IJCSE International Journal of Computer Sciences and Engineering Open Access
Review Paper Volume-6, Issue-4 E-ISSN: 2347-2693

Tactics for Dynamic Data Cleansing and Data Profiling Using Dimensions for Data Quality Assessment

A. Ghouse Mohiddin[1*] , S. Ramakrishna[2]

[1]Dept. of Computer Science, Dravidian University, Kuppam, A.P., India
[2] Dept. of Computer Science, Sri Venkateswara University, Tirupathi, A.P., India

**Corresponding Author: agm_ghouse@yahoo.co.in*

Available online at: www.ijcseonline.org

Received: 30/Mar/2018, Revised: 09/Apr/2018, Accepted: 25/Apr/2018, Published: 30/Apr/2018

ABSTRACT- We classify data quality problems that are directed by data cleaning and provide an overview of the principal Solution approaches.Data cleansing is particularly needed when integrating heterogeneous data sources and Should be directed together with schema-related data transformations. We also discuss current tool support for data cleanup. Data profiling is a specific form of data analysis customer data to detect and characterize important features of data sets. Data Analysis offers a delineation of data structure, content, rules and relationships by using statistical methodologies to deliver a lot of standard characteristics about data -data types, field lengths and cardinality of columns, granularity, value sets, format patterns, content patterns, implied rules, and cross-column and cross-file data relationships and cardinality of those relationships. Data deduplication has been advocated as a promising and effective technique to save the digital space by removing the duplicated data from the data centres or clouds. Data deduplication is a process of identifying the redundancy in data and then removing it. The resulting unique data/Consolidate data into single format using data cleansing and Data standardization. Use scorecards to measure data quality progress and shared URL link to the stakeholder.

Keywords: Data Analysis, Data Profiling, Data Cleansing, Data Standardization, Data Score Cards.

I. INTRODUCTION

Data profiling is a specific form of data analysis customer data to detect and characterize important features of data sets. Data analysis provides a delineation of data structure, content, rules and relationships by using statistical methodologies to deliver a lot of standard characteristics about data -data types, field lengths and cardinality of columns, granularity, value sets, format patterns, content patterns, implied rules, and cross-column and cross-file data relationships and cardinality of those relationships [2]. It deals with detecting and removing errors and inconsistencies of data in order to ameliorate the quality of information.. Data Investigation or Data Profiling using multiple information sources need to be mixed data, e.g., in data warehouses, global web-based information systems, the need for data cleaning increases To provide an accurate and consistent of the customer data, consolidation of different data representations and elimination of duplicate data. [3]**.**

Data Investigation/Analysis results can be compared with documented expectations, or they can provide a foundation on which to build knowledge about the data. It is used for purposes of data discovery and in order to prepare data for storage and use, data profiling can take place at any point in a data asset's lifecycle. It will discuss periodic data analysis and cleansing of the data environment as one of the three assessment scenarios supported by the ***Data Quality Assessment Framework*** (DQAF). It measure data quality and verify that changes to the data meet, an effect of data profiling content through column profile. Data deduplication has been advocated as a promising and effective technique to save the digital space by removing the duplicated data from the data centres or clouds. Data deduplication is a process of identifying the redundancy in data and then removing data [2]. The resulting unique data/Consolidate data into single format using data cleansing and Data standardization. A scorecard is the graphical representation of the valid values for a column or output of a rule in profile results. Use scorecards to measure data quality progress and shared URL link to the stakeholder.

In Section 2 of this paper, we discuss the data analysis and profiling in Customer Relationship Management (CRM) system. We present Statement of problem-Hypotheses-Customer data validation using Data cleaning and Data Standardization in Section 3. We present Effectiveness of DQ Scorecards can be easily shared with Stakeholders via a URL and measure data quality progress in Section 4. Section 5 we present several considerations in the conclusion.

II Data Profiling

Data profiling is a specific form of data analysis customer data to detect and characterize important features of data

sets. Its content different data rules by using statistical methodologies to deliver a lot of standard characteristics from the customer data, data types, field lengths and issue of Data quality[2]. A profile is a set of metadata that describes the content and structure of a dataset. We can run a profile to evaluate the structure of data and verify that data columns are populated with the types of information we expect.

