, as it is strictly applied to numerals. Now, this cleaned and transformed data is easy for analysis. This tool is hugely beneficial for students who are trying to judge their choice of universities based on word of mouth. It will give them proper data to be analyzed and the scaling makes it more comprehensible for a common man to understand.

B) Census Income Dataset

It is a social related dataset. It predicts whether the income exceeds $50K per year based on the census data. The dataset is a multivariate type where attribute characteristics are categorical and numeric. It belongs to the classification type of association,where in it has 48842 records with 14 attributes along with few missing values in it. When we ran the code over this data set, we were successfully able to transform the data into a better format, that could be understood by anyone who wanted to get an overview of the current situation of their region. Using this data, predictions can be made by normalizing the attributes for analysts to get a better idea of the social strata.

VI. CONCLUSION AND FUTURE WORK

Most of the unrelated data has been effectively removed from each of the datasets that had been used. The paper explains the incorporation of such a system that is simple and user-friendly. Some of the enhancements to the project that we are currently working on are to incorporate concept hierarchy trees and data cleaning methodologies for textual data. We are also working towards implementing computing data cubes and extend the capabilities of the data pre-processing system to a fully automated analysis system that can mine interesting patterns as well.

REFERENCES

[1] SINTEF. (2013, May 22). Big Data, for better or worse: 90% of world's data generated over last two years. ScienceDaily. Retrieved August 29, 2014. www.sciencedaily.com/releases/2013/05/130522085217.htm

[2] S.Zhang et al, Data Preparation for Data Mining,Applied Artificial Intelligence, 2003- Taylor & Francis,

[3] http://www.cs.ust.hk/~qyang/Docs/2003/Data_Preparation_for_Data_Mining_ZZY.pdf

[4] Erhard Rahm and Hong Hai Do,University of Leipzig, Germany-Data Cleaning and current approaches http://wwwiti.cs.uni-magdeburg.de/iti_db/lehre/dw/paper/data_cleaning.pdf

[5] Jason W. Osborne, Ph.D,North Carolina State University Practical Assessment, Research & Evaluation http://pareonline.net/getvn.asp?v=8&n=6

[6] David C. Howell, Treatment of Missing Data—Part 1