trainity

Microsoft

Operations and Metric Analytics

Introduction

Project Description

- Operation Analytics is an integral part of the organization through which Analysts generate useful insights related to the company’s operations.
- In this project, I have played the role of a Data Analyst at Microsoft and analyzed given datasets based on two case studies.
- The project mainly focuses on two operations which are Investigating Job Data and Metric spikes.

Tech-Stack used –

- I have used MySQL and Workbench to complete the project and used Canva to make the project report.
- MySQL allowed me to conduct in-depth data analysis and Workbench helped to write sql queries in a flexible manner.

Problem Statement

The project is going to solve total of 9 questions based on two operational case studies of Microsoft

Questions on Case Study 1 (Job_Data)

This case study involves only one table named Job_data which comprises of details of the users and their activities. We are going to take insights on the following questions:

- **A.Number of jobs reviewed:** Calculate the number of jobs reviewed per hour per day for November 2020?
- **B.Throughput:** Calculate 7 day rolling average of throughput? For throughput, do you prefer daily metric or 7-day rolling and why?
- **C.Percentage share of each language:** Calculate the percentage share of each language in the last 30 days?
- **D.Duplicate rows**: How will you display duplicates from the table?

Questions on Case Study 2 (Investigating metric spike)

This case study involves 3 tables which are users, events and email_through which we are going to take insights on the following questions:

- **A.User Engagement:** Calculate the weekly user engagement?
- **B.User Growth:** Calculate the user growth for product?
- **C.Weekly Retention:** Calculate the weekly retention of users-sign up cohort?
- **D.Weekly Engagement:** Calculate the weekly engagement per device?
- **E.Email Engagement:** Calculate the email engagement metrics?

Insights and Approach

Case Study 1 (Job_Data)

A: Number of jobs reviewed:

- The output shows the no of jobs reviewed in each day in 2020. It also showcase hourly data.
- Based on the data, we can see that 25th, 26th and 29th has the highest number of jobs review in 2020
- If we can see the amount of time spent in each job in each hour, 29th November is having the least time taken to review a job

Approach: I have tried to find out the total jobs reviewed in each day using count function and group by method. Also calculated time spent in hour format to follow the question.

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

ds	total_jobs_reviewed	time_spent_in_hour	approx_jobs_per_hour
2020-11-30	7	0.17	40.0636
2020-11-26	6	0.15	39.8524
2020-11-25	5	0.09	54.8780
2020-11-28	3	0.04	80.0000
2020-11-29	1	0.01	180.0000
2020-11-27	1	0.03	34.6154
2020-11-01	1	0.05	20.3390
2020-11-02	1	0.06	15.5172
2020-11-03	1	0.02	65.4545
2020-11-04	1	0.04	24.8276
2020-11-05	1	0.07	14.1176
2020-11-06	1	0.04	22.6415
2020-11-07	1	0.05	19.4595
2020-11-08	1	0.05	20.5714
2020-11-09	1	0.10	10.2857
2020-11-10	1	0.04	24.0000
2020-11-11	1	0.04	24.0000
2020-11-12	1	0.04	24.0000
2020-11-13	1	0.04	24.0000
2020-11-14	1	0.04	24.0000
2020-11-15	1	0.04	24.0000
2020-11-16	1	0.04	24.0000
2020-11-17	1	0.04	24.0000
2020-11-18	1	0.04	24.0000
2020-11-19	1	0.04	24.0000
2020-11-20	1	0.04	24.0000
2020-11-21	1	0.04	24.0000
2020-11-22	1	0.04	24.0000
2020-11-23	1	0.04	24.0000
2020-11-24	1	0.04	24.0000

Result 2

Case Study 1 (Contd..)

B: Calculate 7 day rolling average of throughput?

- The table on the right side shows 7 day rolling average of throughput. (Here throughput = no of jobs reviewed in each hour)

Approach: In the previous question we have already found the no of jobs review in each hour. I have reused that code in a CTE named 'throughput'. After that, I used avg() windows function to find out the 7 day rolling average.

Why 7day rolling average is better than daily metric?

- I prefer 7 day rolling average over daily metric, because it helps us to forecast future trends based on previous data.
- It also enables us to separate out random variations within the data.

