Generate Data Cleansing Rules Using Machine Learning

Azimullah shahzad[1], Devashree Date[2], Parag Rane[3], Prasanna S R[4]
Data scientist[1], AI Consulatant[2], Sr. Data scientist[3], A. Director[4]
[1234]Accenture Services Pvt Ltd

Abstract

There is lot of research around data quality but none of them lays down approach to generate data cleansing rules without involvements of business expert. Currently, Data cleaning activities are dependent on manual inspection by business expert. This paper present various techniques to generate business rules for data cleansing activity with minimal involvement of business expert. Next, a system was developed to filter relevant business rules and can be applied to clean up irregularities in data. The results show that the proposed system is superior to the previous human dependent methods in terms of discovering data quality and insights.

1.1 Introduction

In the age where almost every process in various organizations governed by huge volumes of data, be it internal operations, inventory management, pricing, and planning marketing strategies etc., it is more important than ever that the data used for these tasks are as accurate as possible. Research done by Gartner in 2018, found that organizations believe they are losing an average of $15 million per year because of poor data quality. In subsequent research in 2021, they also found out that 70% organizations were planning to improve the data quality by around 60% by employing various metrics to bring down the operational risks and costs. It identifies and discusses critical data quality dimensions in organization such as data completeness, consistency, accuracy, conformity, similarity and change in schema. To simulate the data quality dimension and business rules, a lighter version of algorithm proposed by Arvid, Quaine, Anja & Jorge (2013) DUCC Algorithm (Scalable Discovery of Unique Column Combinations), Apriori Algorithm, Basic pattern matching, FD_mine (Discovering Functional Dependencies in a Database Using Equivalences) proposed by Hong, Howard & Cory (1999) in their paper were implemented to suggest business rules for each data quality dimension.

Data quality issue are faced by large enterprises, where received data is consolidated and aggregated from multiple sources and stored. High data quality is most important element in managing processing data within an organization. Possessing high data quality helps an organization to formulate better business strategy and unveil business pattern for fast decision making based on right information. Data with quality issues in organization have brought various issues such as false decision due to incorrect data, high cost of operation and lack of customer satisfaction was inspired by research paper published by D. Strong, Y. Lee & R. Wang (1997). Moreover, the explosive growth of data resulting from modernizing cloud data & analytics is only making the problem worse. The increasing numbers of data available today with unknown quality levels, adding to it challenges to optimally analyse and make use of data that are relevant to the organization. However, in current situation the organization lacks documentation from existing systems, but they are starting a new project that requires data to be in good quality.

How is High data quality defined ?
A data that is fit for use and able to meet the purposed set by data user .This definition clearly suggested that quality of data is highly dependent to the context of data usage and relevant to the customer needs, ability to use and access data. Thus, in data quality assessment and improvement process, participation of data users and stakeholder are important as described in paper by Felix(2014) , Susan (2018) & Manasi (2021)

For structured data, the following techniques can be implemented to assess the quality of the data before using them further tasks such as statistical model building etc.

- Data profiling
- Schema detection and schema matching

1.2 Data Profiling

Data profiling is process of exploring and identifying statistics, information, and metadata from available stored in data base. While the importance of data profiling is extremely high, however efficient, and effective data profiling task is extremely challenging and has no established technique.

Among the general results are statistics, such as the number of missing or null values and unique values in a column, its data type, or the most occurring patterns of its values. Metadata that are more difficult to compute usually involve multiple attributes, such as inclusion dependencies or functional dependencies. More advanced techniques detect approximate properties or conditional dependency properties of the data in question.

Data profiling can be utilized at different stages of data extraction, transformation, and governance. Thus, in our opinion, profiling is the process of verifying structured data, semi-structured data, and unstructured data, gathering data structure, data pattern, statistical information and reviewing data attributes for data governance, data management, data migration, data reporting and data quality control. As systematic data profiling, we would focus on data quality dimension Conformity, Consistency, Integrity, Uniqueness and generates human readable business rules.

Certain element of data profiling can be obtained by SQL queries and by manual inspection by business expertise gained by their experiences. A common issue is that the business has not documented these rules or people with knowledge are busy or even left the company without leaving behind the existing business rules. This paper demonstrates the use of Machine learning techniques to perform data profiling, insight generation and cleansing.