It measure data quality and to verify that changes to the customer data i.e.Firstname, last name, Email validation, Phone number validation, country code validation. We can run a column profile on a transformation in a mapping to indicate the effect that the transformation will have on data profiling also includes inspection of data content through a column profile or percentage distribution of values. Understanding the percentages is useful, particularly for high-cardinality value sets and for data sets with a large number of records.

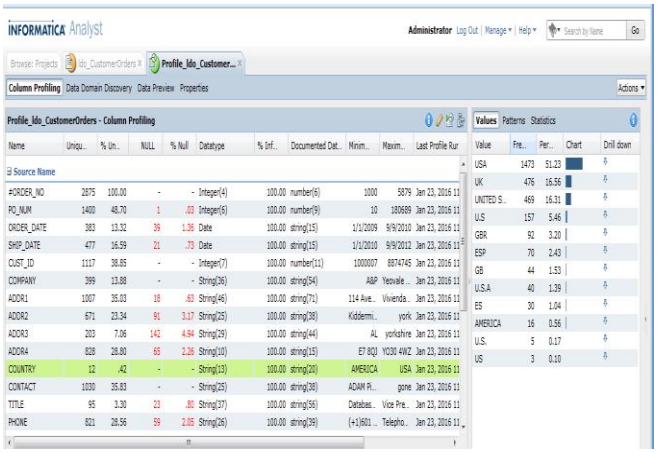

Figure 2.1.1: Data profiling data issue.

Data Investigation/Analysis results can be compared with documented expectations, or they can provide a foundation on which to build knowledge about the data.

2.1 Column Profile

The number of unique and null values in each column, expressed as a number and a percent. The patterns of data in each column and the frequencies with which these values occur. Statistics about the column values, such as the maximum and minimum lengths of values and the first and last values in each column. We can add a rule to the profile to cleanse, change, or validate data. Create scorecards to periodically review data quality.

2.2 Join Profile /Join Analysis

The Join analysis describes the degree of potential joins between two data columns. It use joins between multiple data sources [2]. A join profile displays results as a Venn diagram and as numerical and percentage values. By using a profile model to perform a join analysis on a pair of columns. We can analyse join conditions on columns from one or more data objects, and we can define more than one join analysis in a profile. A join analysis uses Venn diagrams to show the relationships between columns[2].

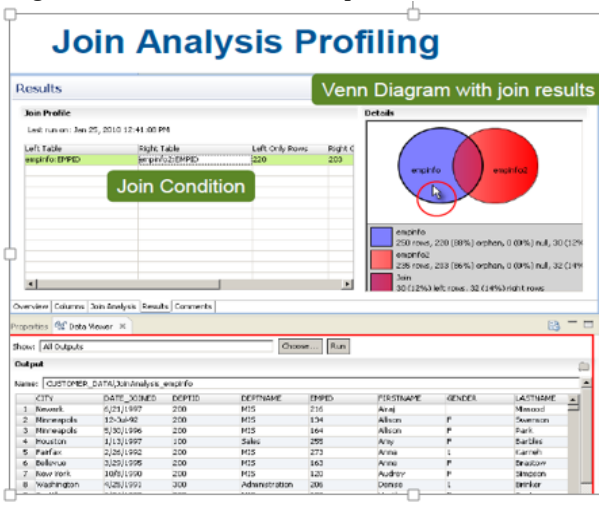

Figure 2.2: Result of Join Analysis Profiling

2.3 Rule Based Profiling

A Rule is a constraint written against data that is used to identify possible inconsistencies in the data. Rule creation and editing (Expression based). Leveraging OOTB Rules / Developer created rules. Apply rules within profiles and analyze results in-line with original source data [2].

2.3.1 Value Frequency Rules

Select the value frequency results to include in the Rule, right click and choose Add Rule, choose to create a Value Frequency Rule. The expression is written based on business data rules, it can be reusable. After running the profile click on the new frequency rule created

2.4 Mid-Stream Profiling

Mid-stream Data profiling at any point within a mapping, targets cannot be profiled.

