Operation Analytics and Investigating Metric Spike

Project Description

Operation Analytics is the analysis done for the complete end to end operations of a company. with the help of this, the company then finds the areas on which it must improve upon, working closely with the ops team, support team, marketing team and help them derive insights out of the data they collect. Being one of the most important parts of a company, this kind of analysis is further used to predict the overall growth or decline of a company's fortune. It also used for better automation, better understanding between cross-functional teams, and more effective workflows. Investigating metric spike is also an important part of operation analytics as being a Data Analyst you must be able to understand or make other teams understand questions like- Why is there a dip in daily engagement? Why have sales taken a dip? Etc. Questions like these must be answered daily and for that its very important to investigate metric spike.

Approach

As working in the product team of Microsoft designated as Data Analyst Lead and is provided with different data sets, tables from which that must derive certain insights out of it and answer the questions asked by different departments.

Tech-Stack Used

MySQL workbench (version 8.0) is an open-source SQL development, that helps to create database and perform SQL operation .

Insights

The knowledge that I gained from this project is how to create the database ,how to use the database , how can we perform SQL operation based on scenarios and how we can use this data to improve the experience altogether while helping the business grow.

Result

Have achieved how to analyze the project , as this helped me to gain more knowledge about SQL to perform better

Submitted by
Rahul Inchal

Operation Analytics and Investigating Metric Spike

Job Data

A. **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

Query
Select ds,
round(1.0*count(job_id)*3600/sum(time_spent),2)
as throughput
from job_data
Where
event in ('transfer','decision') and
ds between '2020-11-01' and '2020-11-30'
GROUP BY ds;

Ds	throughput
30-11-2020	144
29-11-2020	180
28-11-2020	218.18
27-11-2020	34.62
25-11-2020	80

Submitted by

Rahul Inchal

Operation Analytics and Investigating Metric Spike

B. **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?

```sql
with CTE AS (
SELECT
DS,
COUNT(job_id) as num_jobs,
SUM(time_spent) as total_time
From job_data
Where event In ('transfer','decision')
And ds Between '2020-11-01' and '2020-11-30'
Group by ds)
Select ds,round(1.0*sum(num_jobs) over
(Order by ds rows between 6 Preceding and current row) /
sum(total_time) OVER (order by ds rows between 6 Preceding and current row),2)
as throughput_for_7days
From CTE;
```

ds	throughput_for_7days
25-11-2020	0.02
27-11-2020	0.01
28-11-2020	0.02
29-11-2020	0.02
30-11-2020	0.03

Submitted by
Rahul Inchal

Operation Analytics and Investigating Metric Spike

C. **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?

```sql
WITH CTE AS (
SELECT Language,
COUNT(job_id) as num_jobs
From job_data
Where event In('transfer','decision')
And ds Between '2020-11-01' and '2020-11-30'
Group by language),
Total as(Select COUNT(job_id) as total_jobs
From job_data
Where event In('transfer','decision')
And ds Between '2020-11-01' and '2020-11-30'
Group by language)
Select distinct Language,
Round(100*num_jobs / total_jobs,2) as percentage_of_jobs
From CTE
cross join total
Order by percentage_of_jobs DESC;
```

Language	Percentage_of_jobs
Persian	200
Italian	100
French	100
Hindi	100
Arabic	100
Persian	100
Italian	50
French	50
Hindi	50
Arabic	50

Submitted by

Rahul Inchal

Operation Analytics and Investigating Metric Spike

- D. **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?

```sql
select job_id,ds,actor_id,
count(job_id) FROM JOB_DATA
group by job_id,ds,actor_id
having count(job_id)>1;
```

Ds	job_id	actor_id	Event	language	time_spent	org

Submitted by

Rahul Inchal

Operation Analytics and Investigating Metric Spike

Investigating metric spike

A. **User Engagement**: To measure the activeness of a user. Measuring if the user finds quality in a product/service.
   Your task: Calculate the weekly user engagement?

```sql
SELECT date_trunc('week',occurred_at),
count(Distinct user_id) as weekly_active_users
from events
where event_type='engagement'
and event_name='login'
group by 1
order by 1;
```

date_trunc	weekly_active_users
28-04-2014 00:00	701
05-05-2014 00:00	1054
12-05-2014 00:00	1094
19-05-2014 00:00	1147
26-05-2014 00:00	1113
02-06-2014 00:00	1173
09-06-2014 00:00	1219
16-06-2014 00:00	1262
23-06-2014 00:00	1249
30-06-2014 00:00	1271
07-07-2014 00:00	1355
14-07-2014 00:00	1345
21-07-2014 00:00	1363
28-07-2014 00:00	1442
04-08-2014 00:00	1266
11-08-2014 00:00	1215
18-08-2014 00:00	1203
25-08-2014 00:00	1194

