

Operating Analytics and Investigating Metric Spike

Project Description:

The project is done to analyze complete end to end operations of a company. This project consists of two sections Operating Analytics and Investigating Metric Spike

Operating Analytics:

It is used to measure the existing and real time operations of the company so that they can monitor their day to day operations. Once they get the useful insights, the company reflects on these to improve their operations.

Investigating Metric Spike:

This is used to investigate the user engagement, the growth of users, the number of users using a particular product week wise etc. The insights also help in increasing the user engagement with the given products in the data set.

Approach:

The datasets have to be used to create a database in MySQL. After creation of the tables and inserting data, different queries have to be used to answer questions like user engagement, growth in the number of users using a particular project etc. The insights have to be presented to the ps team and the support team so that the needful action can be taken to improve the efficiency and performance of the company.

Tech Stack Used:

MySQL Workbench is used for the database and SQL Queries. The SQL queries can be classified to DDL, DML and TCL. Math functions and Date functions are used throughout the project. Advanced SQL is the main technology used here.

Insights:

Case Study 1:

Operating Analytics:

The following code is for creating the tables given

```
create database if not exists op_metric_analysis;  
show databases;  
use op_metric_analysis;  
show tables;
```

```
DROP TABLE job_data;
```

```
create table job_data
```

```
(job_id INT NOT NULL,
actor_id INT NOT NULL,
event varchar(30), CHECK(event IN ('decision','skip','transfer')),
language varchar(30),
time_spent INT,
org varchar(60),
ds date,
PRIMARY KEY(job_id,actor_id));
```

```
insert into job_data values(21, 1001, 'skip', 'English', 15, 'A', '2020-11-30'),
(22,1006,'transfer','Arabic',25,'B','2020-11-30'),
(23,1003,'decision','Persian',20,'C','2020-11-29'),
(23,1005,'transfer','Persian',22,'D','2020-11-28'),
(25,1002,'decision','Hindi',11,'B','2020-11-28'),
(11,1007,'decision','French',104,'D','2020-11-27'),
(23,1004,'skip','Persian',56,'A','2020-11-26'),
(20,1003,'transfer','Italian',45,'C','2020-11-25');
```

/*

1. Number of jobs reviewed: Amount of jobs reviewed over time.

Your task: Calculate the number of jobs reviewed per hour per day for November 2020?

*/

```
select count(job_id) as num_of_jobs, sum(time_spent)/3600 as 'hours', ds as the_date
from job_data
where ds>='2020-11-01' and ds <='2020-11-30'
group by ds;
```

/*

2. Throughput: It is the no. of events happening per second.

Your task: Let's say the above metric is called throughput. Calculate 7 day rolling average of throughput? For throughput, do you prefer daily metric or 7-day rolling and why?

*/

```
create view temp_tab as
select job_id, ds, event,
count(event) over(partition by job_id)/sum(time_spent) over(partition by job_id) AS throughput
from job_data;
```

```
select * from temp_tab;
```

```
select job_id, ds, event, throughput,  
avg(throughput) over(order by ds rows between 6 preceding and current row) as rolling_avg  
from temp_tab;
```

```
/*
```

3. Percentage share of each language: Share of each language for different contents.
Your task: Calculate the percentage share of each language in the last 30 days?

```
*/
```

```
drop view total_languages;
```

```
create view total_languages as  
select count(distinct(language)) as tot_lang from job_data;
```

```
select tot_lang from total_languages;
```

```
select job_data.language, count(job_data.language)/total_languages.tot_lang*100 AS  
percentage_share  
from job_data, total_languages  
group by language;
```

```
/*
```

4. Duplicate rows: Rows that have the same value present in them.
Your task: Let's say you see some duplicate rows in the data. How will you display
duplicates from the table?

```
*/
```

```
select job_id,actor_id, count(*)  
from job_data  
group by job_id,actor_id  
having count(*) > 1;
```

Case Study 2: **Investigation Metric Spike:**

1. Calculate the weekly user engagement?

```
SELECT EXTRACT(WEEK FROM user_id) AS Weekly Active Users FROM events
WHERE event_type = 'engagement' GROUP BY 1;
```

2. Calculate the user growth for a product?

```
Select Months, Users, Round(((Users/LAG(Users, 1) over order by (Months) -1)*100), 2)
AS Growth_in_per from
(
    Select extract(Month from created_at) as Months, count(activated_at) as Users
    From Users where activated_at not in ( ' ' ) group by 1, order by 1
) sub;
```

