Big Data Profiler

Attender Pal Singh and Megha Godwal
Seidenberg School of Computer Science and Information Systems Pace University
New York, USA
fa94822n@pace.edu and mg90740n@pace.edu

***Abstract*— Data profiling application runs on Spark with Mongo dB as the database to extract and store the output of the profiler function. We have created this application in Spark that will read big datasets as input and will provide the profiling information about the dataset as an output. The goal of this project would be to build an API that would take a file as input stored in Mongo DB pointing to the big dataset and run a Spark job to infer all the statistics and then write the output back to a file (.csv or .txt) and store back in the database (Mongo DB). The idea is to make the profiling application available for not only a large variety of datasets but also to a large volume of datasets.**

INTRODUCTION

In today's world working with a big dataset is imperative. Many applications are being developed to operate on these big datasets and do analytics. As many of these data science applications proliferate, the need to profile these big datasets arises. Often, this profiling step is considered to be the first and the most important component in any data science workflow. The goal of the profiling step is to provide useful insights for the input dataset and steer the rest of the workflow using that information.

DATA PROFILING

Data profiling is the process of examining the data available from an existing information source (e.g., a database or a file) and collecting statistics or informative summaries about that data. It is the act of monitoring and cleansing data to make better data decisions. Data profiling helps to discover, understand and organize the data. The data profile serves as a good data inspection tool and ensures that the data is valid and fit for further consumption.

Data profiling utilizes methods of descriptive statistics such as minimum, maximum, mean, mode, percentile, standard deviation, frequency, variation, aggregates such as count and sum, and additional metadata information obtained during data profiling such as data type, length, discrete values, uniqueness, occurrence of null values, typical string patterns, and abstract type recognition.

PROBLEM

For companies, the ever-increasing quantities of data the need to properly manage is only one part of the problem. Data quality is the other.

For example, if you don't correctly format or standardize your data, you could miss sales opportunities. Also, you can make bad business decisions overall. Data processing and analysis cannot happen without data profiling, the need of reviewing source data for content and quality is imperative. Since data gets bigger and infrastructure moves to the cloud, data profiling has become increasingly important. The students who are not logged in will have a certain level of access to view the posted advertisements. The student will be able to view the advertisement details posted by other students.

GOAL

For this project, we have built a profiler application in Spark that reads big datasets as input and will provide the profiling information about the dataset as an output. The data that profiler application is reading is stored in Mongo DB and the output of the profiler function is stored back in the Mongo DB. The goal of this project is to build an API that would take a file as input (i.e., the file (.csv) pointing to the big dataset) and run a Spark job to infer all the statistics and then write the output back to a file (.csv or .txt). The idea is to make the profiling application available for not only a large variety of datasets but also to a large volume of datasets. The benefits of data profiling are to improve data quality, shorten the implementation cycle of major projects, and improve users' understanding of data.

DATA

This is a public dataset hosted by Google Big Query related to COVID-19 for over 20,000 distinct locations around the world. It includes data relating to demographics, economy, epidemiology, geography, health, hospitalizations, mobility, government response, and weather. There are 108 columns and 21,412 rows with various attributes like:

Country Name, Subregion Name, New Confirmed Cases, New Deceased, New Recovered, New Tested, Total

Confirmed, Total Deceased, Total Recovered, Total Tested, New Hospitalized, Total Hospitalized, Population, Female Population, Male Population, Population with respect to age, Latitude, Longitude, Mortality Rate, Nurses, Physicians, Health Expenditure, Temperature, Rainfall, Snowfall, Dew Point, Humidity

MODELLING

This profiler application runs in Spark that reads even big datasets as input and provide the profiling information about the dataset as an output. Spark is reading the data from the Mongo DB database and after running the profiler function it saves the output of the data in Mongo DB database. The application reads the input dataset and provides insights on the dataset that could be used in several other steps of the workflow.

To be concrete, the following information about the dataset can be identified as a part of the profiling step:

- Number of records in the dataset
- Number of attributes (columns) in the dataset
- Total count of the columns excluding the null values if any in the column
- For each attribute:
- Identifying the distinct values
- Identifying the missing or null values
- Identifying number of white spaces or blanks
- Identifying frequency of most and least occurring values
- Identifying Min, Max, Standard Deviation and Mean of columns
- Identifying Min and Max value counts
- Amount of time profiler takes to profile the data

The application reads the csv file in Spark and then the profiler function runs on the dataset that performs all the actions above mentioned. The output mentions the time taken in seconds to run the profiler function and converting the dataset into the desired output which is performing statistical operations on the attributes and converting each column into rows and displaying the statistical operations like finding type of data, number of rows for each column, mean, standard deviation, min, max of all the columns, finding min and max of all value counts, identifying frequency of least and most occurrence values of each column, finding null values etc.

