I.J. Intelligent Systems and Applications, **2022, 1, 42-56**
Published Online February 2022 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijisa.2022.01.04

Data Quality for AI Tool: Exploratory Data Analysis on IBM API

Ankur Jariwala
U & P U. Patel Department of Computer Engineering, Chandubhai S. Patel Institute of Technology, Charotar University of Science And Technology (CHARUSAT), India
E-mail: 18ce032@charusat.edu.in

Aayushi Chaudhari
U & P U. Patel Department of Computer Engineering, Chandubhai S. Patel Institute of Technology, Charotar University of Science And Technology (CHARUSAT), India
E-mail: aayushichaudhari.ce@charusat.ac.in

Chintan Bhatt
U & P U. Patel Department of Computer Engineering, Chandubhai S. Patel Institute of Technology, Charotar University of Science And Technology (CHARUSAT), India
E-mail: chintanbhatt.ce@charusat.ac.in

Dac-Nhuong Le
Faculty of Information Technology, Haiphong University, Haiphong 180000, Vietnam
Email: nhuongld@dhhp.edu.vn

Received: 18 August 2021; Accepted: 13 October 2021; Published: 08 February 2022

Abstract: A huge amount of data is produced in every domain these days. Thus for applying automation on any dataset, the appropriately trained data plays an important role in achieving efficient and accurate results. According to data researchers, data scientists spare 80% of their time in preparing and organizing the data. To overcome this tedious task, IBM Research has developed a Data Quality for AI tool, which has varieties of metrics that can be applied to different datasets (in .csv format) to identify the quality of data. In this paper, we will be representing how the IBM API toolkit will be useful for different variants of datasets and showcase the results for each metrics in graphical form. This paper might be found useful for the readers to understand the working flow of the IBM data purifier tool, thus we have represented the entire flow of how to use IBM data quality for the AI toolkit in the form of architecture.

Index Terms: Data quality, IBM, artificial intelligence.

1. Introduction

These days, Artificial intelligence and big data have become a topic of high priority for various domains such as industries, science, business, and social media throughout the whole world. Developments in such areas are at high pertinence, as new technologies are thoroughly impacting every walk of life and thus they are also impacting constitutional rights. This paper sets out to contribute on how we can cleanse and manage the data by using various data quality parameters provided in the IBM Data Quality Toolkit for identifying the quality of data. Researchers use to waste huge amounts of their time in clarifying the data, instead, they can use the automated IBM tool to improve the quality of the data, which can help users to save the time of researchers. High-quality data helps strategic systems to integrate related data, which can provide a relational view of the organization and its data. Information quality is a fundamental trademark that decides the dependability of decision-making.

The nature of preparing data massively affects the accuracy, precision, and intricacy of machine learning tasks. Data stays powerless to blunders or inconsistencies that might be encountered during the assortment, conglomeration, or annotation stage. This requires profiling and evaluation of data to comprehend its reasonableness for AI undertakings and the inability to do as can result in mistaken analysis and capricious decisions. While analysts and researchers have zeroed in on working on the nature of models, there are restricted endeavors towards further developing the data quality. So, various tools and algorithms can be used to reduce data preparation time. So, in this paper, we are going to represent how IBM Data Quality Toolkit methodically quantifies the nature of information for building AI models. It reduces the human burden for identifying the quality of data using automated APIs. All the visualizations and graphs created in this

exploratory research are not achieved from IBM Toolkit, they are created by the author based on the metrics and their results.

1.1. Data Quality Use Cases and Features

IBM Research has developed a Data Quality for AI Toolkit that is built using novel algorithms which provides a systematic way to remediate and assess data with well-specified APIs. This Toolkit is mainly built to serve different varieties of use cases such as:

- Building supervised classification models
- Providing data quality for application workflows with intuitive mechanisms to take domain inputs
- Working in the presence of strict privacy constraints by data synthesis
- Automatically reporting on the data quality and capturing the lineage for the data

1.2. Features of Data Quality for AI Toolkit

1. Validators: Algorithms that perform data quality assessment and output a data quality score from 0 – 1.
2. Remediator: Algorithms to provide corrective actions to fix the data quality and impact on the data quality score.
3. Constraints: Explicit input provided by domain experts or implicitly derived by analyzing the data characteristics.
4. Data Synthesizer: If data cannot be shared due to strict privacy constraints, it provides a capability to synthesize data by learning constraints from real data so that it mimics real data.
5. Pipeline: Combines validators and remediators with constraints to address a use case or application workflow and outputs an overall data quality score.
6. Data Readiness Report: Automated documentation of changes that record delta changes in quality metrics and data transformations applied.

So to proceed with checking the data quality for building a supervised classification model are available as a trial version on the IBM API Hub. These APIs can be used at step zero of the Artificial Intelligence lifecycle to identify the quality of the dataset. Data can be assessed from different dimensions like challenges based on data distribution, data labels, data profiling, and data cleanliness using various APIs. The results obtained from all the APIs are in the form of standard structure in JSON object format, which can provide us with a data quality score, points to identify the low data quality, and also provides recommendations to improve the data. The data quality score is a real value between 0 to 1, where 1 indicates perfect quality. You can find the proper documentation of every API about how the data quality score is calculated. These APIs can be used to systematically identify and understand data issues and fix them to improve the data set and accelerate to the next steps of the life cycle. So in this paper, we will be focusing on how structured metrics of data quality works.

2. Architecture/Simulation of Data Quality for AI

2.1. Introduction to API and Technology Used

Many AI problems today are being solved by data scientists by writing custom scripts or manual analysis, which is a time-consuming process, and some challenges like identifying label noise, class parity, and class overlap might take more time to develop. Even other challenges are faced by researchers like a large number of metrics to check for, different modalities of data like time-series data and tabular data, which make this problem more complex. Therefore, the field must be automated to consistently evaluate different data patterns, interpret the evaluation, make recommendations, and run the code for these recommendations.

Data scientists or Researchers can be benefited to make appropriate decisions based on the results showcased in the IBM data quality toolkit. As known, pre-processing is an essential task as the quality of trained data would directly impact the accuracy as well as the complexity of ML models. Thus by inputting the quality data into machine learning would streamline the data preparation process and would improvise the overall reliability of Machine learning models.

2.2. What is Data Quality for AI?

Integrated toolkit of Data Quality on AI provides various quality estimation and data profiling metrics to verify the quality of ingested data objectively and systematically. The metrics result in a score that quantifies data problems as a score between 0 and 1, where 1 would indicate that no problems are detected. These metrics are basically for tabular datasets and they accept the input in the form of comma-separated values files.