3. Calculate the weekly retention of users sign-up cohort?

```
select first as "Week Numbers",
sum(case when week_num = 0 then 1 else 0 end) as "Week 0",
sum(case when week_num = 1 then 1 else 0 end) as "Week 1",
sum(case when week_num = 2 then 1 else 0 end) as "Week 2",
sum(case when week_num = 3 then 1 else 0 end) as "Week 3",
sum(case when week_num = 4 then 1 else 0 end) as "Week 4",
sum(case when week_num = 5 then 1 else 0 end) as "Week 5",
sum(case when week_num = 6 then 1 else 0 end) as "Week 6",
sum(case when week_num = 7 then 1 else 0 end) as "Week 7",
sum(case when week_num = 8 then 1 else 0 end) as "Week 8",
sum(case when week_num = 9 then 1 else 0 end) as "Week 9",
sum(case when week_num = 10 then 1 else 0 end) as "Week 10",
sum(case when week_num = 11 then 1 else 0 end) as "Week 11",
sum(case when week_num = 12 then 1 else 0 end) as "Week 12",
```

```

sum(case when week_num = 13 then 1 else 0 end) as "Week 13",
sum(case when week_num = 14 then 1 else 0 end) as "Week 14",
sum(case when week_num = 15 then 1 else 0 end) as "Week 15",
sum(case when week_num = 16 then 1 else 0 end) as "Week 16",
sum(case when week_num = 17 then 1 else 0 end) as "Week 17",
sum(case when week_num = 18 then 1 else 0 end) as "Week 18"

from

(
    select m.user_id, m.login_week, n.first, m.login_week - first AS week_number
    from
        (select user_id, extract(week from occurred_at) as login_week from events
        group by 1, 2)m,
        (select user_id, min(extract(week from occurred_at)) as first from events
        group by 1, 2)n
    where m.user_id = n.user_id
)sub

group by first order by first;

```

4. Calculate the weekly engagement per device?

```

Select Extract(Week from occurred_at) as "Week Numbers",

count(distinct case when device in('dell inspiron notebook') then user_id else null end) as
"Dell Inspiron Notebook",

```

count(distinct case when device in('iphone 5') then user_id else null end) as "iPhone 5",

count(distinct case when device in('iphone 4s') then user_id else null end) as "iPhone 4S",

count(distinct case when device in('windows surface') then user_id else null end) as
"Windows SURface",

count(distinct case when device in('macbook air') then user_id else null end) as
"Macbook Air",

count(distinct case when device in('iphone 5s') then user_id else null end) as "iPhone 5S",

count(distinct case when device in('macbook pro') then user_id else null end) as
"Macbook Pro",

count(distinct case when device in('kindle fire') then user_id else null end) as "Kindle
Fire",

count(distinct case when device in('ipad mini') then user_id else null end) as "iPad Mini",

count(distinct case when device in('nexus7') then user_id else null end) as "Nexus 7",

count(distinct case when device in('nexus5') then user_id else null end) as "Nexus 5",

count(distinct case when device in('samsung galaxy s4') then user_id else null end) as
"Samsung Galaxy S4",

count(distinct case when device in('lenovo thinkpad') then user_id else null end) as
"Lenovo ThinkPad",

count(distinct case when device in('samsung galaxy tablet') then user_id else null end) as
"Samsung Galaxy Tablet",

count(distinct case when device in('acer aspire notebook') then user_id else null end) as
"Acer Aspire Notebook",

count(distinct case when device in('asus chromebook') then user_id else null end) as
"Asus Chromebook",

count(distinct case when device in('htc one') then user_id else null end) as "HTC One",

count(distinct case when device in('nokia lumia 635') then user_id else null end) as
"Nokia Lumia 635",

```

count(distinct case when device in('samsung galaxy note') then user_id else null end) as
"Samsung Galaxy Note",

count(distinct case when device in('acer aspire desktop') then user_id else null end) as
"Acer Aspire Desktop",

count(distinct case when device in('mac mini') then user_id else null end) as "Mac Mini",

count(distinct case when device in('hp pavilion desktop') then user_id else null end) as
"HP Pavilion Desktop",

count(distinct case when device in('dell inspiron desktop') then user_id else null end) as
"Dell Inspiron Desktop",

count(distinct case when device in('ipad air') then user_id else null end) as "iPad Air",

count(distinct case when device in('amazon fire phone') then user_id else null end) as
"Amazon Fire Phone",

count(distinct case when device in('nexus 10') then user_id else null end) as "Nexus 10"

from events where event_type = 'engagement' group by 1 order by 1;

```

5. Calculate the email engagement metrics?

```

select week, round((weekly_digest/total_users*100),2) as "Weekly Digest Rate",

round((email_opens/total_users*100),2) as "Email Open Rate",

round((email_clickthroughs/total_users*100),2) as "Email ClickThrough Rate",

round((reengagement_emails/total_users*100),2) as "Re Engagement Email Rate"

from

(

select extract(week from occurred_at) as week,

count(case when action = "sent_weekly_digest" then user_id else null end) as
weekly_digest,

```

```
count(case when action = "email_open" then user_id else null end) as email_opens,  
  
count(case when action = "email_clickthrough" then user_id else null end) as  
email_clickthroughs,  
  
count(case when action = "sent_reengagement_email" then user_id else null end) as  
reengagement_emails,  
  
count(user_id) as total_users  
  
from email_events  
  
group by 1  
  
)sub  
  
group by 1 order by 1;
```

Result:

This project helped me take a deeper look at how companies use these insights to improve the day to day operations.

Learning:

The project also includes a lot of data which was very slow to load. It took me a lot of time to load the data. However through this process I learned LOAD statements and an efficient data cleaning process. Inserting dates was also a big milestone and through data cleaning in excel sheets I could fix different date functions. Also, I learnt to first look through the data and clean the data before uploading to the workbench which would make the process a little faster.