Build an ETL Process with Clickstream Data

Problem Statement

GoShop is an online e-commerce company that caters to different products to users across the globe. The company is growing at an exponential rate and has seen tremendous growth in the last 2 years.

The users can place their orders through the company website. The company tracks each and every activity of the user on the website. This information collected from the user activity is known as Clickstream data.

Clickstream data is the information that is collected about a user while they are browsing through a website. This includes a wide range of information such as -

- Which browser did the user use to visit the website?
- What page did the user visit on the website?
- Was the user logged in while visiting the website?
- Did the user click on a certain element of the webpage?

Now, the company would like to use Clickstream data to understand the user behavior on the website. In this hackathon, as a Data Engineer of the company, you will need to build an ETL process with the Clickstream data.

Objective

You will be given the Clickstream data of the company. Your task at hand is to implement an ETL process and generate a table that will help the company to know more about its user base. This table can be used by the company in answering the following questions -

- The users who visited on a particular date.
- How many registered and non-registered users visited the website on a particular date?
- Which was the first URL the user visited on the website on a particular date?
- Determining the no. of Clickstream events occurring on the website.

Environment Setup

You are allowed to use only PySpark to develop the solution.

.

About the Dataset

You are provided with a sample dataset containing 2 files - Clickstream and Login.

1. Clickstream (20% of the total records will be shared with the participants in JSON format)

2. Login (All the records will be shared with the participants in CSV format)

Data Dictionary

Clickstream

This table contains information about the clicks occurred on the website from 1st August, 2022 to 10th August, 2022.

Variable	Datatype	Description
browser_id	string	Id of browser from which user is accessing the website.
session_id	string	Id of the session created for the visiting user.
client_side_data	string	Embedded JSON element containing - current_page_url, time_elapsed **current_page_url** - The URL of the page the user has visited with the click. **time_elapsed** - The time spent by the user on the particular page.
event_date_time	string	Date and time when the event occurred on the website.
event_type	string	Whether the click was a simple **click** event or a **pageload** event. A new page is loaded on a **pageload** event.

Login

This table contains information about when the user logged in on the website from 1st August, 2022 to 10th August, 2022.

Variable	Datatype	Description
login_date_time	string	Date and time at which the user logged in to the website.
session_id	string	Id of the session created for the visiting user.
user_id	string	Unique id of registered user on the website.

Submission File Format

You need to submit the .py file that creates the output file similar to that of the sample submission file. The format of the sample submission file is given below:

Variable	Datatype	Description

current_date	string	The date of the click.
browser_id	string	Id of browser from which user is accessing the website.
user_id	string	Unique id of a registered user on the website.
logged_in	string	0/1 flag determining whether the user is logged in or not. Consider 0 as logged out and 1 as logged in
first_url	string	Url of the page on which the user first interacted with on the particular date.
number_of_clicks	string	Number of **click** events for the given user on the given date.
number_of_pageloads	string	Number of **pageload** events for the given user on the given date.

Points to Remember

- The dataset contains the following scenarios

- Scenario 1 -
    - It is given that at any given point of time, a single **user_id** maps to a single **session_id**.
    - **session_id** for a **user_id** can change for various reasons like the user logged out of the website, or the session for a user became stale after being inactive for some time, etc.

- Scenario 2 -
    - A session_id can map to multiple browser_ids at any given time. This can happen when a user has logged in from two different browsers.

Process of Submission

Evaluation Process

1. Your code file will be evaluated on the entire clickstream dataset residing at our end.
2. The data is further divided into Public (40%) and Private (60%) data. Your initial responses will be checked and scored on the public data.
3. Your code file will generate the output file. On success, the output file is compared with the solution file and a score is generated according to the custom evaluation metric.
4. **The custom evaluation metric is a sum of 1-Mean Absolute Percentage Error and accuracy. 1-MAPE or Accuracy is calculated for each column and then aggregated. Depending upon the type of variable, either MAPE or accuracy is selected.**
    - **1- Mean Absolute Percentage Error (Normalized Distance) is computed in the case of continuous column**
    - **Accuracy is considered in the case of categorical column**
5. Your script will get a maximum execution time of 10 minutes to run at our end. If it executes for more than 10 minutes, the process is killed.
6. On submission, the status of the solution file can be either success or a failure.

- On success, an output file along with the log file will be generated and the score is calculated.
- On failure, one of the following could be the possible reasons -
    - The solution was executed for more than 10 minutes. In this case, no log file will be returned and no output file will be created.
    - The solution faced some errors during the execution. In this case, a log file will be returned but no output file will be created.

Submission Tutorials

1. All Submissions are to be done at the solution checker tab.
2. For a step-by-step view on how to make a submission check the below video

Hackathon Rules and Conditions

1. **No. of submissions per day is 25.**
2. The final rankings would be based on your private score which will be published once the competition is over.
3. Setting the final submission is recommended. Without a final submission, the submission corresponding to the best public score will be taken as the final submission.
4. Only individual participation is allowed in this hackathon.
5. Use of external data is not allowed.
6. Entries submitted after the contest is closed, will not be considered.
7. Throughout the hackathon, you are expected to respect fellow hackers and act with high integrity.
8. The use of multiple Login IDs will lead to immediate disqualification.
9. Analytics Vidhya holds the right to disqualify any participant at any stage of the competition if the participant(s) are deemed to be acting fraudulently.