2.3. Architectural Flow for Accessing Data Quality for AI

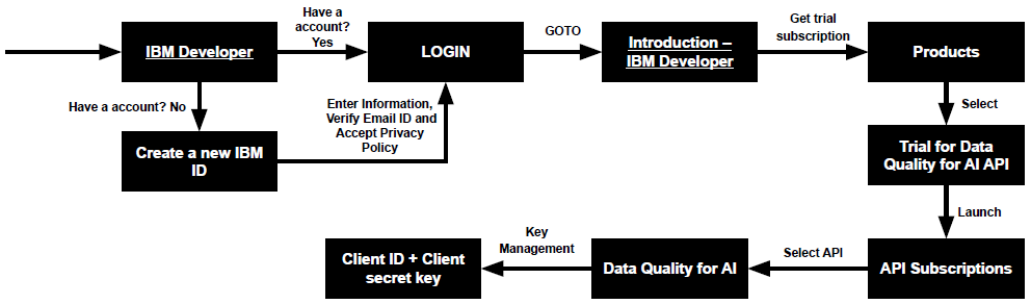

Fig.1. Flow to access IBM API of Data Quality for AI

Step 1: Visit https://www.ibm.com/in-en
Step 2: If not registered, go to create a new IBM ID else log in to the website.
Step 3: Go to Introduction IBM Developer (https://developer.ibm.com/apis/catalog/dataquality4ai--data-quality-for-ai/Introduction/#)
Step 4: Go to Product Catalogs and click on get trial subscription
Step 5: Go to Trial for Data Quality for AI API
Step 6: Launch API Subscriptions
Step 7: Select Data Quality for AI
Step 8: Check Key Management, for Client ID + Client Secret Key

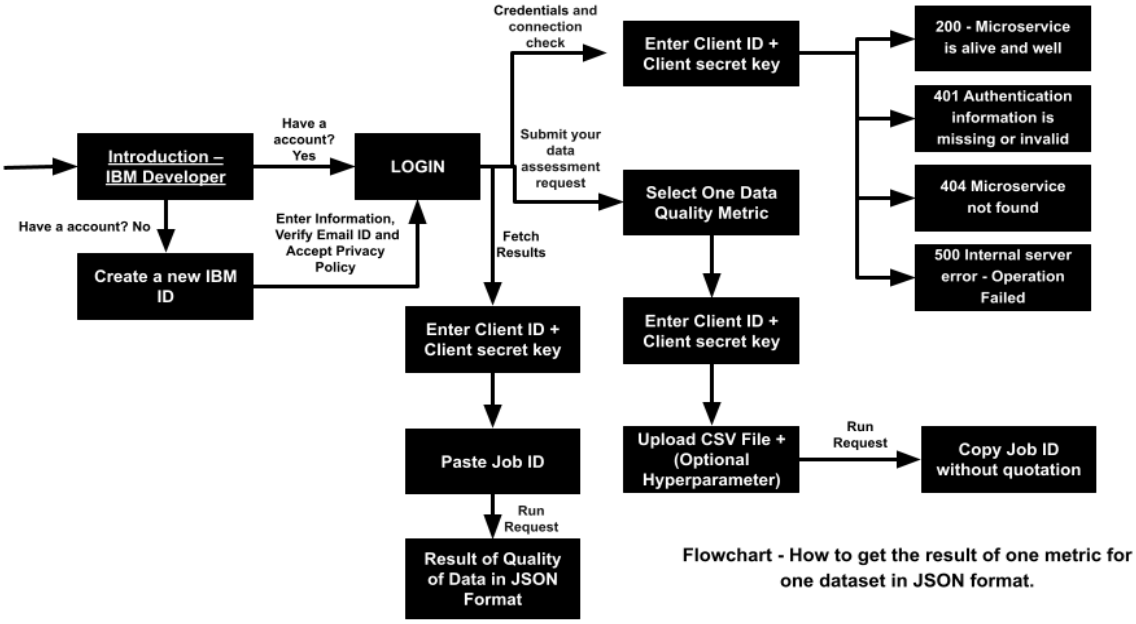

Fig.2. Flow to get the result of one metric for one dataset in JSON format

Step 1: Visit Introduction – IBM Developer (https://developer.ibm.com/apis/catalog/dataquality4ai--data-quality-for-ai/Introduction/#)
Step 2: If not registered, go to create a new IBM ID else log in to the website.
Step 3: For Connection Checking in Data Quality, Enter Client ID and Client Secret Key. These many Responses will come after the run request, 200 - Microservice is alive and well, 401 - Authentication information is missing or invalid, 404 - Microservice not found, and 500 - Internal server error - Operation Failed.
Step 4: For submitting a dataset for assessment request, Enter Client ID and Client Secret Key. Upload CSV File and Enter Hyperparameter (Optional). After the request is run, Get Job ID and copy it.
Step 5: For fetching results, Enter Client ID and Client Secret Key. Paste Copied Job ID. After the request runs, it gets the result of Quality of Data in JSON Format.

3. Result and Observation

3.1. Class Overlap

Definition: Finds class-wise overlapping regions in the data to give an aggregated class overlap score and feature ranges contributing to overlap. A score equal to 1 indicates no class overlap.

Type-quality: Quality

Dataset Type Accepted: Only Supervised Structured Datasets

$$\text{Score: } \frac{Total\ number\ of\ points\ in\ all\ non-overlapping\ regions}{Total\ number\ of\ points\ in\ the\ dataset} \quad (1)$$

Sample Dataset: [20] bill_authentication.csv

- Number of Columns: 5
- Number of Samples/Rows: 1372
- Numerical_columns: ["Variance", "Skewness", "Class", "Entropy", "Curtosis"]
- String_columns: []
- Max_Categorical_Column_String_Length: {}
- "Max_Numerical_Column_Value": {"Class": 1, "Curtosis": 17.9274, "Entropy": 2.4495, "Skewness": 12.9516, "Variance": 6.8248}
- "Min_Categorical_Column_String_Length": {}
- "Min_Numerical_Column_Value": {"Class": 0, "Curtosis": -5.2861, "Entropy": -8.5482, "Skewness": -13.7731, "Variance": -7.0421}
- Unique_Columns: """Class"": {""is_unique"": false, ""num_unique_values"": 2}", "Curtosis": { "is_unique": false, "num_unique_values": 1270 }, "Entropy": { "is_unique": false, "num_unique_values": 1156 }, "Skewness": {"is_unique": false, "num_unique_values": 1256}, "Variance": { "is_unique": false, "num_unique_values": 1338 }
- The accuracy provided by IBM: 1
- Visualization of class overlapping:

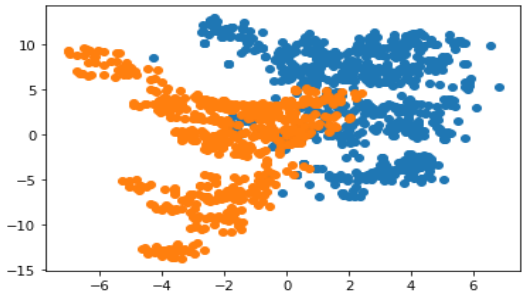

Fig.3. Visualization of Binary Class overlap for Sample Dataset.

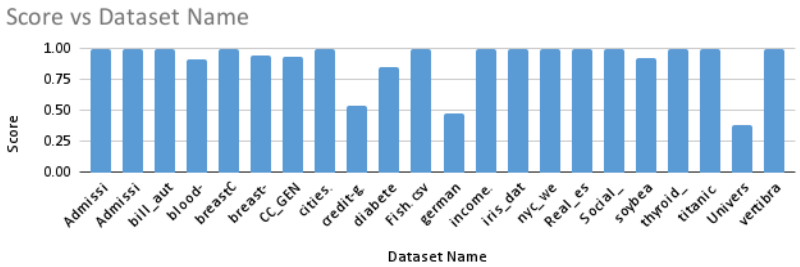

Fig.4. Clustered Column plot of Class Overlap Accuracy for various [26] Datasets

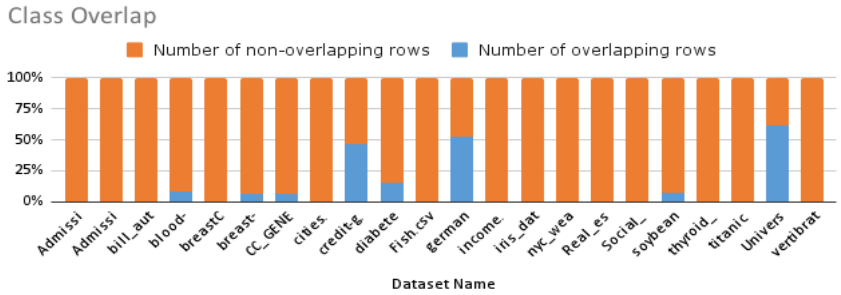

Fig.5. Clustered Column plot of Number of non-overlap rows and overlap rows

3.2. Class Parity

Definition: Identifies noise, overlap, size, and sample counts in the data to give a class parity score. The score is 1 when the ratio of majority class samples to minority class samples is less than 70:30 regardless of any other issue present in the dataset.