2.5 Profile Results

We can view the profile results after we run a profile based on Business Rule/Data Rules. We can view a summary, values, patterns, and statistics for columns and rules in the profile. We can view properties for the columns and rules in the profile. We can preview profile data [2].

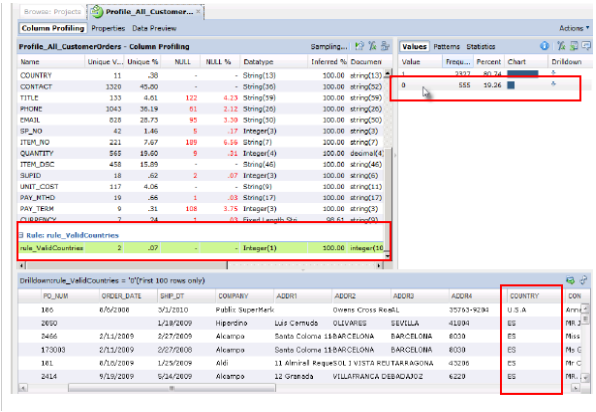

Figure 2.5: Profile Results

III. STATEMENT OF PROBLEM-HYPOTHESES-CUSTOMER DATA VALIDATION

In this section, data quality problems that are addressed by data cleaning and data standardization 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 in data warehouses, data cleaning is a major element of the ETL process. We also discuss current tool support for Data cleansing, support for data cleanup. Standardization addresses the data quality issues identified through data profiling to transform and parse data from single multi-token fields to multiple fields to correct completeness, conformity, and consistency problems and to standardize the field through data rule, data tokenization process, Regular expression.

3.1 Data cleaning

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. A data cleaning should find and remove all major faults and inconsistencies both in individual data sources. Data cleaning should not be done in isolation, but together with schema-related data transformations based on comprehensive metadata. The major data quality problems to be puzzled out by data cleaning and information translation [9, 10].

3.2 Data Standardization

The Data Standardizer is standardizes characters and strings in data. It can be used to remove noise from a field. It is a passive transformation an input strings and creates standardized versions of those strings.. Standardization addresses the data quality issues identified through data profiling [9]. The key objectives in data standardization are.

- To transform and parse data from single multi-token fields to multiple fields.
- To correct completeness, conformity, and consistency problems.
- To standardize field formats and extract important data from free text fields [9].

The customer Data standardized to examine a column of address information that contains the Strings Street, St., and STR. Each strategy can contain multiple standardization operations. The Standardizer transformation creates columns that contain standardized versions of input strings. The transformation can replace or remove strings in the input data when creating these columns. The verify the a column of address data that contains the strings Street, St., and STR. AVE. or AVE or AVNUE to AVENUE etc., [10] The labeler transformation is a passive transformation that examines input fields and creates labels that describe the type of characters or strings in each field.

- Ex: # $ %^ & as symbol or S
- (or) 17242 as 9999

3.2.1 Standardization Strategies / Properties

Use standardization strategies to create columns with standardized versions of input strings. When you configure a standardization strategy, you add one or more operations. Each operation implements a specific standardization task. We can add the following types of operations to a standardization strategy [10].
To configure properties for standardization strategies and operations, select the Strategies view in the Standardizer transformation.

Strategy Properties

Strategy properties apply to all the operations within a strategy. You can configure the following strategy properties