Submitted by
Rahul Inchal

Operation Analytics and Investigating Metric Spike

B. **User Growth**: Amount of users growing over time for a product.
   Your task: Calculate the user growth for product?

```
select date_trunc('day', created_at) as day,

count(*) as all_users,

count(case when activated_at is

NOT NULL THEN user_id

else NULL END)

as activated_users

from users

where created_at >= '2013-01-01'

and created_at < '2013-01-31'

group by 1

order by 1;
```

day	all_users	activated_users
01-01-2013 00:00	13	7
02-01-2013 00:00	11	7
03-01-2013 00:00	14	6
04-01-2013 00:00	11	1
05-01-2013 00:00	3	2
06-01-2013 00:00	4	3
07-01-2013 00:00	13	4
08-01-2013 00:00	13	2
09-01-2013 00:00	11	6
10-01-2013 00:00	12	6
11-01-2013 00:00	11	6
12-01-2013 00:00	4	3
13-01-2013 00:00	3	2
14-01-2013 00:00	13	8
15-01-2013 00:00	15	11
16-01-2013 00:00	14	7
17-01-2013 00:00	13	9
18-01-2013 00:00	14	10
19-01-2013 00:00	4	1
20-01-2013 00:00	4	1
21-01-2013 00:00	13	7

Operation Analytics and Investigating Metric Spike

C. **Weekly Retention**: Users getting retained weekly after signing-up for a product.

   Your task: Calculate the weekly retention of users-sign up cohort?

```sql
SELECT DATE_TRUNC('week',z.occurred_at) AS "week",
AVG(z.age_at_event) AS "Average age during week",
COUNT(DISTINCT CASE WHEN z.user_age > 70 THEN z.user_id ELSE NULL END)
AS "10+ weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 70 AND z.user_age >= 63 THEN
z.user_id ELSE NULL END) AS "9 weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 63 AND z.user_age >= 56 THEN
z.user_id ELSE NULL END) AS "8 weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 56 AND z.user_age >= 49 THEN
z.user_id ELSE NULL END) AS "7 weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 49 AND z.user_age >= 42 THEN
z.user_id ELSE NULL END) AS "6 weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 42 AND z.user_age >= 35 THEN
z.user_id ELSE NULL END) AS "5 weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 35 AND z.user_age >= 28 THEN
z.user_id ELSE NULL END) AS "4 weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 28 AND z.user_age >= 21 THEN
z.user_id ELSE NULL END) AS "3 weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 21 AND z.user_age >= 14 THEN
z.user_id ELSE NULL END) AS "2 weeks",
COUNT(DISTINCT CASE WHEN z.user_age < 14 AND z.user_age >= 7 THEN
z.user_id ELSE NULL END) AS "1 week",
COUNT(DISTINCT CASE WHEN z.user_age < 7 THEN z.user_id ELSE NULL END) AS
"Less than a week"
```

Submitted by

Rahul Inchal

Operation Analytics and Investigating Metric Spike

```sql
FROM (
SELECT e.occurred_at,
u.user_id,
DATE_TRUNC('week',u.activated_at) AS activation_week,
EXTRACT('day' FROM e.occurred_at - u.activated_at) AS age_at_event,
EXTRACT('day' FROM '2014-09-01'::TIMESTAMP - u.activated_at) AS user_age
FROM users u
JOIN events e
ON e.user_id = u.user_id
AND e.event_type = 'engagement'
AND e.event_name = 'login'
AND e.occurred_at >= '2014-05-01'
AND e.occurred_at < '2014-09-01'
WHERE u.activated_at IS NOT NULL
) z
GROUP BY 1
ORDER BY 1
```

Submitted by

Rahul Inchal

Operation Analytics and Investigating Metric Spike

Output

week	Average age during week	10+ weeks	9 weeks	8 weeks	7 weeks	6 weeks	5 weeks	4 weeks	3 weeks	2 weeks	1 week	Less than a week
28-04-2014 00:00	124.0072389	701	0	0	0	0	0	0	0	0	0	0
05-05-2014 00:00	124.3816908	1054	0	0	0	0	0	0	0	0	0	0
12-05-2014 00:00	131.9386442	1094	0	0	0	0	0	0	0	0	0	0
19-05-2014 00:00	132.3266284	1147	0	0	0	0	0	0	0	0	0	0
26-05-2014 00:00	132.3453634	1113	0	0	0	0	0	0	0	0	0	0
02-06-2014 00:00	131.8311091	1173	0	0	0	0	0	0	0	0	0	0
09-06-2014 00:00	131.0425824	1219	0	0	0	0	0	0	0	0	0	0
16-06-2014 00:00	136.4805654	1255	0	0	0	0	0	0	0	0	0	0
23-06-2014 00:00	136.2789056	1034	210	0	0	0	0	0	0	0	0	0
30-06-2014 00:00	136.4192975	917	151	199	0	0	0	0	0	0	0	0
07-07-2014 00:00	135.8887505	899	100	130	223	0	0	0	0	0	0	0
14-07-2014 00:00	143.4488157	832	62	82	152	215	0	0	0	0	0	0
21-07-2014 00:00	141.70278	791	44	60	95	144	228	0	0	0	0	0
28-07-2014 00:00	144.0786604	805	30	43	83	91	155	234	0	0	0	0
04-08-2014 00:00	140.732238	678	24	34	52	52	82	154	189	0	0	0
11-08-2014 00:00	125.9943101	562	19	33	39	33	59	94	126	250	0	0
18-08-2014 00:00	128.0217181	522	15	26	26	19	40	64	69	163	259	0
25-08-2014 00:00	128.2698104	474	15	14	23	20	31	47	48	82	173	266