Type-quality: Quality

Dataset Type Accepted: Only Supervised Structured Datasets

$$\text{Score: } \frac{Number\ of\ indices\ having\ no\ difficult\ samples\ in\ minority\ class}{Number\ of\ Minority\ class\ items} \tag{2}$$

Sample Dataset: [21] Admission_Predict_Ver1.1.csv

- Number of Columns: 9
- Number of Samples/Rows: 500
- Numerical_columns: ["GRE Score", "CGPA", "Chance of Admit ", "Research", "TOEFL Score", "Serial No.", "LOR ", "University Rating", "SOP"]
- String_columns: []
- Max_Categorical_Column_String_Length: {}
- "Max_Numerical_Column_Value": {"CGPA": 9.92, "Chance of Admit ": 0.97, "GRE Score": 340, "LOR ": 5, "Research": 1, "SOP": 5, "Serial No.": 500, "TOEFL Score": 120, "Uni versity Rating":
- "Min_Categorical_Column_String_Length": {}
- "Min_Numerical_Column_Value": {"CGPA": 6.8, "Chance of Admit ": 0.34, "GRE Score": 290, "LOR ": 1, "Research": 0, "SOP": 1, "Serial No.": 1, "TOEFL Score": 92, "Universi ty Rating": 1}, Unique_Columns: "CGPA": { "is_unique": false, "num_unique_values": 168 }, """Chance of Admit "": {""is_unique"": false, ""num_unique_values"": 60}", """GRE Score"": {""is_unique"": false, ""num_unique_values"": 49}", "LOR ": {"is_unique": false, "num_unique_values": 9}, """Research"": {""is_unique"": false, ""num_unique_values"": 2}", "SOP": { "is_unique": false, "num_unique_values": 9 }, "Serial No.": { "is_unique": true, "num_unique_values": 400 }, "TOEFL Score": { "is_unique": false, "num_unique_values": 29 }, "University Rating": { "is_unique": false, "num_unique_values": 5 }
- The accuracy provided by IBM: 0.04

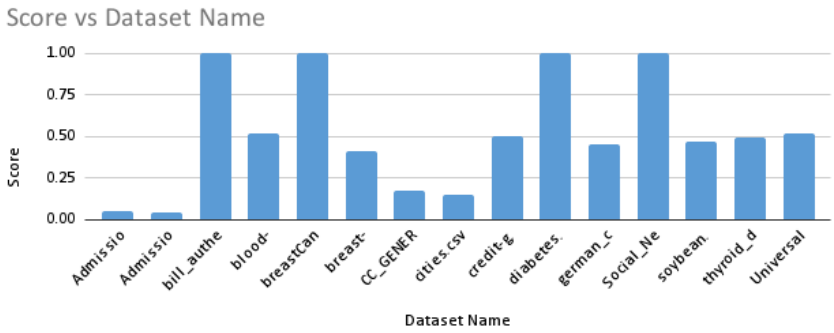

Fig.6. Clustered Column plot of Class Parity Accuracy for various [26] Datasets

3.3. Correlation Detection

Definition: Identifies correlated numerical columns in the data. A score of 1 indicates no correlated columns found in the data.

Type-quality: Quality
Dataset Type Accepted: Unsupervised and Supervised Structured Datasets
Sample Dataset: [22] Fish.csv

- Number of Columns: 7
- Number of Samples/Rows: 159
- Numerical_columns: ["Weight", "Length1", "Length2", "Length3", "Height", "Width"]
- String_columns: [“Species”]
- Max_Categorical_Column_String_Length: {“Species”:9}
- "Max_Numerical_Column_Value": {"Height": 18.957, "Length1": 59, "Length2": 63.4, "Length3": 68, "Weight": 1650, "Width": 8.142}
- "Min_Categorical_Column_String_Length": {“Species”:4}
- "Min_Numerical_Column_Value": {"Height": 1.7284, "Length1": 7.5, "Length2": 8.4, "Length3": 8.8, "Weight": 0, "Width": 1.0476}, Unique_Columns: """Height"": {""is_unique"": false, ""num_unique_values"": 154}", """Length1"": {""is_unique"": false,, ""num_unique_values"": 116}", """Length2"": {""is_unique"": false, ""num_unique_values"": 93}", """Length3"": {""is_unique"": false,. ""num_unique_values"": 124}", """Weight"": {""is_unique"": false,, ""num_unique_values"": 101}", """Width"": {""is_unique"": false,, ""num_unique_values"": 152}", """ ï»¿Species"": {""is_unique"": false, ""num_unique_values"": 7}"
- The accuracy provided by IBM: 0.99

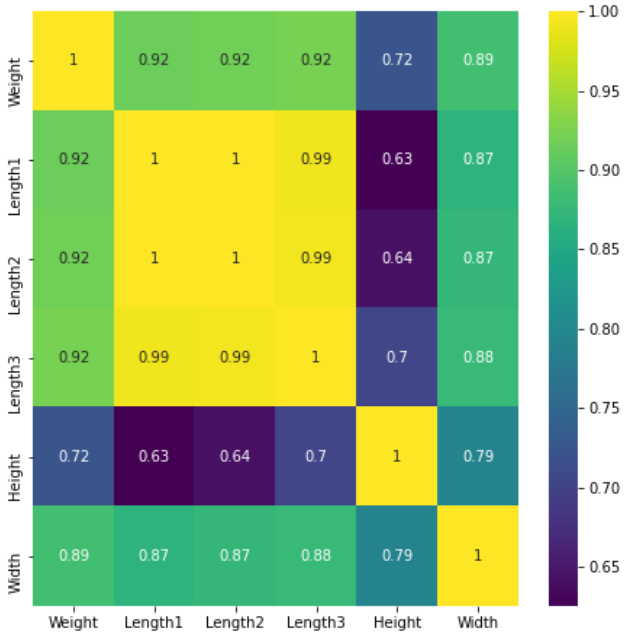

Fig.7. Visualization of Correlation Detection for Sample Dataset

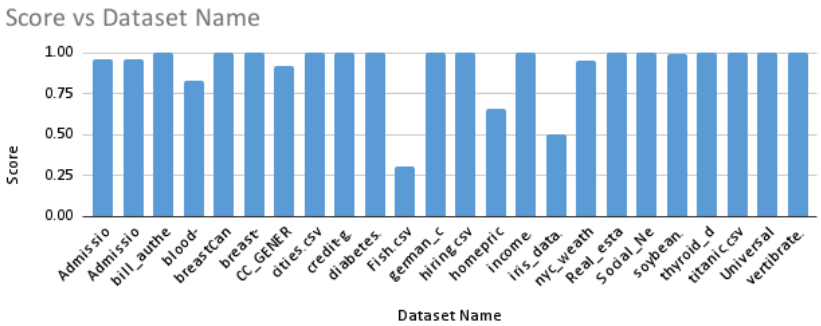

Fig.8. Clustered Column plot of Correlation Detection Accuracy for various [26] Datasets