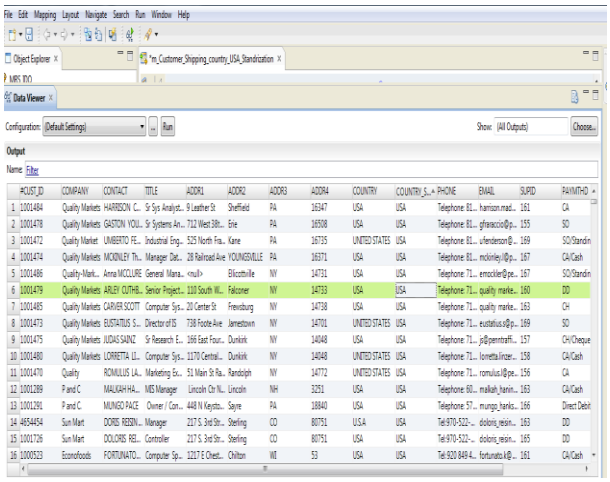

Figure 3.2.1: Output of Standardizer

3.2.2 Probilistic Modelling Technique using Tokens and Parser

3.2.2.1 Character Labelling

The character labelling is an identified each character in the customer input data, for example the Labeller transformation can label the ZIP Code 517299 as "nnnnnn," where "n" stands for a numeric character. It describes the character structure of the input string, including punctuation and spaces. The transformation writes a single label for each row in a column. We can use a labeller transformation to understand the types of information that a port contains. By using a Labeller transformation when we do not know the types of information on a port, or when we want to identify records that do not contain the expected types of information on a port. A label is a string one or more characters that describes an input string [11]. We configure the labeller transformation to assign labels to input strings based on the data that each string contain.

3.2.3. Probabilistic Models

It identifies tokens by the types of information they contain and by their positions in an input string. We can use probabilistic models with the labeller and Parser transformations. A probabilistic model using to label or parse values on an input port into separate output ports. [11] A probabilistic model uses a structured set of tokens as a reference data set and labelling or parsing operation. An input column that represents the data on the input port. We populate the column with sample information from the input port. We can add the pillars to the data model and we assign labels to the data tokenized process in each string. By using the label columns to indicate the correct position of the tokens in the string.

When we configure a token labelling operation with a probabilistic model, the labeller's transformation writes the column name from the probabilistic model to an output port on the transformation. For example, the labeller can use a probabilistic model to label the string "Franklin Delano Roosevelt" as "FIRSTNAME MIDDLENAME LASTNAME."

When we configure a token parsing operation with a probabilistic model, each column we add to the model becomes an output port on the Parser transformation. The transformation writes each token to an output port based on its position in the model [11].

The probabilistic model does not need to list every token. We update the fuzzy logic rules when we compile the probabilistic model.

3.3.4. Token Parsing Operations

We can add the following types of operations to a token parsing strategy:

Parse using Token Set

Use predefined or user-defined token sets to parse input data. Token set operations can use custom regular expressions that write to one or more outputs.

Tokenized Output Ports

Passes input strings that correspond to each label in the output. Select this port if we will add a Parser transformation downstream of the labeller transformation in a mapplets or mapping, and we will configure the Parser transformation to run in pattern-based parsing mode. The Parser transformation associates the token labelling output with the data on the tokenized output ports [12].

Score Output Ports

The score values generated by probabilistic matching techniques in a token labelling operation. We can run a token labelling operation that uses a probabilistic model, the operation generates a numerical score for each labelled string. [2] The score represents the degree of similarity between the input string and the patterns defined in the probabilistic model**.**

3.3.5. Parser Transformation Modes

A Parser transformation, select either token parsing mode or pattern-based parsing mode. This mode is used to parse input values that equal values in reference data objects such as token sets, regular expressions, probabilistic models, and reference tables. We can use multiple token parsing strategies in a transformation [12].

Pattern-based parsing mode. Use this mode to parse input values that match values in pattern sets. Use the Parser transformation when the data fields in a column contain more than one type of information and we want to move the field values to new columns. The Parser transformation is to create new column for each type of information in a data set, we can create a data structure that parses name data from a single column into multiple columns. For example, first names, middle names, and surnames.

We can configure the transformation with a probabilistic model that represents the structures of the person names on the input port.

