Analytics Vidhya

JOB-A-THON - June 2021

Client ComZ Confidential

Input Feature Extraction:

The input feature table consists of 9 columns. They are as follows:

1. UserID
2. No_of_days_Visited_7_Days
3. No_Of_Products_Viewed_15_Days
4. User_Vintage
5. Most_Viewed_product_15_Days
6. Most_Active_OS
7. Recently_Viewed_Product
8. Pageloads_last_7_days
9. Clicks_last_7_days

This feature has to given to the Data Science team. The approach to get this features dataset consists of 3 phases.

- Data Collection
- Exploratory Data Analysis
- Filling of null values
- Transforming data columns to standard format
- Develop input feature based on dataset

The technology used in the project is **Python.** Libraries used are Pandas, numpy, datetime.

Data Collection:

The dataset that is used to extract input feature consists of 2 datasets.

- Visitor Log table
- User Table

The User Table has the data of every user and its signup date

The Visitor Log table has webclientid, Productid, user's activity, OS, etc.,

Exploratory Data Analysis:

A info in the Visitor Log table :

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 6588000 entries, 0 to 6587997
Data columns (total 7 columns):
 #   Column        Dtype
---  ------        -----
 0   webClientID   object
 1   VisitDateTime  object
 2   ProductID     object
 3   UserID        object
 4   Activity      object
 5   Browser       object
 6   OS            object
dtypes: datetime64[ns](1), object(6)
memory usage: 402.1+ MB
```

	webClientID	VisitDateTime	ProductID	UserID	Activity	Browser	OS
count	6588000	5929085	6060863	650695	5698554	6588000	6588000
unique	1091455	5393484	17459	34050	4	82	30
top	WI10000057	5/16/2018 12:54	Pr100017	U100347	click	Chrome	Windows
freq	8877	8	103845	14671	3041039	4709203	3948358
first		00:01.4					
last		59:59.6					

I have also attached the **pandas profiling** findings also.

From this information, the observations are webclient ID is present for all the rows of data. There are null values in other columns.

Data set range:

The input feature date should range from 7-May-2018 to 27-May-2018. To make it scalable I have used max(VisitDateTime) and then subtracted with 21 gives the start date and end date is the max date.

Date Time:

The Visit Date Time has 2 different Date format,

- normal human readable format
- Unix epoch

To convert these two formats, first we need to distinguish between these date format and then use different technique to convert. Initially form the above info we can see that VisitdateTime column is in object dtype.

I have used .apply and lamda function integrating with if condition (str.contains).

If VisitDateTime contains substring 2018- :

Then use .to_datetime(VisitDateTime)

Else :

Then use .to_datetime(VisitDateTime, unit ='s')

Where unit = 's' is for Unix conversion of date time

Now we will have the standard time throughout the dataset.

UserID & webClient ID:

There are totally 34050 registered users. From the observation we know that a UserID can have multiple webclientID. Based on the webclientID, we can map userID by using groupby

23]:

UserID	webClientID	VisitDateTime	ProductID	Activity	Browser	OS
U100002	WI10000020634	2018-05-18 10:35:29.008	Pr100258	pageload	Chrome Mobile	Android
		2018-05-18 10:39:41.834	pr100258	None	Chrome Mobile	Android
		2018-05-18 10:39:46.000	Pr100258	click	Chrome Mobile	Android
		2018-05-18 10:39:53.385	Pr101097	None	Chrome Mobile	Android
		2018-05-18 10:43:56.628	Pr100258	pageload	Chrome Mobile	Android

Fig shows the userid U100002 after groupby

OS:

The OS column has different values for the same category.

For eg.

- 'Android', ------ 'android',
- 'Windows', ------ 'windows',

Replace all the duplicate values with the single value for a category

Activity:

The same goes for Activity, there are 2 category clicks and pageload. The activity column has different values for the same category. Upper and low

er case for each category.

Replace all the duplicate values with the single value for a category

Develop input feature based on dataset:

Before developing input features, 3 dataframe should be created

- DataFrame with 7 days range
- DataFrame with 15 days range
- DataFrame with 21 days range
-

No of days visited 7 days:

- With the 7 days range dataframe, groupby userid and unique dates.
- Every user would have not logged in last 7 days. So, the remaining user will be marked as 0.

No Of Products Viewed 15 Days:

- With the 15 days range dataframe, count the unique products.
- Every user would have not logged in last 7 days. So, the remaining user will be marked as Product101

User Vintage:

- Merge userData and Visitor Log data.
- Get the maximum date of every user
- Subtract the maximum date with the signup date, the result will be the user vintage

Most Viewed product 15 Days:

- Extract the table which has activity as *pageload* from 15 days range dataframe
- From the table groupby userId and productid
- Get the maximum value index which gives the productid.

Most Active OS:

- After replacing duplicate values from the OS column, group the userid and OS
- Get the size of each os of every user
- Get the maximum value index which gives the most active OS.

Recently Viewed Product:

- Extract the table where activity is *pageload*
- Sort the VisitDateTime column can get first row alone using tail(1)
- The first row will have most recent time
- The ProductID from the row will provide most recent product

Pageloads & Click for last 7 days:

- The process is same for both click and pageload
- Group the dataframe by UserID and Activity using size of activity
- Fill 0 if there is no activity performed by the user