3.4. Data Completeness

Definition: Identifies missing values in the given data. A score of 1 indicates no missing values found in the data.
Type-quality: Quality
Dataset Type Accepted: Unsupervised and Supervised Structured Datasets

$$\text{Score: } \frac{Number\ of\ Non-Missing\ Value\ detected}{Number\ of\ Given\ Value\ Data} \tag{3}$$

Sample Dataset: [23] titanic.csv

- Number of Columns: 14
- Number of Samples/Rows: 1310
- Numerical_columns: ["survived", "Sibsp", "Parch", "Pclass", "Fare", "Age", "body"]
- String_columns: ["name", "Sex", "Ticket", "Cabin", "Embarked", "Boat", "home.dest"]
- Max_Categorical_Column_String_Length: {"boat": 7, "cabin": 15, "embarked": 1, "home.dest": 50, "name": 82, "sex": 6, "ticket": 18}
- "Max_Numerical_Column_Value": {"age": 80, "body": 328, "fare": 512.3292, "parch": 9, "pclass": 3, "sibsp": 8, "survived": 1}
- "Min_Categorical_Column_String_Length": {"boat": 1, "cabin": 1, "embarked": 1, "home.dest": 5, "name": 12, "sex": 4, "ticket": 3}
- "Min_Numerical_Column_Value": {"age": 0.1667, "body": 1, "fare": 0, "parch": 0, "pclass": 1, "sibsp": 0, "survived": 0}
- Unique_Columns: "age": {"is_unique": false, "num_unique_values": 98}, "boat": {"is_unique": false, "num_unique_values": 27}, "body": {"is_unique": true, "num_unique_values": 121}, "cabin": {"is_unique": false, "num_unique_values": 186}, "embarked": {"is_unique": false, "num_unique_values": 3}, "fare": {"is_unique": false, "num_unique_values": 281}, "home.dest": {"is_unique": false, "num_unique_values": 369}, "name": {"is_unique": false, "num_unique_values": 1307}, "parch": {"is_unique": false, "num_unique_values": 8}, "pclass": {"is_unique": false, "num_unique_values": 3}, "sex": {"is_unique": false, "num_unique_values": 2}, "sibsp": {"is_unique": false, "num_unique_values": 7}, "survived": {"is_unique": false, "num_unique_values": 2}, "ticket": {"is_unique": false, "num_unique_values": 929}
- The accuracy provided by IBM: 0.789040348964013
- Visualization of Data Completeness:

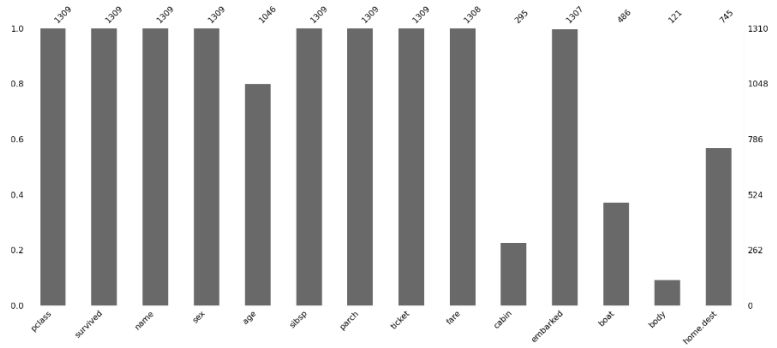

Fig.9. Clustered Column plot of Data Completeness for Sample Dataset's fields

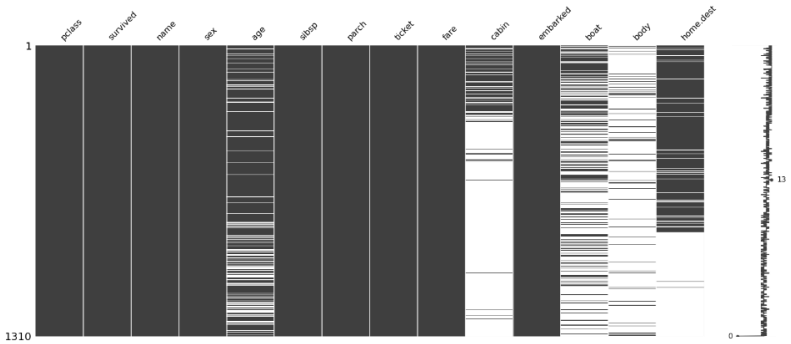

Fig.10. Bar plot of Data Completeness for Sample Dataset's fields

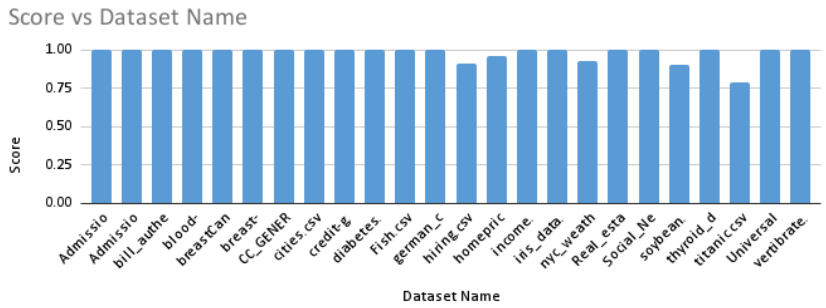

Fig.11. Clustered Column plot of Data Completeness Accuracy for various [26] Datasets

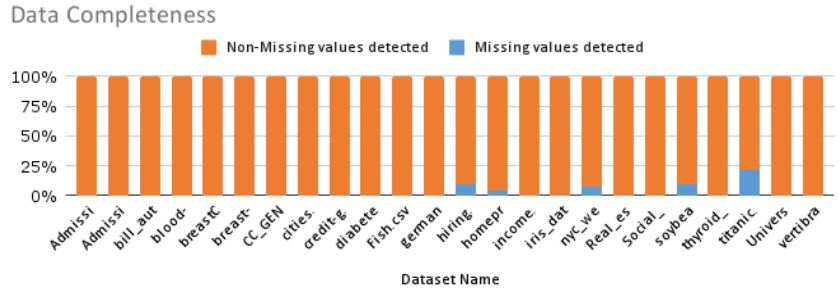

Fig.12. Clustered Column plot of Number of non-missing values and missing values

3.5. Data Duplicates

Definition: Find duplicates in the data using all values in the record. Quality score equals 1 indicates no duplicates found.

