Sakdas: A Python Package for Data Profiling and Data Quality Auditing

Sakda Loetpipatwanich
Graduate School of Applied Statistics
National Institute of Development Administration
Bangkok, Thailand
sakda.loe@stu.nida.ac.th

Preecha Vichitthamaros
Graduate School of Applied Statistics
National Institute of Development Administration
Bangkok, Thailand
preecha@as.nida.ac.th

***Abstract*— Data Profiling and data quality management become a more significant part of data engineering, which an essential part of ensuring that the system delivers quality information to users. In the last decade, data quality was considered to need more managing. Especially in the big data era that the data comes from many sources, many data types, and an enormous amount. Thus it makes the managing of data quality is more difficult and complicated. The traditional system was unable to respond as needed. The data quality managing software for big data was developed but often found in a high-priced, difficult to customize as needed, and mostly provide as GUI, which is challenging to integrate with other systems. From this problem, we have developed an opensource package for data quality managing. By using Python programming language, Which is a programming language that is widely used in the scientific and engineering field today. Because it is a programming language that is easy to read syntax, small, and has many additional packages to integrate. The software developed here is called "Sakdas" this package has been divided into three parts. The first part deals with data profiling provide a set of data analyses to generate a data profile, and this profile will help to define the data quality rules. The second part deals with data quality auditing that users can set their own data quality rules for data quality measurement. The final part deals with data visualizing that provides data profiling and data auditing report to improve the data quality. The results of the profiling and auditing services, the user can specify both the form of a report for self-review. Or in the form of JSON for use in post-process automation.**

Keywords—Data Quality Management, Data Profiling, Data Quality Auditing, Python Package, Data Pipeline

I. Introduction

Today, data is like an essential raw material for organizations that want to succeed[1]. Organizations can use this raw material to extract knowledge to support any decision making. It makes a competitive advantage in the business. The use cases can be found in almost every process, such as strategic planning. Especially in the big data era, that has many data, so we have many raw materials. Thus, if an organization can convert the raw data into enormous value information, it means that organizations can make an unfair advantage. To refine the data into valuable information, it moves via the data value chain[2], which consists of three essential processes: data discovery, data integration, and data exploration.

However, An organization does not have a practical assessment and plans to solve problems from the beginning. Using inadequate quality information to support decision-making would cause many adverse effects, such as using inadequate information to do strategic planning. The plan may lead the organizations in the wrong direction and lose a competitive advantage. Statistics showed that the cost of using inadequate quality data in the US business sector up to 600$ billion per year[3] and can cost businesses 20% - 30% of their operating revenue[4].

Data quality is an increasingly important area. Researchers have been researching and developing methods to manage these problems since the 19th century[5, 6]. The framework was developed to deal with poor data quality such as Total Data Quality Management (TDQM), The Data warehouse Quality Methodology (DWQ), Data Quality Assessment (DQA), and Comprehensive methodology for Data Quality management (CDQ)[7, 8]. In this paper, we used the TDQM framework as a guideline for data quality management. This framework is divided into four processes: the determination of data quality requirements, measurement, analysis, and improving data quality[9].

The past decade has seen the rapid development of data quality management tools in many commercial products. These products often have an extensive system, expensive, and difficult to custom data quality rules. Thus small businesses or researchers or even students have a few choices to deal with poor data quality. Especially when dealing with many datasets, it requires much effort to handle the data quality. For example, it starts with loading data into a table for each dataset, then queries the data for each column and for each table to collect information to calculate the data quality metrics—the estimating of queries if we have ten tables and 20 columns for each table. We use five queries to collect the quality information for each column. Thus we have 1,000 queries to collect all data quality information.

From these challenges, We have developed a data quality management package as open source. The package provides the data profiling for basic general measures and data auditing for specific measures[10], In This development, we use the Python programming language, which is a programming language that is widely used in science and engineering, that has a structure that is easy to read the syntax and has many extensions that can be integrated. This package was integrated with well-known packages such as Pandas, NumPy, and matplotlib. This package is divided into three parts. The first part is data profiling. The second part is the data quality auditing. Finally is the data visualization part.

II. Implementation and Architecture

In architectural design, we divided into three parts that consist of the specification of the requirements, data analysis, and the data visualization.

A. Requirements Determination

In this part, Users can configure various parts of their job. It divided into three parts. The first part is various settings for creating data profiles such as file location, and profile name. The second part is the settings for auditing the data, which users can determine the exact characteristics of the data look

978-1-7281-8106-6/20/$31.00 ©2020 IEEE

like, such as what the data pattern must be, and range of value. The final part is about setting the desired result. Users can set to generate the report to review it by himself or results in the form of JSON used for further use in the development of automated systems.