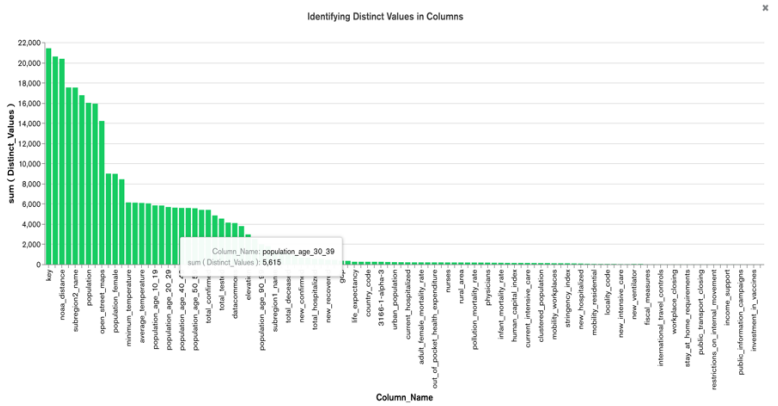
RESULT

It reads the input dataset and provides insights on the dataset that could be used in several other steps of the workflow in Data Sciences or Machine Learning. As the application runs on Spark it provides up to 100 times faster performance for a few applications with in-memory primitives.

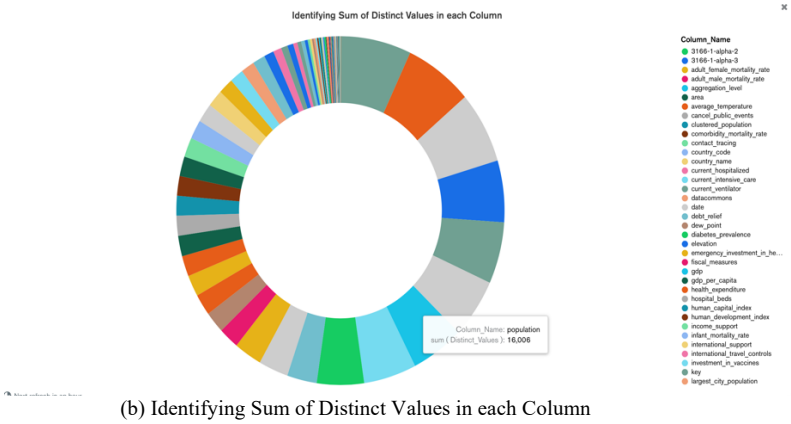
In data science, machine learning is one of the significant elements used to maximize value from data. Therefore, it becomes essential to work on the distribution and statistics of the data to get useful insights. Also, operations on Spark Dataframe run in parallel on different nodes in a cluster, which is not possible with Pandas as it does not support parallel processing.

To store the result of the application we used MongoDB as the database because its horizontal, scale-out architecture can support huge volumes of both data and traffic. And the document data model is a powerful way to store and retrieve data that allows developers to move fast.

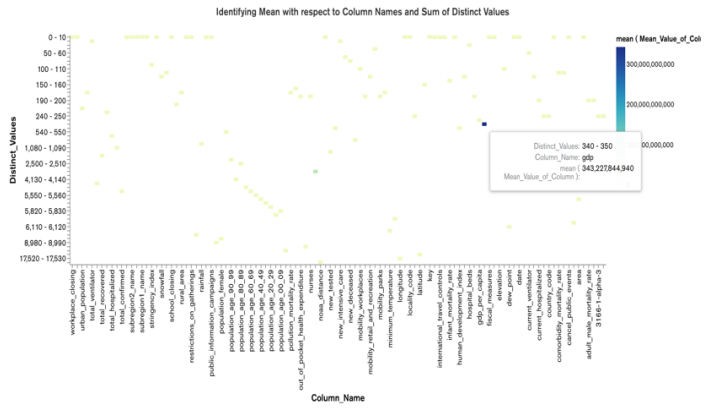
Below are some of the visualization along the columns of our application:



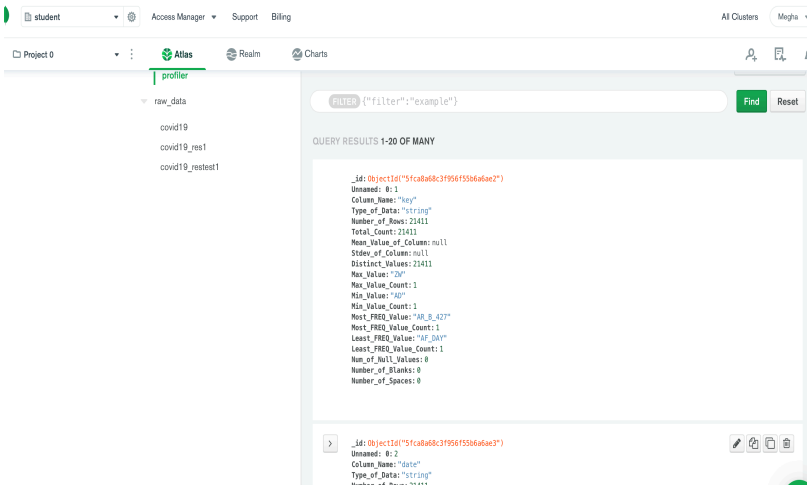
(a)Identifying Distinct Values in Columns



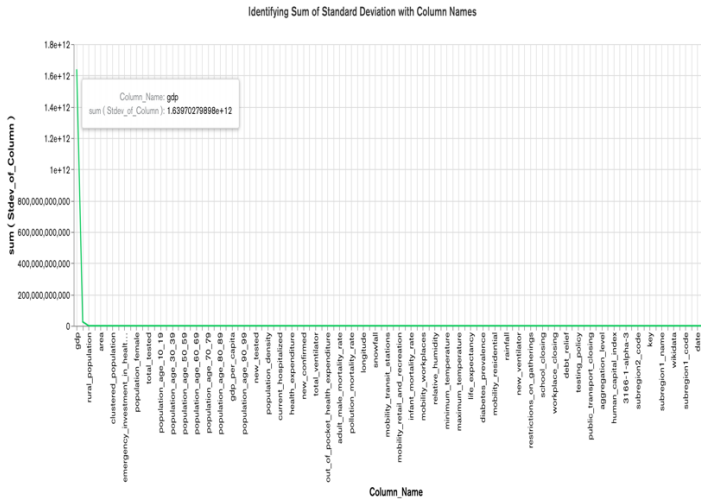
(b) Identifying Sum of Distinct Values in each Column



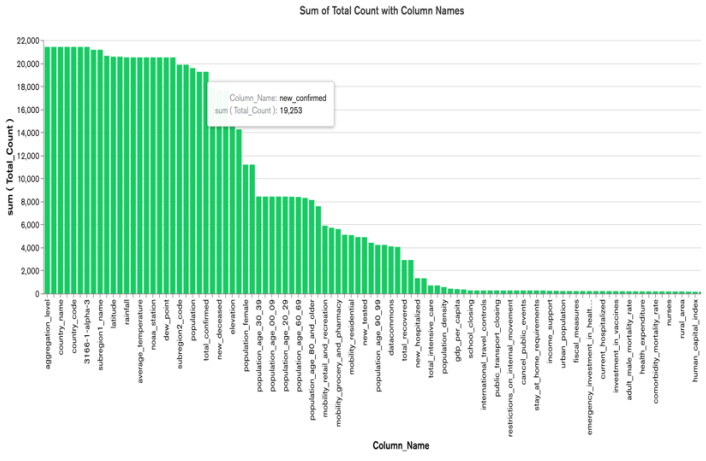
(c) Identifying Mean with respect to Column and Sum of Distinct Values



(f) Output result in Mongo DB



(d) Identifying Sum of Standard Deviation with Columns



(e) Sum of Total Counts with Columns

CONCLUSION

For any kind of analytics workload there is no substitute to knowing the data in and out. Profiling the data should be the first step before using it for any Machine Learning exercise followed by visualizations to better understand the relationships between different data elements and form a hypothesis about how to best capture them with a model. A good understanding of the data can not only help generate valuable insights but also assist with clever feature engineering for robust machine learning models.