Type-quality: Quality

Dataset Type Accepted: Unsupervised and Supervised Structured Datasets

$$\text{Score: } \frac{Number\ of\ Non-Duplicate\ rows\ detected}{Total\ rows} \tag{4}$$

Sample Dataset: [24] blood-transfusion-service-center.csv

- Number of Columns: 5
- Number of Samples/Rows: 748
- Numerical_columns: ["V1", "V3", "Class", "V2", "V4"]
- String_columns: []
- “Max_Categorical_Column_String_Length”: {}
- "Max_Numerical_Column_Value": {"Class": 1, "V1": 74, "V2": 50, "V3": 12500, "V4": 98}
- "Min_Categorical_Column_String_Length": {}
- "Min_Numerical_Column_Value": {"Class": 0, "V1": 0, "V2": 1, "V3": 250, "V4": 2}
- Unique_Columns: """Class"": {""is_unique"": false, ""num_unique_values"": 2}", """V1"": {""is_unique"": false,}, ""num_unique_values"": 31}", "V2": { "is_unique": false, "num_unique_values": 33}, "V3": { "is_unique": false, "num_unique_values": 33}, "V4": { "is_unique": false, "num_unique_values": 78}
- The accuracy provided by IBM: 0.712566844919786

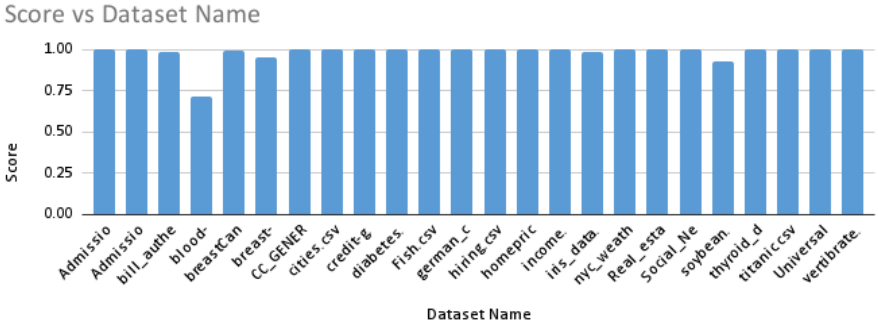

Fig.13. Clustered Column plot of Data Duplicates Accuracy for various [26] Datasets

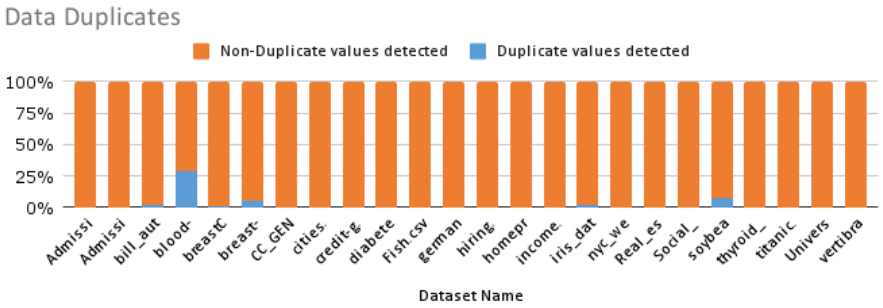

Fig.14. Clustered Column plot of Number of non-Duplicate values and Duplicate values

3.6. Data Homogeneity

Definition: Identifies homogeneity in each column in the data. A score of 1 indicates that no Inhomogeneity is present in the given data.

Type-quality: Quality

Dataset Type Accepted: Unsupervised and Supervised Structured Datasets

Sample Dataset: [23] titanic.csv

- Number of Columns: 14
- Number of Samples/Rows: 1310
- Numerical_columns: ["survived", "Sibsp", "Parch", "Pclass", "Fare", "Age", "body"]
- String_columns: ["name", "Sex", "Ticket", "Cabin", "Embarked", "Boat", "home.dest"]
- Max_Categorical_Column_String_Length: {"boat": 7, "cabin": 15, "embarked": 1, "home.dest": 50, "name": 82, "sex": 6, "ticket": 18}
- "Max_Numerical_Column_Value": {"age": 80, "body": 328, "fare": 512.3292, "parch": 9, "pclass": 3, "sibsp": 8, "survived": 1}
- "Min_Categorical_Column_String_Length": {"boat": 1, "cabin": 1, "embarked": 1, "home.dest": 5, "name": 12, "sex": 4, "ticket": 3}
- "Min_Numerical_Column_Value": {"age": 0.1667, "body": 1, "fare": 0, "parch": 0, "pclass": 1, "sibsp": 0, "survived": 0}
- Unique_Columns: "age": {"is_unique": false, "num_unique_values": 98}, "boat": {"is_unique": false, "num_unique_values": 27}, "body": {"is_unique": true, "num_unique_values": 121}, "cabin": {"is_unique": false, "num_unique_values": 186}, "embarked": {"is_unique": false, "num_unique_values": 3}, "fare": {"is_unique": false, "num_unique_values": 281}, "home.dest": {"is_unique": false, "num_unique_values": 369}, "name": {"is_unique": false, "num_unique_values": 1307}, "parch": {"is_unique": false, "num_unique_values": 8}, "pclass": {"is_unique": false, "num_unique_values": 3}, "sex": {"is_unique": false, "num_unique_values": 2}, "sibsp": {"is_unique": false, "num_unique_values": 7}, "survived": {"is_unique": false, "num_unique_values": 2}, "ticket": {"is_unique": false, "num_unique_values": 929}
- The accuracy provided by IBM: 0.8

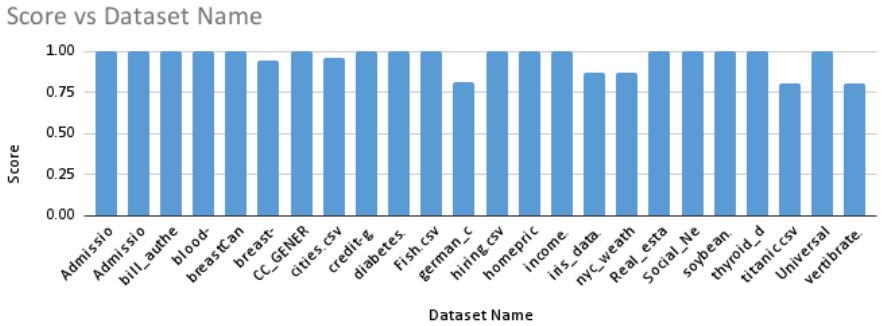

Fig.15. Clustered Column plot of Data Homogeneity Accuracy for various [26] Datasets

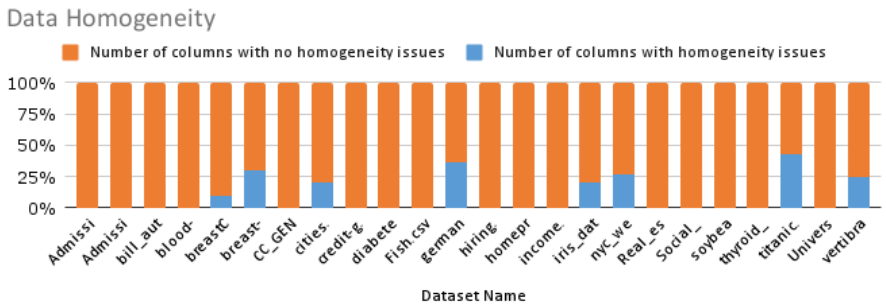

Fig.16. Clustered Column plot of Number of columns with no homogeneity issues and Number of columns with homogeneity issues

3.7. Feature Relevance

Definition: Identifies and ranks the feature based on Relevance. A score of 1 indicates that all features are relevant.
Type-quality: Quality
Dataset Type Accepted: Unsupervised and Supervised Structured Datasets

$$\text{Score: } 1 - \left(\frac{\text{less relevant features}}{Total\ rows}\right) \tag{5}$$