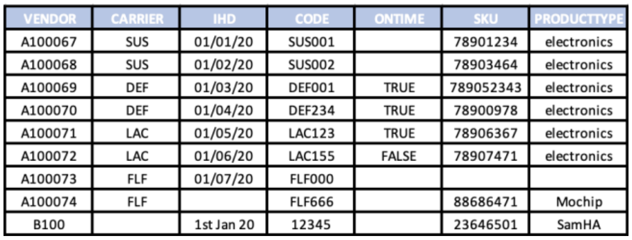
TABLE 1: DATA QUALITY DIMENSION AND ASSOCIATED RULES DESCRIPTION

Dimension	Rules description
Conformity	Finding all the patterns existing in a field with occurrence more than a pre-defined threshold
Conformity	Identify patterns in the field that occur only when a set of fields have certain values.
Conformity	A field has values which are fully or partially derived from values in another field within the table
Conformity	Identify the datatype of the values within the field e.g., string, datetime etc
Conformity	Identifies range value within a column.
Conformity	Identifies length value within a column.
Consistency	Identify if a field is mandatory only when a set of values are populated in one or more fields.
Consistency	Identify if a field is only allowed to have a set of values when a set of other fields have certain values.
Integrity	Identifies how active is the attribute in a transactional data
Similarity	Removal of instances meaning same thing
Deduplication	Removal of duplicated values
Uniqueness	A combination of fields is unique

Conformity :

It is one of the important data quality dimension which means data value of an attributes specified format and data types.

TABLE 2: SAMPLE DATA SET



VENDOR	CARRIER	IHD	CODE	ONTIME	SKU	PRODUCTTYPE
A100067	SUS	01/01/20	SUS001		78901234	electronics
A100068	SUS	01/02/20	SUS002		78903464	electronics
A100069	DEF	01/03/20	DEF001	TRUE	789052343	electronics
A100070	DEF	01/04/20	DEF234	TRUE	78900978	electronics
A100071	LAC	01/05/20	LAC123	TRUE	78906367	electronics
A100072	LAC	01/06/20	LAC155	FALSE	78907471	electronics
A100073	FLF	01/07/20	FLF000			
A100074	FLF		FLF666		88686471	Mochip
B100		1st Jan 20	12345		23646501	SamHA

Given a relation R with schema S (the set of n attributes), each attributes from sets are explored independently at for level 1 format rules with 'Basic pattern matching' algorithm and apriori algorithm for mining association rules.

VENDOR attribute values follows {CNNNNNN} format. The first letter should be Character, followed by 6 numeric value. All entries which doesn't follow the pattern generated by rules are erroneous data, hence either needs to be deleted or updated with right format. The relevance of rules generated are decided by confidence score ,support score and lift value. If we have any domain specific data, enterprise will always have prior information of exact format for attributes. Prior information on format can be used for comparison on data generated format . It would also generate various hidden insights on the attributes.

VENDOR attribute value follows {A10000}NN format. The first 6 letter should always be A10000, followed by 2 numeric values.

Apart from format , it also identifies various attributes statistic. For example Weight attribute ranges from 20 grams to 45 grams (not shown in data set). Such rules straightaway confirms if the weights are within range and if not, then it gives business insight to look back for issue and update data to improve data quality.

Data patterns for combination of attributes are time consuming exercise, hence they are either limited by specifying threshold or supplying predefined sets of attributes or values to algorithm (BPA, Apriori, Market basket analysis) for rules generation. This reduces execution time and filters out most irrelevant rules.

It identifies all the patterns in format which are conditionally dependent over another attributes.

These metrics can be defined as follows:

Support (%) = (Number of records in the field without null or blank)/ (Total Number of records in the data) *100

Confidence (%) = (Number of records in the field matching the pattern)/ (Number of records without null or blank) *100

TABLE 3: RULES GENERATED UNDER CONFORMITY

Dimension	Rules	Support	Confidence
Conformity	Attribute VENDOR should follow {CNNNNNN}	100%	89%
Conformity	Attribute VENDOR should follow {A10000}NN	100%	89%
Conformity	Attribute Weight range between 20-45grams	100%	91%
Conformity	Attribute CARRIER should follow {CCC}	89%	100%
Conformity	Attribute CODE is of string data type	100%	100%
Conformity	Prefield CARRIER can be derived from Postfiled CODE follows CCC*	89%	100%
Conformity	VAT Registration Number should be in XX99999999X format for Country UK	100%	100%

Consistency:

Consistency means data are consistent across all systems reflects the same information and are in synch with each other across the enterprise. When data resides in multiple source or table and store same data in different format, it becomes challenging to maintain consistency across the sources.

It basically involves checking an attribute value conditioned on other business attributes .

Functional dependence among data and attributes were explored using FD_mine algorithm. FD_mine is valuable because it drastically reduces the size of the dataset or the number of checks required, without lead to any loss of information or eliminate any valid candidates.