Configure the transformation with reference tables that contain recognizable address elements, such as ZIP Codes, state names, and city names. Create a token parsing strategy that writes each address element to a new port. We cannot use a reference table to parse street address data from an input string, because street name and number data is too general to be captured in a reference table. However, we can use the **overflow port** to capture this data. We can have parsed all city, state, and ZIP data from an address, the remaining data contains street information. For example, use a token parsing strategy to split the following address into address elements:

123 MAIN ST NW STE 12 ANYTOWN NY 12345

The parsing strategy can write the address elements to the following columns

Table 3.3.5: The parsing strategy with probabilistic matching

Column Name	Data
Overflow	123 MAIN ST NW STE 12
City	ANYTOWN
State	NY
ZIP	12345

Create product data columns

We can create a data structure that parses a single column of product data into multiple columns that describe the product inventory details. Configure the transformation with token sets that contain inventory elements, such as dimension, color, and weight. Create a token parsing strategy that writes each inventory element to a new port. For example, use a token parsing strategy to split the following paint description into separate inventory elements.

500ML Red Matt Exterior

The parsing strategy can write the address elements to the following columns.

Table 3.3.5: The parsing strategy with Product elements

Column Name	Data
Size	500ML
Color	Red
Style	Matt
Exterior	Y

3.3.6 Token Parsing Ports

A Parser transformation in token parsing mode has the following port types.

Input Ports:

It contains data that we pass to the parser transformation. The transformation merges all input ports into a combined data string using the **input join character** specified on the strategies tab. If we do not specify an input join character, the transformation uses a space character by default.

Parsed Output Ports

User-defined output port(s) that contains successfully parsed strings. In cases where multiple parsing strategies use the same output, the transformation merges the output into a combined data string using the **Output Join Character** specified on the Strategies tab. If we do not specify an output join character, the transformation uses a space character by default.

Overflow

Contains successfully parsed strings that do not fit into the number of outputs defined in the transformation. For example, if the transformation only has two "WORD" outputs, the string "John James Smith" results in an overflow output of "Smith." The Parser transformation creates an overflow port for each strategy that we add, select the **Detailed Overflow** option, the transformation creates an overflow port for each label in the model [12]. Unparsed Contains strings that the transformation cannot parse successfully. The Parser transformation creates an unparsed port for each strategy field.

IV Data Quality Scorecards

It is the graphical representation of the valid values for a pillar or output of a rule in profile results. Use scorecards to measure data quality progress. We can create a scorecard from a profile and monitor the progress of data quality over time. A scorecard is the graphical representation of valid values for a column in a profile [13].

Score card is the graphical representation of valid values for a column in a profile, it can be easily shared with stakeholders via a URL. Scorecards to measure data quality progress, scores based on value frequencies. single scorecard supports scores from multiple data objects [13].

It has multiple parts, such as metrics, metric groups, and doorways. After we run a profile, add source columns as metrics to a scorecard and configure the valid values for the prosody. A threshold identifies the range, in percentage, of bad data that is acceptable for columns in a data. We can set thresholds for good, acceptable, or unacceptable ranges of data [10].

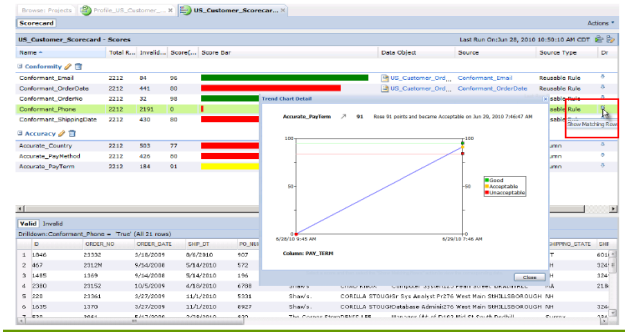

Figure 4.1: Result of scorecard for valid customer data

CONCLUSION

Data Investigation results can be compared with documented expectations, or they can provide a basis on which to establish knowledge around the data. It is used for purposes of data discovery and in order to prepare data for storage and use, data profiling can take place at any point in a data asset's lifecycle. It will discuss periodic data analysis and cleansing of the data environment as one of the three assessment scenarios supported by the Data Quality Assessment Framework (DQAF). Data deduplication has been advocated as a promising and effective technique to save the digital space by removing the duplicated data from the data centers or clouds. Data deduplication is a process of

identifying the redundancy in data and then removing customer data. A set of processes that measure and improve the quality of important data on an ongoing basis, ensures that data dependent business processes and applications deliver expected results. Data Standardization is the problems with the data have been identified, to cleanse the data through standardization process, enrichment and validate the good data. The Address standardization is the data quality issues identified through data profiling to transform and parse data from single fields to multiple fields. Finally data quality is to correct completeness, conformity, and consistency problems, to standardize field through data rule, data tokenization process, Regular expression.