Sample Dataset: [24] blood-transfusion-service-center.csv

- Number of Columns: 5
- Number of Samples/Rows: 748
- Numerical_columns: ["V1", "V3", "Class", "V2", "V4"]
- String_columns: []
- “Max_Categorical_Column_String_Length”: {}
- "Max_Numerical_Column_Value": {"Class": 1, "V1": 74, "V2": 50, "V3": 12500, "V4": 98}
- "Min_Categorical_Column_String_Length": {}
- "Min_Numerical_Column_Value": {"Class": 0, "V1": 0, "V2": 1, "V3": 250, "V4": 2}
- Unique_Columns: """Class"": {""is_unique"": false, ""num_unique_values"": 2}", """V1"": {""is_unique"": false,}, ""num_unique_values"": 31}", "V2": { "is_unique": false, "num_unique_values": 33}, "V3": { "is_unique": false, "num_unique_values": 33}, "V4": { "is_unique": false, "num_unique_values": 78}
- The accuracy provided by IBM: 0.75
- Visualization of Feature Relevance:

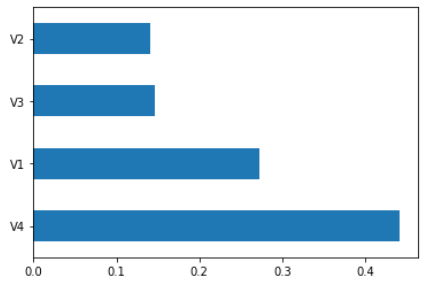

Fig.17. Bar plot of Feature Relevance for Sample Dataset’s fields.

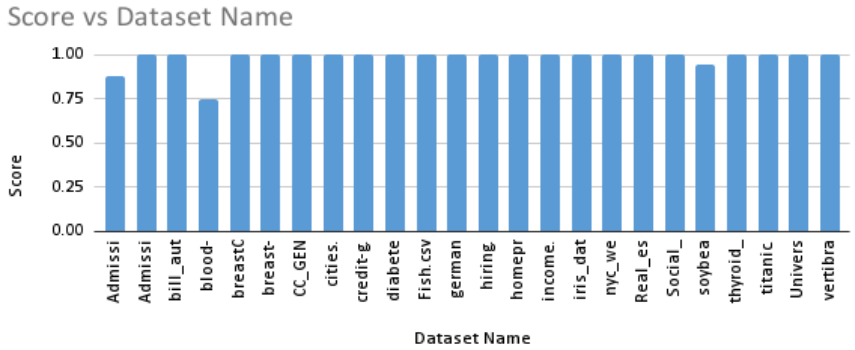

Fig.18. Clustered Column plot of Feature Relevance Accuracy for various [26] Datasets

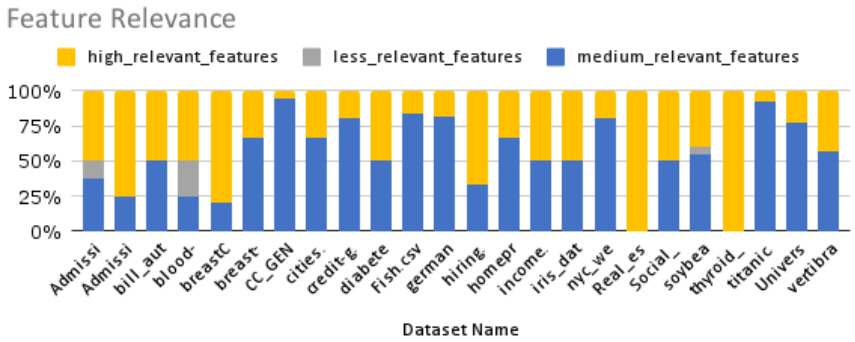

Fig.19. Clustered Column plot of Number of high relevant features, less relevant features, and medium relevant features

3.8. Label Purity

Definition: Identifies noise ratio and noisy samples in the data. A score of 1 indicates no noise is present in the data.
Type-quality: Quality
Dataset Type Accepted: Unsupervised and Supervised Structured Datasets

$$\text{Score: } 1 - \left(\frac{Noisy\ labels}{Total\ labels}\right) \tag{6}$$

Sample Dataset: [24] blood-transfusion-service-center.csv

- Number of Columns: 5
- Number of Samples/Rows: 748
- Numerical_columns: ["V1", "V3", "Class", "V2", "V4"]
- String_columns: []
- “Max_Categorical_Column_String_Length”: {}
- "Max_Numerical_Column_Value": {"Class": 1, "V1": 74, "V2": 50, "V3": 12500, "V4": 98}
- "Min_Categorical_Column_String_Length": {}
- "Min_Numerical_Column_Value": {"Class": 0, "V1": 0, "V2": 1, "V3": 250, "V4": 2}
- Unique_Columns: """Class"": {""is_unique"": false, ""num_unique_values"": 2}", """V1"": {""is_unique"": false,}, ""num_unique_values"": 31}", "V2": { "is_unique": false, "num_unique_values": 33}, "V3": { "is_unique": false, "num_unique_values": 33}, "V4": { "is_unique": false, "num_unique_values": 78}
- The accuracy provided by IBM: 0.962566844919786

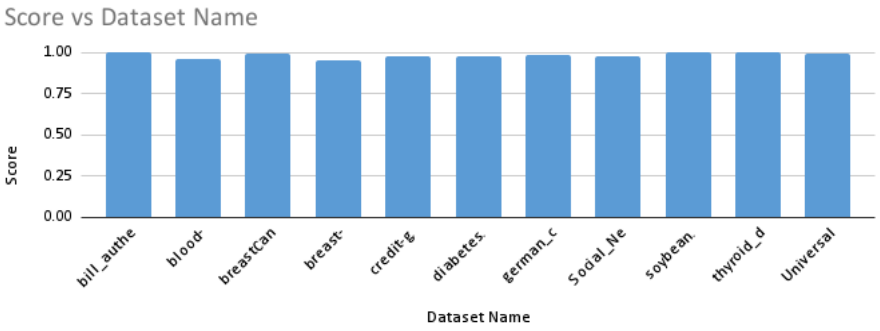

Fig.20. Clustered Column plot of Label Purity Accuracy for various [26] Datasets

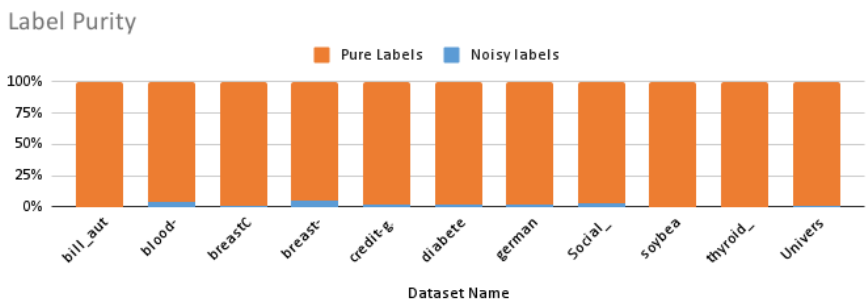

Fig.21 Clustered Column plot of Label Purity Accuracy for various Datasets

3.9. Outlier Detection

Definition: Identifies outlier samples in the data. A score of 1 indicates no outliers found in the data.
Type-quality: Quality
Dataset Type Accepted: Unsupervised and Supervised Structured Datasets
Algorithm Used: LocalOutlierFactor