TABLE 4: SAMPLE DATA SET TO DEMONSTRATE FD_MINE WORKING CONCEPT

Row_Id	A	B	C	D	E	F
T1	0	0	0	1	0	7
T2	0	1	0	1	0	7
T3	0	2	0	1	2	7
T4	0	3	1	1	0	7
T5	4	1	1	2	4	5
T6	4	3	1	2	2	5
T7	0	0	1	1	0	3

Step1:
Find groups of values in every set of attribute combination. Example A's group would be (T1,T2,T3,T4,T7) and (T5,T6) having same values. This clustering is equivalent to D's.
Which implies a FD in *A→D and D→A , A←→ D.*

Step 2:
During Step1 count for each distinct value and keep track of differences in the count across fields. (for example : A and F), this will be used for confidence calculation.

Step 3: Using Armstrong's Axioms to explore more:

If X is a set of Attributes, Y is a subset of X then X→Y holds
If X→Y holds, and P is a set of attributes then PX→PY holds.
If X→Y holds, and Y→P holds, then X→P also holds.
If X→Y holds and X→P holds, then X→PY also holds.
If X→Y holds, and PY→L holds, then XP→L holds.

On the dataset in our case , sample of rules generated under consistency dimension are as follows:

TABLE 5: RULES GENERATED UNDER CONSISTENCY DIMENSION

Dimension	Rules	Confidence
Consistency	If SKU starts with *7890* then PRODUCTTYPE is *electronics*	98%
Consistency	If SKU starts with *7890* then VENDOR is ABC	100%

SKU, PRODUCTTYPE, VENDOR is the attributes, and *7890, electronics, ABC are attributes values.*

Similarity:
Similar data quality issues arise in the context source of truth (Master Data Management). The goal of an MDM system is to maintain a unified view of non-transactional data entities (e.g., customers, products) of an enterprise. These master databases often grow through incremental updates or batch insertion of new entities on every new entry. However, enterprise system receive data from different source or third party. The same information is represented in different way for each associated data source.

TABLE 6: EXAMPLE DATA

Source A	Source B
Cable tele services	Cable telev. services

To identify if both the entries mean the same, Jaccard similarity index with custom weights was implanted.
Sets of 3-grams for 'Cable tele services',
s1 = {Cab, abl, ble, le1, e1t, 1te, tel, ele, le1, e1s, 1se, ser, erv, rvi, vic, ice, ces}
n(s1) = 17
Sets of 3-grams for 'Cable television services'
s2={Cab,abl,ble,le1,e1t,1te,tel,ele,lev,ev.,v.1,.1s,1se,ser,ser,erv,rvi ,vic,ice,ces}
n(s1) = 19

cardinality of intersection:
n(overlap(s2, s1))= 15
overlapA(s1) =88%
overlapB(s2) = 78%
Strings are similar when overlap percentage are higher than defined threshold.
overlapA, overlap >= 75%, we conclude they are similar for large strings.
This resulted in higher accuracy than cosine similarity.

For cases where we had single strings, cosine similarity and phonetic approach based on Soundex concept performed better and was time efficient.

This helped in maintaining consistent data ,and reduction of data error.

Uniqueness:
One important task of data profiling is to discover unique attributes combinations and non-unique attributes combinations. A unique is a set of attributes whose projection has no duplicates in the data sets. Knowing all unique and non-unique data values helps understand the structure and the properties of the data .Unique and non-unique are useful in several areas of data management, such as anomaly detection, data integration, data modelling, duplicate detection, indexing, and query optimization. For instance, in databases, unique are primary key candidates. Furthermore, newly discovered uniqueness constraints information can be re-used in other profiling fields, such as to identify functional dependency detection or new foreign key detection.
The number of possible attribute combinations to be analyzed is exponential in the number of attributes. For instance, a brute-force approach would have to enumerate (294 – 1) attributes combinations to find all unique and non-unique in a dataset with 94 attributes.
Customised DUCC algorithm, proposed by Arvid & Jorge (2013) in their paper, a highly scalable and efficient approach to find (non-)unique column combinations in very large datasets implemented in R language was developed and implemented.

Steps:
1.Create a list of all possible two field combinations from the input field list, Also calculate percentage of distinctness % (PD) for each combination as
PD(A&B) = PD(A) + PD(B) – PD(A)*PD(B).
Order this list with descending order of PD(A&B). This list is called the SEED.

2.Select field combination from the seed with highest PD(A&B), calculate the actual Distinctness for the field combination. One possible way is Count (Distinct (Concatenate (A&B))) / Total Count * 100.

1. If found 100 % unique then find all possible superset combinations of these fields with other fields, remove them from SEED (if found) and place them in a list called TU (Trivial Unique).
2. If found non-unique, then remove all subset combinations of these fields from SEED (if found), and calculate the predicted distinctness (PD) for all possible three field combinations with these fields and add them to SEED. Use the real distinctness of A&B for this calculation.
3. Select the field combination from SEED with highest PD. Repeat step 2 and 3 till no more SEEDs are left.