Submitted by

Rahul Inchal

D. **Weekly Engagement**: To measure the activeness of a user.
Measuring if the user finds quality in product/service_weekly.
Your task: Calculate the weekly engagement per device?

```sql
select DATE_TRUNC('week', occurred_at) as week,
COUNT(distinct e.user_id) as weekly_active_users,
count(distinct CASE WHEN e.device in('macbook pro','lenovo thinkpad','macbook air',
'dell inspiron notebook','hp pavilion desktop','accer aspire desktop','mac mini')
Then e.user_id else null end) as computer,
count(distinct case when e.device in('iphone 5','samsung galaxy s4','nexus 5','iphone 5',
'nokia lumia 635','htc one',
'samsung galaxy note','amzon fire phone') then e.user_id else null end) as phone,
count(distinct case when e.device in('ipad air','nexus 7','ipad mini','nexus 10','kindle fire',
'windows surface','samsung galaxy tablet')
then e.user_id else null end) as tablet
from events e
where e.event_type='engagement'
and e.event_name='login'
group by 1
order by 1
limit 100;
```

Submitted by
Rahul Inchal

Operation Analytics and Investigating Metric Spike

Output

Week	weekly_active_users	computer	phone	tablet
28-04-2014 00:00	701	347	211	103
05-05-2014 00:00	1054	590	335	176
12-05-2014 00:00	1094	632	352	191
19-05-2014 00:00	1147	647	385	181
26-05-2014 00:00	1113	605	379	176
02-06-2014 00:00	1173	663	388	197
09-06-2014 00:00	1219	671	397	195
16-06-2014 00:00	1262	685	400	227
23-06-2014 00:00	1249	706	395	210
30-06-2014 00:00	1271	676	424	218
07-07-2014 00:00	1355	735	434	227
14-07-2014 00:00	1345	762	421	218
21-07-2014 00:00	1363	767	436	218
28-07-2014 00:00	1442	798	414	241
04-08-2014 00:00	1266	770	354	166
11-08-2014 00:00	1215	722	326	153
18-08-2014 00:00	1203	754	311	145
25-08-2014 00:00	1194	725	312	150

Submitted by

Rahul Inchal

Operation Analytics and Investigating Metric Spike

E. **Email Engagement**: Users engaging with the email service.

　Your task: Calculate the email engagement metrics?

```sql
select date_trunc('week', occurred_at) as week,
count(case when e.action='sent_weekly_digest' then e.user_id else null end) as weekly_emails,
count(case when e.action='sent_reengagement_email' then e.user_id else null end) as reengagement_email,
count(case when e.action='email_open' then e.user_id else null end) as email_opens,
count(case when e.action='email_clickthrough' then e.user_id else null end) as email_clickthroughs
from emails e
group by 1;
```

Submitted by

Rahul Inchal

Operation Analytics and Investigating Metric Spike

<u>Output</u>

Week	weekly_emails	reengagement_email	email_opens	email_clickthroughs
14-07-2014 00:00	3499	226	1260	607
28-07-2014 00:00	3706	230	1386	633
19-05-2014 00:00	2733	179	995	498
05-05-2014 00:00	2602	164	919	434
11-08-2014 00:00	3897	224	1357	430
12-05-2014 00:00	2665	175	971	479
25-08-2014 00:00	4111	263	1533	493
28-04-2014 00:00	908	98	332	187
26-05-2014 00:00	2822	179	1026	453
07-07-2014 00:00	3399	214	1230	622
09-06-2014 00:00	3003	190	1070	533
16-06-2014 00:00	3105	234	1161	563
30-06-2014 00:00	3302	222	1168	559
21-07-2014 00:00	3592	206	1211	584
02-06-2014 00:00	2911	199	993	492
18-08-2014 00:00	4012	257	1421	487
23-06-2014 00:00	3207	187	1090	524
04-08-2014 00:00	3793	206	1336	432

<u>Submitted by</u>

<u>Rahul Inchal</u>