$$\text{Score: } 1 - \left(\frac{Noisy\ labels}{Total\ labels}\right) \tag{7}$$

Sample Dataset: [25] thyroid_data.csv

- Number of Columns: 6
- Number of Samples/Rows: 215
- Numerical_columns: ["T3_resin", "Serum_thyroxin", "Basal_TSH", "Serum_triiodothyronine", "Abs_diff_TSH", "Outcome"]
- String_columns: []
- Max_Categorical_Column_String_Length: {}
- "Max_Numerical_Column_Value": {"Abs_diff_TSH": 56.3, "Basal_TSH": 56.4, "Outcome": 3, "Serum_thyroxin": 25.3, "Serum_triiodothyronine": 10, "T3_resin": 144}
- "Min_Categorical_Column_String_Length": {}
- "Min_Numerical_Column_Value": {"Abs_diff_TSH": -0.7, "Basal_TSH": 0.1, "Outcome": 1, "Serum_thyroxin": 0.5, "Serum_triiodothyronine": 0.2, "T3_resin": 65}
- Unique_Columns: "Abs_diff_TSH": {"is_unique": false, "num_unique_values": 85}, "Basal_TSH": {"is_unique": false, "num_unique_values": 47}, "Outcome": {"is_unique": false, "num_unique_values": 3}, "Serum_thyroxin": {"is_unique": false, "num_unique_values": 100}, "Serum_triiodothyronine": {"is_unique": false, "num_unique_values": 47}, "T3_resin": {"is_unique": false, "num_unique_values": 55}
- The accuracy provided by IBM: 0.823255814
- Visualization of Outlier Detection:

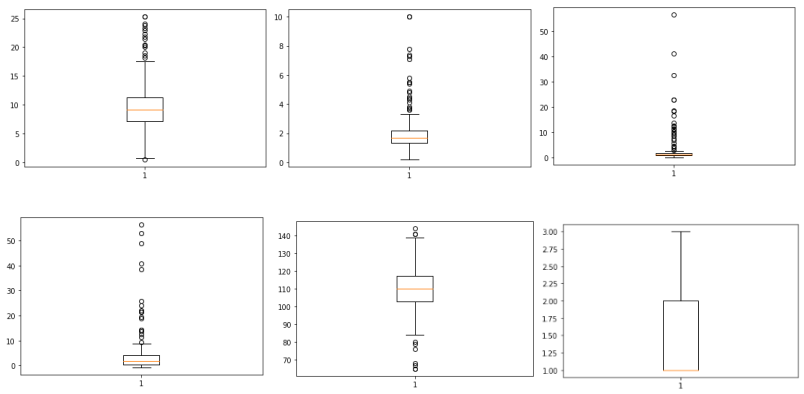

Fig.22. Visualization of Box Plot for Sample Dataset

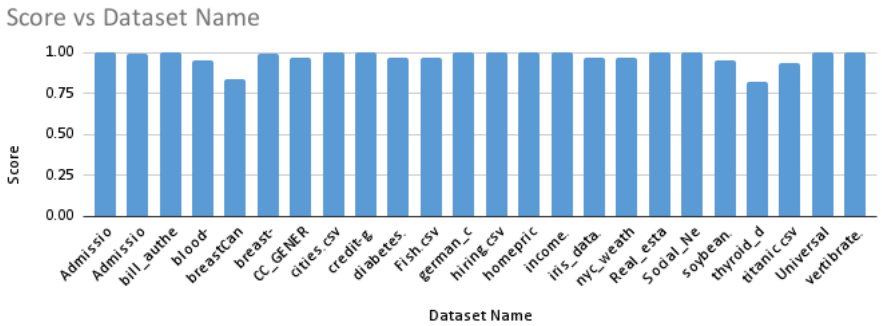

Fig.23. Clustered Column plot of Outlier Detection Accuracy for various [26] Datasets

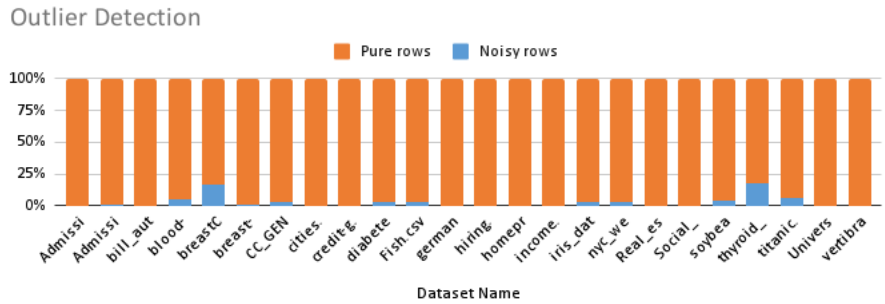

Fig.24. Clustered Column plot of Number of pure rows and Noisy rows

3.10. Data Profiler

Definition: This algorithm allows a user to analyze the data from different angles by providing statistical details for all the columns

Type-quality: Profiling

Dataset Type Accepted: Unsupervised and Supervised Structured Datasets

Sample Dataset: [24] blood-transfusion-service-center.csv

- Number of Columns: 5
- Number of Samples/Rows: 748
- Numerical_columns: ["V1", "V3", "Class", "V2", "V4"]
- String_columns: []
- “Max_Categorical_Column_String_Length”: {}
- "Max_Numerical_Column_Value": {"Class": 1, "V1": 74, "V2": 50, "V3": 12500, "V4": 98}
- "Min_Categorical_Column_String_Length": {}
- "Min_Numerical_Column_Value": {"Class": 0, "V1": 0, "V2": 1, "V3": 250, "V4": 2}
- Unique_Columns: """Class"": {""is_unique"": false, ""num_unique_values"": 2}", """V1"": {""is_unique"": false,}, ""num_unique_values"": 31}", "V2": { "is_unique": false, "num_unique_values": 33}, "V3": { "is_unique": false, "num_unique_values": 33}, "V4": { "is_unique": false, "num_unique_values": 78}

4. Limitation of the API

File Type and Size Limit: The data assessment metrics are suitable for structured/tabular datasets, which can be uploaded in the form of a comma-separated value (CSV) file. Below are some additional points to keep in mind.

- Size of CSV should be <= 15MB.
- We do not store your data, beyond the purpose of data quality analysis. Once the data quality analysis is completed, the data is deleted from our server.

API Call Limit: For the trial version, the following limits apply for usage.

- Daily rate limit - 100 APIs / Day
- Hourly rate limit - 20 APIs / Hour

You can't perform more than one structured metric on one or more datasets.

You can't upload more than one CSV. If you have multiple CSVs of the same dataset, please merge them into one and submit the job.

5. Conclusion and Future Work

In this paper, we have represented automated assessments on various metrics of data quality for AI from IBM which can be used for machine learning, to reduce the data preparation time and improve the training of data quality. The entire flow of accessing the metrics and getting the results is explained in this paper through architecture. Different datasets are experimented on data quality metrics to identify their quality and are represented in the form of graphs.

As future work, we will be reviewing the other metrics of Data Quality for API, whichever would be further added by IBM or else if they bring out any updates in the existing ones, then we will represent them by experimenting using various datasets.

