INDIUM
Digital | QA | Gaming

Success Story

Data Quality Validation

Application:
Learning Analytics Platform

Services Offered:
Data Quality Validation

Tools:
Python, MongodB

Key Highlights

Domain:
Education Tech

Technology:
Used Python, Java, Scala to write and validate the use cases | Worked with data sources such as NoSQL databases, MongoDB, Flat files, API's, JSON etc.

Client

Client is a leader providing intelligent teaching and learning platform which is built on behavioral analytics.

Application Overview

Application provides analytics on how students interact with test & assignments (Learning Material) provided by various instructors. The product helps instructors to track and evaluate the student's progress.

Our role included writing test cases to verify sanctity of data across several Data Transformation steps including,

- Joining of data sets
- Creation of new derived columns
- Loading of flat files and ingesting data accurately
- Ensuring accurate number of columns post complex join operations

1 Problem Statement

» Today, several Big Data ETL tools are black-box with respect to a GUI
» This makes it challenging to manually assess the data quality after data transformations
» This problem is further complicated when the data pipelines require data to be transformed in real time.
» This created a need to perform automated validations of such data pipelines to ensure data integrity

2 Our Approach

» We first performed a thorough analysis of the work flow & environment that is involved in the data transformation process
» Determined whether the ETL process which is to be replicated, runs in Batch mode or Real time.
» Identified data sources from which data is captured before beginning the ETL process
» Broke down data transformations and data checks into multiple steps that can be checked individually
» Replicated client environment locally
» Implemented data transformation logic on a step by step basis and performed necessary data checks
» Data transformation & business logic was implemented in a different language to that of original implementation to ensure that the business logic was correctly implemented in both the scenarios
» Examples of languages that were re-written while ETL transformation included Scala, Python, Java

3 Testing Use Cases

Exploratory Data Analysis comparison

This included use cases such as comparing the
» Number of records
» Number of missing values
» Number of columns
» Data types of each column
» Accurate column mapping
» Number of unique categories for categorical variables
» Mean, Median, Standard deviation of numeric columns etc. b/w input and output data
» Row-by-row match for non-transformational steps

Process based comparison

This included use cases where the process execution correctness was checked, such as:
» If a table join is performed, validation is done to ensure that the right number of columns are present after the join operation
» If an aggregation is performed, a similar validation is done to find the unique number of groups that exist after the aggregation
» The lower and upper bound for the aggregate values is also defined
» Number of NAs induced after a join operation

4 Solution Deployment

- The use cases written were placed between two ETL steps
- A log file was generated for every validation query run
- All the failed tests were recorded in the log file
- Log file was replaced by real time notifications

INDIUM
Digital | QA | Gaming

INDIA
Chennai
+91 44 6606 9100
Bengaluru
+91 80 4645 7777
Mumbai
+91 022 6215 4028

USA
Cupertino | Princeton
Toll-free: 1 888 207 5969

SINGAPORE
+65 9630 7959

UK
London
+44 773 653 9098

General Inquiries : info@indiumsoftware.com
Sales Inquiries : sales@indiumsoftware.com