	A	B
1	ds	rolling_average_7days
2	11/1/2020	20.339
3	11/2/2020	17.9281
4	11/3/2020	33.77023333
5	11/4/2020	31.534575
6	11/5/2020	28.05118
7	11/6/2020	27.14956667
8	11/7/2020	26.05098571
9	11/8/2020	26.08418571
10	11/9/2020	25.33682857
11	11/10/2020	19.41475714
12	11/11/2020	19.29652857
13	11/12/2020	20.7083
14	11/13/2020	20.90237143
15	11/14/2020	21.55101429
16	11/15/2020	22.04081429
17	11/16/2020	24
18	11/17/2020	24
19	11/18/2020	24
20	11/19/2020	24
21	11/20/2020	24
22	11/21/2020	24
23	11/22/2020	24
24	11/23/2020	24
25	11/24/2020	24
26	11/25/2020	28.41114286
27	11/26/2020	30.67577143
28	11/27/2020	32.19225714
29	11/28/2020	40.19225714
30	11/29/2020	62.47797143
31	11/30/2020	64.77277143

C: the percentage share of each language in the last 30 days?

- The below table shows the percentage share of each language based on last 30days data
- We can see that Hindi language has the highest percentage share among the others which is ***34.04%***
- Arabic language has the lowest percentage share which is ***4.26%***

Approach: I have used count() aggregate function, a simple percentage formula and group by method to show percentage. Also used between operator to filter out last 30days.

	A	B
1	language	percentage
2	English	27.66
3	Arabic	4.26
4	Persian	8.51
5	Hindi	34.04
6	French	10.64
7	Italian	8.51
8	Enlish	6.38

	A	B	C	D	E	F	G	H
1	ds	job_id	actor_id	event	language	time_spent	org	row_no
2	11/23/2020	31	1025	skip	Hindi	150	A	2
3	11/24/2020	27	1029	transfer	French	150	C	2

D: Duplicate rows

- The above two rows are found as duplicate rows within our data.

Approach: I have use window function row_number() to give row numbers to each unique records. Also used sub- query to find out the rows which has row_number other than 1, which shows all the duplicate rows.

Insights on Case Study 2 (Investigating metric spike)

A: Weekly user engagement

- According to the output, we can see that ***week 30*** has the highest amount of user engagement which is 21,533 users in that week
- Similarly, ***Week 35*** has the lowest user engagement level.

Approach: I have used extract function to extract the weeks and used count and group by to find engagement. Using where clause I filtered only the 'engagement' values as well.

	A	B
1	Weeks	engagement
2	30	21533
3	28	20776
4	29	20067
5	27	19881
6	26	19061
7	24	19052
8	25	18642
9	31	18556
10	22	18413
11	23	18280
12	20	17911
13	18	17341
14	19	17224
15	21	17151
16	32	16612
17	33	16145
18	34	16127
19	17	8019
20	35	784

B: User Growth: Amount of users growing over time for a product.

We can see that there is ***85% growth*** of new users in 2014 compared to 2013

Approach: Our dataset includes user details of 2013 and 2014. So, I have tried to find out the user growth from 2013 to 2014.
First, I used subquery to find out the total new users registered in each year. Then I used lag() windows function to find the percentage.

	A	B	C
1	years	total_new_users	yearly_users_growth_percentage
2	2013	3283	NULL
3	2014	6098	85.7447
4			

Insights on Case Study 2 (Contd...)

C: Weekly retention of users-sign up cohort

- The two part-wise screenshot on the right side shows total number of users retained in each week.
- We can see that ***Week 34*** has the highest number of users retained which is 337 users.

Approach: I have extracted the weeks from created_at column and counted the users who are 'active' after sighnup. Finally displayed the output with group by clause.

	A	B
1	weeks	retained_users
2	34	337
3	33	334
4	32	316
5	30	305
6	29	288
7	28	287
8	27	274
9	24	274
10	25	264
11	31	260
12	26	257
13	22	250
14	23	246
15	19	242
16	21	232
17	16	225
18	17	219
19	20	215
20	15	207
21	18	207
22	13	206
23	14	197
24	10	186
25	12	181

	A	B
25	12	181
26	5	181
27	9	176
28	6	173
29	7	167
30	8	163
31	11	161
32	4	160
33	2	157
34	1	156
35	3	149
36	50	124
37	49	116
38	0	106
39	47	102
40	51	102
41	42	99
42	48	97
43	44	96
44	45	91
45	38	90
46	43	89
47	46	88
48	40	87
49	37	85
50	39	84
51	35	81
52	41	73
53	36	72
54	52	47

Insights on Case Study 2 (Contd...)

