Explain Key Errors and Duplicate Rows in fetched Google Universal Analytics reports API

Introduction

This report document is written to explain the duplicate rows that appeared when pulled out and pushed the data to postgreSql. However, before delving into that, we need to

- show and discuss some errors that we faced when fetching the data which leads to create “acrack” table.
- state the meaning of all tables that are created and stored in PostgrSql for this project “Extracting Data from Google Universal Analytics (ga3)”.
- present and analyze some rows that are duplicated from the duplication file.
- detect and how to fix duplicate transactions.

KeyError

In this project, we faced three kinds of keyerrors that caused stopping the running codes. Two of them during run and pull out the data from the User Explorer report with Python (userActivity.search). And the third one, when importing Data from a Pandas to a PostgreSql using SQLAlchemy.

- **KeyError**: 'sessions'

  This key error is appeared two times, on May 03, 2020 and on May 10, 2021 without stating any information to understand what is the reason that caused this error and crushed the running codes on those two days. Spending a lot of time to search for the meaning of this key error, without successed to get any hint. We registered those two days and kept going retrieving the data until facing the second httperror on August 16, 2022. At that time, we understood the meaning of this key error.

  This KeyError is exactly as the second HttpError. But the problem here, we did not know which client_id is not found. The first code is used to retrieve all client_id with the numbers of sessions and the second code is used to read client_id file and to pull out all data from Analytics Reporting API V4 using (userActivity.search).

Conversely, the second code could not find the specific client_id from reporting API using (userActivity.search).

To solve this problem, we tried to divide the day to hours and retrieve the data according to number of hours per day to discover at which hour the code is crushed. However, Google Universal Analytics does not support to retrieve data using specific hours in the RangeDate as Google Analytics 4.

Therefore, we had to divide the client_id file to many files to detect which client_id number that caused to crush the running code until we found them as showing in the following table and deleted them.

- **HttpError**:

<HttpError 400 when requesting https://analyticsreporting.googleapis.com/v4/userActivity:search?quotaUser=my-user-1&alt=json returned "CLIENT_ID: 982610884.1660673523 not found.". Details: "CLIENT_ID: 982610884.1660673523 not found.">

We faced this error on August 16, 18, and 19, 2022. The good thing, it seems google analytics developed its system. The error shows the reason and where is the client_id is not found. Directly, this client_id number is deleted from the client_id file and continue retrieving the data.

- **DataError**

InvalidTextRepresentation: invalid input syntax for type numeric: "7c591c86-bac0-4a91-a3fa-3932644b3ddf"
LINE 1: ...pageview.pagePath", "pageview.pageTitle") VALUES ('7c591c86-...

DataError: (psycopg2.errors.InvalidTextRepresentation) invalid input syntax for type numeric: "7c591c86-bac0-4a91-a3fa-3932644b3ddf"
LINE 1: ...pageview.pagePath", "pageview.pageTitle") VALUES ('7c591c86-...(Background on this error at: https://sqlalche.me/e/20/9h9h) or (Background on this error at: https://sqlalche.me/e/20/f405).

This error is appeared when wanted to importing Data from a Pandas to a PostgreSql using SQLAlchemy. This error as explaining above it shows there is a problem when creating the client_id by Google Analytics software. The client_id should be only numbers as (982610884.1660673523), but the google analytics software created invalid input syntax for type numeric: "7c591c86-bac0-4a91-a3fa-3932644b3ddf".

Thus, we had to create a file called “acrack.csv” contains all invalid input syntax for client _id which were 38 rows, and stored it in a separate table on PostgreSql under the same n ame.

<table>
<tr><th>Date</th><th>Error</th><th>Comment</th></tr>
<tr><td>May 03, 2020</td><td>KeyError: 'sessions'</td><td><table><tr><th>index</th><th>Client_id</th><th>session</th></tr><tr><td>41</td><td>1155862254.1588524236</td><td>1</td></tr></table></td></tr>
<tr><td>May 10, 2021</td><td>KeyError: 'sessions'</td><td><table><tr><th>index</th><th>Client_id</th><th>session</th></tr><tr><td>1173</td><td>977169848.1619928896</td><td>1</td></tr></table></td></tr>
<tr><td>August 16, 2022</td><td>HttpError: <HttpError 400 when requesting https://analyticsreporting.googleapis.com/v4/userActivity:search?quotaUser=my-user-1&alt=json returned "CLIENT_ID: 982610884.1660673523 not found.". Details: "CLIENT_ID: 982610884.1660673523 not found.”></td><td>Details: "CLIENT_ID: 982610884.1660673523 not found.”></td></tr>
<tr><td>August 18, 2022</td><td>HttpError: <HttpError 400 …… Details: "CLIENT_ID: 551831055.1660822532 not found.”></td><td>Details: "CLIENT_ID: 551831055.1660822532 not found.”></td></tr>
<tr><td>August 19, 2022</td><td>HttpError: <HttpError 400 …. Details: "CLIENT_ID: 2128447429.1660888036 not found.”></td><td>Details: "CLIENT_ID: 2128447429.1660888036 not found.”></td></tr>
<tr><td>March 25, 2020.
There are 38 dates under the same error.</td><td>InvalidTextRepresentation: invalid input syntax for type numeric: "7c591c86-bac0-4a91-a3fa-3932644b3ddf"
LINE 1: ...pageview.pagePath", "pageview.pageTitle") VALUES ('7c591c86-...