TABLE 7: RULES GENERATED UNDER UNIQUENESS DIMENSION

Dimension	**Rules**	**Confidence**
Uniqueness	Attribute CARRIER is unique	50%
Uniqueness	Combination of attribute CARRIER&CODE is unique	100%
Uniqueness	Combination of attribute CARRIER&ONTIME is unique	50%

1.3 Schema matching with statistical or machine learning techniques

Along with finding patterns in a single dataset, it is also important to find the fields in different datasets that represent the similar information. Such tasks are needed to be carried out to facilitate the data integration between various datasets.

There are heuristic matching algorithms that use distance-based similarity metric to identify the matching pairs of columns in two or more datasets.

There also are machine learning based approaches that consider the schema matching problem as a classification problem. Here, a matching candidate, which is a pair of fields from two schemas is to be classified as True if they match and False if they don’t.

In this paper, we apply a distance-based method to find matching elements or fields from the two schemas.

We tested the Levenstein distance-based approach on the text columns from each schema. Conformity methods described in the previous section were used to identify the text columns from the schema. Also, the fields that are found in the analysis containing identity related information like Personnel ID, Account ID etc. are converted to strings if they are numeric. We create pairs of columns from the two schema, such that there is one column from each schema and Levenstein distance is calculated between the columns. Column pairs with the metric value greater than a pre-defined threshold are further explored by the domain expert. For numeric type of columns, we used Euclidian distance to identify the column matches

This approach was tested on datasets in retail and manufacturing domain. In both the cases, transaction datasets we looked for matches between transaction datasets and MDM datasets. It was found that there were promising results on both the data while matching the string columns, with retail data having 85% F1 score and the manufacturing data having 87% F1 score at the threshold of 0.7. The distance-based approach did not work effectively when comparing numeric columns with Euclidian distance and was discarded.

Rodrigues and Silva (2021), in their paper suggest a ML-based approach to find matching columns in two or more schema. For this paper we restricted to finding the matching pairs in two schemas. We tested the random forest algorithm based approach on the retail and manufacturing schema.

With this, the F1 score was improved to 89% and 90% as the best score for various parameter combinations for retail and manufacturing schema respectively.

Although, the ML-based method performed better, it was also computationally expensive as well as time consuming, as there was a need to generate labelled data from training the models. If faster results are required for initial analysis, we can use distance-based heuristic approach rather than training ML models.

2. Conclusion

In this paper, we addressed the problem
potential data quality issues by comparing the records and frequency of occurrence, support as well cleanse these issues using machine learning techniques.
This improves data reporting and business decision and assists in data governance. Apart from addressing data quality issues, it also provides deeper understanding data insights & potential business rules (based on the existing master & transactional data attributes) that are not properly documented or known by the business. The insights and business rules can be documented and be served as source of truth for future ETL services.

- Quickly produce data insights and suggested business rules based on the existing data attributes for any data object
- Identify issues that cannot easily be identified by the business
- Recommend correct values to identified data quality issues
- Thousands of suggested rules in minutes
- 70% of time reduction in data rule development and profiling
- Savings due to less effort in building data quality and less manual work required, due to the automation of the business rules generation and cleansing recommendations
- Accelerate the source to target mapping for data migration projects

Along with data profiling for one schema, we also touched upon methods to match two schemas. We discussed a heuristic text-similarity based method and a ML-based method and compared the performance of the two. We concluded that, the overall performance of the ML-based algorithms was better that text-similarity based method. However ML algorithms can be used when we are more focused on accuracy. For the initial analysis of schema, text similarity based methods provide faster results.

3. References

Arvid Heise, Jorge-Arnulfo Quiane-Ruiz ,F Ziawasch Abedjan, Anja Jentzsch, Felix Naumann ,‘*Scalable Discovery of Unique Column Combinations’ 2013*

Hong Yao, Howard J.Hamilton, and Cory J.Butz Department of Computer Science, University of Regina *Discovering Functional Dependencies in a Database Using Equivalences’ 1999*

D. M. Strong, Y. W. Lee, and R. Y. Wang, “*Data Quality in Context,” Communications of the ACM, vol. 40, no. 5, pp. 103–110, May 1997.*

Felix Naumann, *Data Profiling Revisited (2014)*

Flach, P. A., and Savnik, I., *Database Dependency Discovery: A Machine Learning Approach. AI Communications, 12(3):139-160 1999.*

R. Wang and D. Strong, "*Beyond accuracy: What data quality means to data consumers," Journal of management information systems, vol. 12, no. 4, pp. 5–33, 1996.*

Susan Moore, *'How to Create a Business Case for Data Quality Improvement' for Gartner, 2018*

Manasi Sakpal, *'How to Improve Your Data Quality' for Gartner, 2021*

Rodrigues, D., Silva, A.d. *A study on machine learning techniques for the schema matching network problem. J Braz Comput Soc 27, 14 (2021)*