REFERENCES

[1]. Chaudhuri, S., Dayal, U.: An Overview of Data Warehousing and OLAP Technology. ACM SIGMOD Record 26(1), 1997.

[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]. Bouzeghoub, M.; Fabret, F.; Galhardas, H.; Pereira, J; Simon, E.; Matulovic, M.: Data Warehouse Refreshment. In [16]:47-67.

[4]. Abiteboul, S.; Clue, S.; Milo, T.; Mogilevsky, P.; Simeon, J.: Tools for Data Translation and Integration. In [26]:3-8, 1999.

[5]. Lee, M.L.; Lu, H.; Ling, T.W.; Ko, Y.T.: Cleansing Data for Mining and Warehousing. Proc. 10th Intl. Conf.Database and Expert Systems Applications (DEXA), 1999.

[6]. Rundensteiner, E. (ed.): Special Issue on Data Transformation. IEEE Tech. Bull. Data Engineering 22(1), 1999.

[7]. Cohen, W.: Integration of Heterogeneous Databases without Common Domains Using Queries Based Textual Similarity. Proc. ACM SIGMOD Conf. on Data Management, 1998.

[8]. Bernstein, P.A.; Dayal, U.: An Overview of Repository Technology. Proc. 20th VLDB, 1994.

[9]. Quass, D.: A Framework for Research in Data Cleaning. Unpublished Manuscript. Brigham Young Univ., 1999.

[10]. Hernandez, M.A.; Stolfo, S.J.: Real-World Data is Dirty: Data Cleansing and the Merge/Purge Problem. Data Mining and Knowledge discovery 2(1):9-37, 1998.

[11]. Erhard Rahm and H. Hai Do. Data cleaning: Problems and current approaches. IEEE Data Engineering Bulletin, 23(4):3--13, December 2000.

[12]. M.Jayakameswaraiah, Dr.S.Ramakrishna, "A Study on Prediction Performance of some Data Mining Algorithms", International Journal of Engineering & Technology, ISSN: 2321 7782, Volume-2, Issue-10, pp 141-144 (2014).

[13]. K.S.N.Prasad, S.Ramakrishna "Text Analytics to Data Warehousing" (IJCSE) International Journal on Computer Science and Engineering" Vol.02,No.06,2010,PP:2201-2207.

[14]. K.S.N.Prasad,S.Ramakrishna"An Autonomous Forest Fire Detection System Based On Spatial Data Mining and Fuzzy Logic"(IJCSNS) International Journal of Computer Science and Network Security,Vol.8 No.12,December 2000.

Authors Profile

Mr. A.Ghouse Mohiddin Master of Computer Application from M.K.University of Madurai, Tamil Naidu, in year 2003 and Master of Philosophy in Computer Science from Periyar University,Salam,Tamil Naidu,India in year 2008. He is currently pursuing Ph.D. and currently working as Senior Technical Consultant in Capgemini Technology Services India Limited, Bangalore. His main research work focuses on Data warehousing, Data De-duplication and Data Standardization, fuzzy Logic Algorithms, Data Base Management System, Cloud Security and Privacy, Big Data Analytics, Data Mining.

Mr. S.Ramakrishna Master of Science from S.V.University of Tirupathi, A.P. India in year 1983. Doctor of Philosophy from S.V.University of Tirupathi, A.P. India in year 1988. He is currently working as Professor in Department of Computer Science, S.V.University of Tirupati, A.P. India. He has published more than 40 research papers in reputed international journals. He has more than 30 years of teaching experience and more than 10 years of Research Experience.