B. Data Analysis

In the area of data analysis, we use Pandas[11] and Numpy[12] as a main data analytic package, It can be divided into two parts, which are data analysis to create a profile and data analysis to check the quality of the data. For the data analysis that is creating a profile, the package provides a set of operations to collect and calculate the quality metrics[13, 14]. The results will show information lo the number of records, amount of data, amount of duplicate data, and the amount of data missing. There is also an analysis drill down to the column level, which shows results as data type, number of unique data, amount of data missing, number of data formats, amount of data with the same value. For data quality auditing, It will analyze as specified by the user at the beginning. The results will show information such as the number of user-defined items, the amount of data that has passed the specified conditions.

C. Data Visualization

In this visualization part, it collects the information from the data analysis part to create a data visualization. The data visualization that we provided using matplotlib[15], which consists of a bar chart, histogram, boxplot chart, and pie chart, which will combine with other information to format as a report. This report is generated as an HTML document, in which users can open this document using a general internet browser, including the ability to change the report format to a PDF document as needed.

III. Data Quality and Statistical Metrics

Data quality metrics can tell us about the data quality of the dataset. We purposely use three dimensions that can be used by considering a single dataset[13]. Completeness is the first dimension of data quality that we proposed. This dimension shows that a dataset is complete to the degree that it contains required attributes and meets the consumer's expectations. For this dimension, the package demonstrates the metrics that consist of a number of records, number of columns, number of blank columns, number and percentage of complete records, and number of primary keys. The second metric, integrity, tells us about the degree to which data conform to the data-relationship rules that are intended to ensure the complete, consistent, and valid presentation of data representing the same concepts. For this dimension, we provide a number of duplicate data, number of records after removing duplicate data, and number and percentage of missing data. The last dimension is consistency, which shows the degree to which data conform to an equivalent set of data, usually a set produced under similar conditions or a set produced by the same process over time. The user can consider this dimension via the length of data and the percentage of the data pattern. We provide the statistical metrics upon the type of data[16]. Both string and integer data types we provide most frequently are data and data patterns. Besides, we provide calculated metrics for integer data type such as min, max, mean, mode, median, variance, standard deviation, first quartile, third quartile, interquartile range, minimum, maximum, negative outlier, and positive outlier.

IV. Availability

Operating system

Linux and OS X

Programing Language

Python 2.7 and 3.7

Dependencies

matplotlib 3.2+, NumPy 1.18+, Pandas 1.0+

Software location

Archive

Name: Sakdas

License: BSD

Publisher: Sakda Loetpipatwanich

Version published: 2.4.8

Date published: 1 May 2020

Code repository

Name: GitHub

Identifier: sakdaloe/sakdas

License: BSD

Date published: 1 May 2020

Language

English

V. DEMONSTRATION

The example starts with a pure Python function to generate a data profile and audit the data quality. First, we import the Pandas and Sakdas package into a script, then create a Pandas data frame; in this example, we use the CSV file from Kaggle.

```
from Sakdas import Sakda as sd
import pandas as pd

df = pd.read_csv("/.../supermarket_sales.csv")

auditing_config = {'audit':{
    'audit_missing_value': False,
    'define_custom_missing_value': ['.','999'],
    'audit_data_pattern':[
                {'column_name':'Gender', 'regex_pattern': '(Female|Male)'},
                {'column_name':'Invoice_ID', 'regex_pattern': '(^[0-9]{3}-[0-9]{2}-[0-9]{4})'}
            ],
    'audit_outlier': False,
    'audit_primary_key': False,
    'audit_data_range' : [{'column_name':'Quantity', 'min': 0, 'max': 100}]
    }
}
sample_supermarket_sales = sd(
    df,'sample_supermarket_sales',
    '/.../Sakdas_Result',
    auditing_config = auditing_config)