References

[1] Wang, R. Y., Ziad, M., & Lee, Y. W. (2006). *Data quality* (Vol. 23). Springer Science & Business Media.
[2] Zahedi, Z., & Costas, R. (2018). General discussion of data quality challenges in social media metrics: Extensive comparison of four major altmetric data aggregators. *PloS one*, *13*(5), e0197326.
[3] Alves, V. M., Auerbach, S. S., Kleinstreuer, N., Rooney, J. P., Muratov, E. N., Rusyn, I., ... & Schmitt, C. (2021). Curated data in—trustworthy in silico models out: The impact of data quality on the reliability of artificial intelligence models as alternatives to animal testing. *Alternatives to Laboratory Animals*, 02611929211029635.
[4] Elmore, J. G., & Lee, C. I. (2021). Data Quality, Data Sharing, and Moving Artificial Intelligence Forward. *JAMA Network Open*, *4*(8), e2119345-e2119345.
[5] Bertossi, L., & Geerts, F. (2020). Data quality and explainable AI. *Journal of Data and Information Quality (JDIQ)*, *12*(2), 1-9.
[6] Vayghan, J. A., Garfinkle, S. M., Walenta, C., Healy, D. C., & Valentin, Z. (2007). The internal information transformation of IBM. *IBM Systems Journal*, *46*(4), 669-683.
[7] Bisong, E. (2019). Introduction to Scikit-learn. In *Building Machine Learning and Deep Learning Models on Google Cloud Platform* (pp. 215-229). Apress, Berkeley, CA.
[8] Svendsen, S. M. (2021). *In Search of Lost Time: A Deep Dive in Overlapping Computation and Communication in Memory Bound MPI Applications* (Master's thesis).
[9] Shung, K. P. (2018). Accuracy, precision, recall or F1. *Towards data science*.
[10] Torgo, L., & Ribeiro, R. (2009, October). Precision and recall for regression. In *International Conference on Discovery Science* (pp. 332-346). Springer, Berlin, Heidelberg.
[11] Crawford, S. L. (2006). Correlation and regression. *Circulation*, *114(19)*, 2083-2088.
[12] Artasanchez, A., & Joshi, P. (2020). *Artificial Intelligence with Python: Your complete guide to building intelligent apps using Python 3. x*. Packt Publishing Ltd.
[13] Badr, W. (2019). Why Feature Correlation Matters.... A Lot!. *Towards Data Science*.
[14] Santoyo, S. (2017). A brief overview of outlier detection techniques. *Towards data science*.
[15] Reichart, R., & Rappoport, A. (2009, June). The NVI clustering evaluation measure. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL-2009)* (pp. 165-173).
[16] Raschka, S., Julian, D., & Hearty, J. (2016). *Python: deeper insights into machine learning*. Packt Publishing Ltd.
[17] Li, G., Zhou, X., & Cao, L. (2021). Machine learning for databases. *Proc. VLDB Endow*, *14*(12), 3190-3193.
[18] Zhong, S., Zhang, K., Bagheri, M., Burken, J. G., Gu, A., Li, B., ... & Zhang, H. (2021). Machine Learning: New Ideas and Tools in Environmental Science and Engineering. *Environmental Science & Technology*.
[19] Raschka, S. (2015). *Python machine learning*. Packt publishing ltd.
[20] Dataset bill_authenticatśion.csv: https://www.kaggle.com/c178angshumaankesh/bill-authentication?select=bill_authentication.csv
[21] Dataset Admission_Predict_Ver1.1.csv: https://www.kaggle.com/shabiransari/input-admission-predict-ver1-1-csv/data?select=Admission_Predict_Ver1.1.csv
[22] Dataset Fish.csv: https://www.kaggle.com/aungpyaeap/fish-market?select=Fish.csv
[23] Dataset titanic.csv: https://www.kaggle.com/c/titanic/data
[24] Dataset blood-transfusion-service-center.csv: https://www.kaggle.com/ninalabiba/blood-transfusion-dataset?select=transfusion.csv
[25] Dataset thyroid_data.csv: https://www.kaggle.com/dilippuripuri/thyroidcsv?select=thyroid.csv
[26] Other Datasets for Graph Visualizations: https://www.kaggle.com/datasets
[27] Data Quality for AI API - Data Quality for AI API
[28] Data Quality for AI – IBM Developer - Learning Path
[29] Doss, S., Paranthaman, J., Gopalakrishnan, S., Duraisamy, A., Pal, S., Duraisamy, B., ... & Le, D. N. (2021). Memetic Optimization with Cryptographic Encryption for Secure Medical Data Transmission in IoT-Based Distributed Systems. *CMC-COMPUTERS MATERIALS & CONTINUA*, *66*(2), 1577-1594.
[30] Gaur, L., Afaq, A., Solanki, A., Singh, G., Sharma, S., Jhanjhi, N. Z., ... & Le, D. N. (2021). Capitalizing on big data and revolutionary 5G technology: extracting and visualizing ratings and reviews of global chain hotels. *Computers & Electrical Engineering*, *95*, 107374.
[31] Le, D. N., Parvathy, V. S., Gupta, D., Khanna, A., Rodrigues, J. J., & Shankar, K. (2021). IoT enabled depthwise separable convolution neural network with deep support vector machine for COVID-19 diagnosis and classification. *International journal of machine learning and cybernetics*, 1-14.

Authors' Profiles

Ankur Jariwala - Currently pursuing 4th year in Computer Engineering of B.Tech. from Chandubhai S. Patel Institute of Technology, CHARUSAT University, Gujarat. I have done three internships throughout the three years of my Engineering. My research interests are Data Structure and Algorithms, Theory of Computations, Discrete Mathematics, Artificial Intelligence, and Data Science.

Aayushi Chaudhari - Received my Bachelor's Degree of Engineering in the year 2015 and pursued my master's of Computer Engineering in 2017 from Gujarat Technological University.Currently I am pursing Ph.D. from CHARUSAT University along with this, I am holding an academic position as an Assistant Professor Cum Research Fellow, at Chandubhai S. Patel Institute of Technology, CHARUSAT. I have 3 years of teaching experience and industrial experience of 7 months.

Chintan Bhatt is currently working as an Assistant Professor in Computer Engineering department, Chandubhai S. Patel Institute of Technology, Charotar University of Science And Technology (CHARUSAT). He is a member of IEEE, EAI, ACM, CSI, AIRCC and IAENG (International Association of Engineers). His areas of interest include Internet of Things, Data Mining, Networking, Mobile Computing, Big Data and Software Engineering. He has more than 10 years of teaching experience and research experience, having good teaching and research interests. He has more than 70 publications in Internet of Things, Computer Vision and Software Engineering, among which many publications are Scopus indexed. He has been awarded many CSI National Awards and a few CHARUSAT Research Paper Awards.

Dac-Nhuong Le has a M.Sc. and PhD. in computer science from Vietnam National University, Vietnam in 2009 and 2015, respectively. He is Associate Professor, Deputy Head of Faculty of Information Technology, Haiphong University, Haiphong, Vietnam. He has a total academic teaching experience of 15+ years. His researches are in fields of evolutionary multi-objective optimization, network communication and security, VR/AR. He has 80+ publications in the reputed international conferences, journals and book chapter contributions (Indexed by: SCIE, SSCI, ESCI, Scopus, ACM, DBLP). Recently, he has been the technique program committee, the technique reviews, the track chair for international conferences under Springer Series. Presently, he is serving in the editorial board of international journals and 20+ computer science edited/authored books which published by Springer, Wiley, CRC Press.

Further info on his homepage: https://dhhp.edu.vn/nhuongld/.

Scopus: http://www.scopus.com/authid/detail.url?authorId=56438928900

How to cite this paper: Ankur Jariwala, Aayushi Chaudhari, Chintan Bhatt, Dac-Nhuong Le, "Data Quality for AI Tool: Exploratory Data Analysis on IBM API", International Journal of Intelligent Systems and Applications(IJISA), Vol.14, No.1, pp.42-56, 2022. DOI: 10.5815/ijisa.2022.01.04