DataError: (psycopg2.errors.InvalidTextRepresentation) invalid input syntax for type numeric: "7c591c86-bac0-4a91-a3fa-3932644b3ddf"
LINE 1: ...pageview.pagePath", "pageview.pageTitle") VALUES ('7c591c86-...

(Background on this error at: https://sqlalche.me/e/20/9h9h)</td><td>This error is appeared when wanted to importing Data from a Pandas to a PostgreSql using SQLAlchemy.
The problem is Google Analytics software created invalid input syntax for type numeric: "7c591c86-bac0-4a91-a3fa-3932644b3ddf". Which should be numeric as (2128447429.1660888036).</td></tr>
</table>

All Tables that were Stored in PostgrSql

The tables that were created and stored in PostgrSql for this project “Extracting Data from Google Universal Analytics (ga3)”, as following;

SCHEMA : wcd_ga3ua_raw

contains five tables.

First table : stg_user_activity

It represents all visitors that visited weclouddata.ca site without applying any filter. The unique client_id is 305,767 and the number of sessionId is 448,747.

Second table (main): user_activity

It represents all visitors that visited weclouddata.ca site without applying any filter with deleting all duplicated rows. The unique client_id is 304,212 and the number of sessionId is 412,474.

Third table : useractivity_filtersessiongreater2

It represents the real client that visited weclouddata.ca site more than 2 times. The filter is set to be the session greater than 2 times (visiting). Without deleting the duplicated rows. The unique client_id is 15,567 and the number of sessionId is 88,030.

Fourth table : user_activity_duplicateRows

It represents all duplicated rows that deleted from the stg_user_activity table. The purpose of creating this table to be able to analyze those transactions and to figure out what the reasons behind the duplicated those transactions. The number of client_id is 36,273.

Fifth table : acrack

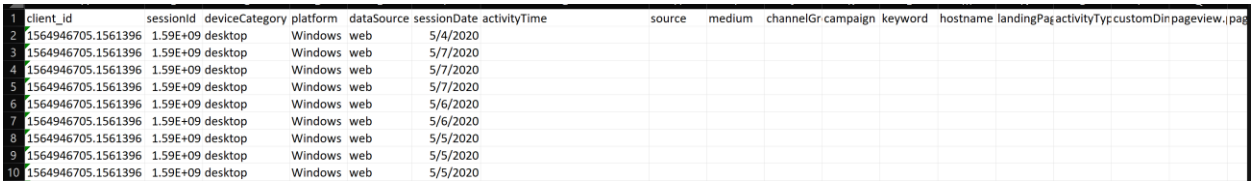
It represents all client_id that have invalid input syntax for type numeric like "7c591c86-bac0-4a91-a3fa-3932644b3ddf" which is completely wrong client_id form.
The unique client_id is 37 and the number of sessionId is 38.

Present and Analyze some Duplicated Rows

Let us discuss some examples from the Fourth table (user_activity_duplicateRows) to understand and analyze the reasons behind causing the duplication. The number of client_id is 36,273.

First example:

Look at this screenshot.



	client_id	sessionId	deviceCategory	platform	dataSource	sessionDate	activityTime	source	medium	channelGr	campaign	keyword	hostname	landingPa	activityTyp	customDin	pageview.	pag
2	1564946705.1561396	1.59E+09	desktop	Windows	web	5/4/2020												
3	1564946705.1561396	1.59E+09	desktop	Windows	web	5/7/2020												
4	1564946705.1561396	1.59E+09	desktop	Windows	web	5/7/2020												
5	1564946705.1561396	1.59E+09	desktop	Windows	web	5/7/2020												
6	1564946705.1561396	1.59E+09	desktop	Windows	web	5/6/2020												
7	1564946705.1561396	1.59E+09	desktop	Windows	web	5/6/2020												
8	1564946705.1561396	1.59E+09	desktop	Windows	web	5/5/2020												
9	1564946705.1561396	1.59E+09	desktop	Windows	web	5/5/2020												
10	1564946705.1561396	1.59E+09	desktop	Windows	web	5/5/2020												

This client_id: 1564946705.1561396 is duplicated 9 times in May, 2020. There are data in those columns ['client_id', 'sessionId', 'deviceCategory', 'platform', 'dataSource', 'sessionDate']; whereas those columns ['activityTime', 'source', 'medium', 'channelGrouping', 'campaign', 'keyword', 'hostname', 'landingPagePath', 'activityType', 'customDimension', 'pageview.pagePath', 'pageview.pageTitle', 'event.eventCategory', 'event.eventAction', 'event.eventLabel', 'event.eventCount'] are completely empty.