D: Weekly engagement per device

- Due to long list of rows, we are not showing the entire table here, however we can see that ***mackbook pro*** users has the highest amount of engagement in most of the weeks.
- Considering a specific product (macbook pro), ***week 31*** got the highest engagement of users.

Approach: I have extracted the weeks from occured_at column and counted the event_type 'Engagement'.
I also applied group by method to both weeks and device to find engagement of each product in each week.
Finally used order by to sort it according to descending order of engagement level.

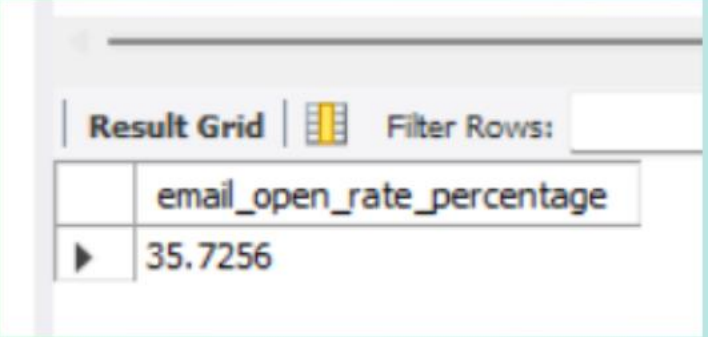
	A	B	C
1	device	Weeks	engagement
2	macbook pro	31	3608
3	macbook pro	30	3578
4	macbook pro	27	3548
5	macbook pro	28	3461
6	macbook pro	32	3320
7	macbook pro	26	3309
8	macbook pro	18	3301
9	macbook pro	33	3182
10	macbook pro	19	3159
11	macbook pro	29	3155
12	macbook pro	34	3141
13	macbook pro	23	3123
14	macbook pro	20	3097
15	macbook pro	22	3046
16	macbook pro	21	3044
17	macbook pro	24	3028
18	macbook pro	25	2932
19	lenovo thinkpad	30	2584
20	lenovo thinkpad	28	2564
21	lenovo thinkpad	29	2438
22	lenovo thinkpad	27	2233
23	lenovo thinkpad	26	2214
24	lenovo thinkpad	20	2203
25	lenovo thinkpad	33	2156
26	lenovo thinkpad	19	2143
27	lenovo thinkpad	31	2114
28	lenovo thinkpad	25	2096
29	lenovo thinkpad	34	1908
30	lenovo thinkpad	32	1898

E: Email Engagement Metrics:

Open Rate (in percentage)

- The open rate is a vital email engagement metrics and based on our dataset, open rate of email is **35.72%.**

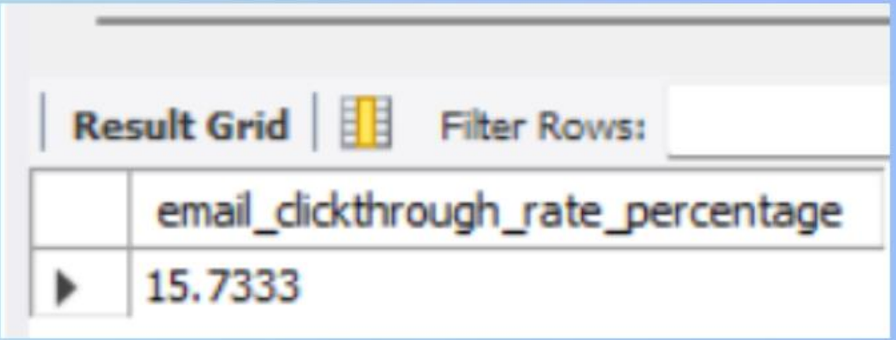
Approach: I have counted both 'email_open' and 'sent_weekly_digest' records to calculate the open rate. Used CTE and subquery to display the required output.



email_open_rate_percentage
35.7256

Clickthrough Rate (in percentage)

- Clickthrough rate is the number of users who have clicked on at least one link from the email.
- I found out that, click through rate of our emails is ***15.73%***



email_clickthrough_rate_percentage
15.7333

CONCLUSION

- As a result of this intensive SQL project, I got hands-on experience on advanced concepts of SQL such as Windows function, CTEs.
- I had a good exposure to learning about various operations done in a product-based company and how Data analysts can bring insight regarding that.
- I have improved my problem-solving skills with this project by going through various obstacles, finding out their root cause and coming up with right solutions through internet research.

Thank You

trainity

Microsoft