Operation Analytics and Investigating Metric Spike

Project - 3

Project Description:

The project includes_consists of 2 case studies ->

- First is regarding Operation Analytics where job data is provided and number of jobs reviewed, 7day rolling average of throughput, percentage share of language used and duplicates are found out.
- Second is Investigating Metric Spike where user engagement, user growth, weekly retention, weekly engagement, and email engagement is determined.

Approach:

The required information was determined via SQL queries where the data base was created first in SQL and table job_data is created in the database and moreover for the

second case study due to the size of the data excel was used to make charts for better visualisation.

Tech-Stack Used:

I used the following tools in this project ->

- MySQL was used to run the queries in Case Study1.
- The language was selected because of comfort and experience in the same.
- MS Excel was used in the Case Study2 for better visualisations.

Case Study1-Job Data Analysis:

Tasks

A. Jobs Reviewed Over Time-

```
SELECT ds as Dates, round(count(job_id)/ (SUM(time_spent))*3600) as Number_of_jobs
from job_data
group by ds;
```

Result ->

Dates	Number_of_jobs
11/25/2020	80
11/26/2020	64
11/27/2020	35
11/28/2020	218
11/29/2020	180
11/30/2020	180

B. Throughput Analysis –

Weekly Throughput

```
1 Select Round(COUNT(event)/Sum(time_spent),4) as "Weekly_ThroughPut"
2 from job_data;
```

```
Weekly_ThroughPut
0.0268
```

Weekly Throughput is '0.0268'

Daily Throughput

```
1 Select ds as Date ,Round(COUNT(event)/Sum(time_spent),4) as "Daily_ThroughPut"
2 from job_data
3 GROUP by ds
4 ORDER by ds;
```

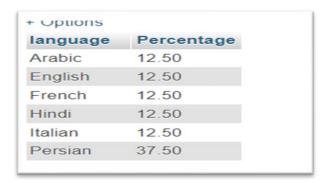
+ Options

Date	${\bf Daily_ThroughPut}$
11/25/2020	0.0222
11/26/2020	0.0179
11/27/2020	0.0096
11/28/2020	0.0606
11/29/2020	0.0500
11/30/2020	0.0500

Metrics will always go up and down and daily basis. We well get number everyday if we want. So, rolling metrics are superb at showing if our metrics are trending up or down on a daily level.

C. Language Share Analysis -

```
SELECT language, Round(COUNT(*)*100/ total,2) AS Percentage
from job_data
CROSS JOIN(SELECT COUNT(*) as total from job_data)sub
GROUP by language;
```



D. <u>Duplicate Rows Detection –</u>

```
1 with cte as(
2 SELECT *,row_number() over(PARTITION by job_id) as Row_Num
3    from job_data)
4 SELECT * from cte where Row_Num>1;
```

+ Options

ds	job_id	actor_id	event	language	time_spent	org	Row_Num
11/26/2020	23	1004	skip	Persian	56	Α	2
11/29/2020	23	1003	decision	Persian	20	С	3

Case Study2- Investigating Metric Spike:

Tasks

A. Weekly User Engagement -

```
SELECT extract(week FROM occurred_at) as Week_Number,
count(DISTINCT user_id) as Active_Users
from events
GROUP BY Week_Number
ORDER BY Week_Number
```

Week_Number	Active_Us	ers
20	1046	
23	1872	
24	2182	
27	1306	
28	2888	
32	2553	2553
33	1621	

B. <u>User Growth Analysis –</u>

```
SELECT year, week_num, num_users, SUM(num_users)

over(order by year, week_num) as Cum_Users

from(

SELECT extract(year from created_at) as year, extract(week from created_at) as week_num, count(DISTINCT user_id) as num_users

from users

WHERE state = 'active'

GROUP by year, week_num

ORDER BY year, week_num

sub
```

C. Weekly Retention Analysis -

```
1 with cte1 as(
     SELECT DISTINCT user_id,
     extract(week from occurred_at) as signup_week
    from events
     where event_type = 'signup_flow'
      and event_name = 'complete_signup' and extract(week from occurred_at) = 18),
     SELECT DISTINCT user_id,
     extract(week from occurred_at) as engagement_week
      where event_type = 'engagement')
2 SELECT count(user_id)total_engaged_users,
3 SUM(case when retention_week > 0 then 1 else 0 end) as retained_users
4 from(SELECT a.user_id, a.signup_week,
       b.engagement_week,b.engagement_week-a.signup_week as retention_week
.6
      from ctel a
7
     LEFT JOIN cte2 b
8
     on a.user_id = b.user_id
      ORDER BY a.user_id)sub
9
```

```
total_engaged_users retained_users

236
```

D. Weekly Engagement Per Device –

```
with cte as (SELECT extract(year from occurred_at)||'-'||extract(week from occurred_at)
as week_num,device,COUNT(DISTINCT user_id) as user_count

FROM events
where event_type = 'engagement'
GROUP BY week_num, device
ORDER BY week_num
)

SELECT week_num,device,user_count
from cte
```

week_num	device	user_count
2014-18	acer aspire desktop	10
2014-18	acer aspire notebook	21
2014-18	amazon fire phone	4
2014-18	asus chromebook	23
2014-18	dell inspiron desktop	21

E. Email Engagement Analysis –

```
SELECT

100 * sum(case when email_cat = 'email_open' then 1 else 0 end)/
    sum(case when email_cat = 'email_sent' then 1 else 0 end) as email_open_rate,

100 * sum(case when email_cat = 'email_clicked' then 1 else 0 end)/
    sum(case when email_cat = 'email_sent' then 1 else 0 end) as email_click_rate

6 from(SELECT *,

CASE

when action in('sent_weekly_digest','sent_reengagement_email') then 'email_sent'
    when action in('email_open') then 'email_open'

when action in('email_clickthrough') then 'email_clicked'

end as email_cat

from email_events) sub
```

```
email_open_rate email_click_rate 31.1921 10.4745
```

Result:

The project's key results included identification of reviewed jobs and their distribution across languages, the calculation of retention rates, and the identification of retained users through an in-depth analysis that relied on predefined assumptions. SQL is one of the most crucial skills for anyone in a data driven position. Additionally, this project helped me to gain insight of various factors which are crucially important for the business to run for a long period and grow as well. Brainstorming is the key to run successful business