```

Fig. 1. Example of data profiling and data auditing configuration

Then we define the three data quality rules as the JSON format. One is a column named “Invoice_ID” that must have a pattern as “XXX-XX-XXXX” by defining as a Regulation Expression language. The second is a column named “Quantity” must have a value between 1 - 100. The final rule is a column named “Gender” should have only “Male” or “Female”. After we have the Pandas data frame and data quality rules, put it into Sakdas function and set output as a report.


```
{
    "audit_result": {
        "file_name": "sample_supermarket_sales",
        "audit_datetime": "11/05/2020 14:12:36",
        "audit_data_range": [
            {
                "column_name": "Quantity",
                "range": {
                    "min": 0,
                    "max": null
                },
                "pass": 1000,
                "not_pass": 0,
                "pass_ratio": 1.0
            }
        ],
        "audit_data_pattern": [
            {
                "regex_pattern": "(Female|Male)",
                "column_name": "Gender",
                "pass": 1000,
                "not_pass": 0,
                "pass_ratio": 1.0
            },
            {
                "regex_pattern": "(^[0-9]{3}-[0-9]{2}-[0-9]{4})",
                "column_name": "Invoice_ID",
                "pass": 1000,
                "not_pass": 0,
                "pass_ratio": 1.0
            }
        ],
        "overall_pass": 1000,
        "overall_pass(%)": 1.0
    }
```


Fig. 2. Date Auditing result in JSON format.

After script executed, result will show in the terminal that consists of a link to open data profile and data quality auditing report and also result as JSON format.

Fig. 3. Data profile and auditing results.

VI. CONCLUSIONS

The using this package demonstrated capability to data quality managing by taking a few efforts, saving time, and also revealed utility in a downstream application, including an in-line data-quality pipeline, data profile, and auditing report. Notably, the data profile can be used as a base of information to create an advanced application. For example, use the data profile to create an automatic data-linked discovery system, or use it as a feature for a personal data-classification model, an automatic data-cleansing system, and an automatic data-product-approving system.

Finally, we have provided the source code of the package, supporting files, installation instructions, and example scripting that are available on the repository.

VII. REFERENCES

[1] Eckerson, W., "Data warehousing special report: Data quality and the bottom line", *Applications Development Trends*, 2002. 1(1): p. 1-9.J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[2] Miller, H. and P. Mork, "From Data to Decisions: A Value Chain for Big Data", *IT Professional*, 2013. 15: p. 57-59.

[3] Orr, K., "Data quality and systems theory", *Commun.ACM*, 1998. 41(2): p. 66–71.

[4] Blog, W., "A Comprehensive List of Big Data Statistics" , *Bringing Big Data to the Enterprise*, 2012.

[5] Donoghue, O., et al., "Modified Early Warning Scorecard: The Role of Data/Information Quality within the Decision Making Process" , 2011

[6] Stausberg, J., D. Nasseh, and M. Nonnemacher, "Measuring Data Quality: A Review of the Literature between 2005 and 2013", *Studies in health technology and informatics*, 2015. 210: p. 712-6.

[7] Wang, R.Y., V.C. Storey, and C.P. Firth, "A Framework for Analysis of Data Quality Research", *IEEE Trans*. on Knowl. and Data Eng., 1995. 7(4): p. 623–640.

[8] Batini, C., et al., "Methodologies for data quality assessment and improvement", *ACM Comput. Surv*., 2009. 41(3): p. Article 16.

[9] Wang, R.Y., "A product perspective on total data quality management", *Commun. ACM*, 1998. 41(2): p. 58–65.

[10] Dungey, S., et al. "A pragmatic approach for measuring data quality in primary care databases", *IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI)*. 2014.

[11] McKinney, W., "pandas: a foundational Python library for data analysis and statistics". *Python for High Performance and Scientific Computing*, 2011. 14(9).

[12] Walt, S.v.d., S.C. Colbert, and G. Varoquaux, "The NumPy array: a structure for efficient numerical computation", *Computing in Science & Engineering*, 2011. 13(2): p. 22-30.

[13] Sebastian-Coleman, L., "Measuring data quality for ongoing improvement: a data quality assessment framework", 2012: Newnes.

[14] Fan, W., "Data Quality: From Theory to Practice", *SIGMOD Rec*., 2015. 44(3): p. 7–18.

[15] Hunter, J.D., "Matplotlib: A 2D graphics environment", *Computing in science & engineering*, 2007. 9(3): p. 90-95.

[16] Berenson, M.L., D.M. Levine, and K.A. Szabat, "Basic business statistics : concepts and applications", 2015.

[17] Hönigl, J. and J. Küng, "Obtaining a data quality index with respect to case bases", *Vietnam J. of Computer Science*, 2015. 2(1): p. 47–56.

[18] Salati, M., et al., "The European thoracic data quality project: An Aggregate Data Quality score to measure the quality of international multi-institutional databases", *Eur J Cardiothorac Surg*, 2016. 49(5): p. 1470-5.

[19] Lawrence, N.D., "Data Readiness Levels", 2017