Then what's happened, this client for 4 days tried to access to the website, but he\she could not. Maybe there was a problem with his\her device.

Is that a special problem for this customer?

Actually, it is not. There are 14,016 rows like this. Around half of them are from the year 2020.

Second example:

Look at this screenshot.

21	1079946310.1591482	1.59E+09	mobile	Android	web	6/10/2020 2020-06-10T21:17:19.501916Z	google	cpc	Display	wcd_pt_machine learning	(content targeting)	weclouddata/		
22	1079946310.1591482	1.59E+09	mobile	Android	web	6/6/2020 2020-06-06T22:31:52.736382Z	google	cpc	Display	wcd_pt_machine learning	(content targeting)	weclouddata/		

Display	wcd_pt_machine learning	(content targeting)	weclouddata/	PAGEVIEW	[{'index': 1}]	/	Best Data Science and AI Courses in Canada | WeCloudData
Display	wcd_pt_machine learning	(content targeting)	weclouddata/	PAGEVIEW	[{'index': 1}]	/	Best Data Science and AI Courses in Canada | WeCloudData

This client_id: 1079946310.1591482 is duplicated 2 times in June, 2020. It seems as normal transactions.

Then What could cause duplicate rows in fetched Google Analytics reports API ?

In the following subject, we will discuss the most important reasons that cause duplicated rows.

Detect Duplicated Transactions

Duplicate transactions in Google Analytics means a single transaction was counted more than once. This can dramatically skew the data because along with the transaction count, this also inflates the revenue, quantity and other metrics directly related to transactions.

The Google Universal Analytics reporting API shares similarities with the Google Analytics 4 Reporting API, and the causes of duplicate rows in fetched reports are generally similar. Here are some potential reasons for encountering duplicate rows in Universal Analytics reports API:

Sampling:

Google Analytics may sample large data sets when generating reports. If your query exceeds the allowed threshold for unsampled data, Google Analytics might provide an approximation, leading to duplicated or overlapping data.

Multiple Dimensions or Metrics:

If your query includes multiple dimensions or metrics, it can cause in a more granular report with combinations of these dimensions and metrics, leading to seemingly duplicated rows.

Date Ranges:

If your query spans multiple date ranges or includes overlapping periods, you might receive duplicate data for the overlapping time periods.

Pagination:

If using pagination to retrieve large datasets, make sure that handling the pagination correctly. If the page size or start index is not set appropriately, that might get duplicate data in different pages.

Filters and Segments:

Applying filters or segments in your query can affect the data retrieved. Ensure that filters and segments are set up correctly to avoid unintentional duplication.

Time Zone Differences:

Time zone differences between your local time zone and the Google Analytics account settings may cause duplicates, especially when querying data around the daylight saving time changes.

Data Processing Delays:

There can be delays in data processing, especially for real-time reporting. If you query data shortly after an event occurs, you might receive duplicates as the data is still being processed.

User Activity:

If users generate multiple hits or events within the specified time range, each hit or event may be included in the report, resulting in apparent duplicates.

Querying Multiple Views or Properties:

If you are querying data from multiple views or properties, make sure that there is no overlap in the data, and the selected views or properties are distinct.

Data Collection Issues:

Occasionally, data collection issues, such as network errors or problems on the website, may lead to duplicate entries in the Google Analytics dataset.

To address and troubleshoot duplicate rows, consider the following steps:

Carefully review the API request parameters, including dimensions, metrics, date ranges, filters, and any other relevant settings. Adjust them as needed and re-run the query. Additionally, check Google Analytics documentation and release notes for any updates or changes that might affect the API requests.

Review Query Parameters:

Double-check the parameters in your API query, including dimensions, metrics, date ranges, filters, and segments. Ensure they are configured correctly.

Check Sampling Rate:

If sampling is a concern, try reducing the date range or modifying the query to minimize the likelihood of sampling.

Verify Data Consistency:

Check the data in the Google Analytics web interface to see if the duplicates are also present there. This can help you determine whether the issue is with the API or the data itself.

Consult Google Analytics Documentation:

Review the Google Analytics API documentation for any specific considerations or limitations that might be contributing to duplicate rows.

If you continue to experience issues, you may need to contact Google Analytics support for further assistance.

By:

Ameenah Al-Haidari