Operation Analytics

Investigating Metric Spike

Submitted by Anuj Pratap Singh

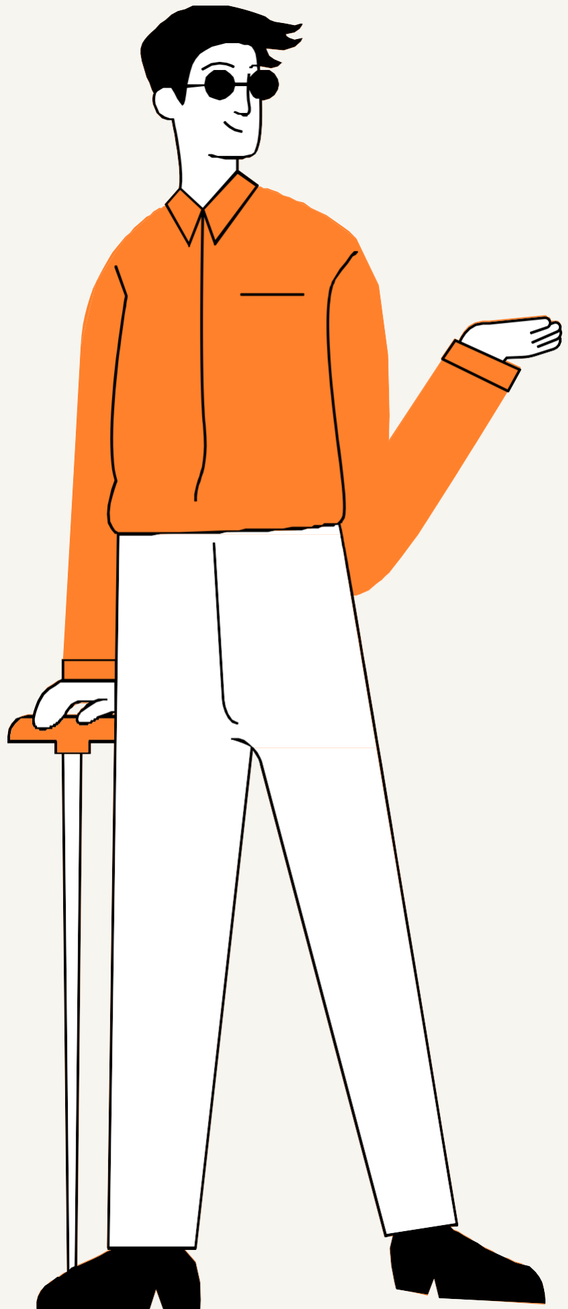
Project Description

-The project revolves around operation analytics that involved analyzing the end-to-end operations of the company, crucial for them to identify scalable opportunities and key improvement areas.
-While working as the Lead Data Analyst of a product company, I was given two separate case studies wherein I had to work closely with the ops team, support team, marketing team, etc. and help them derive insights out of the data they collected.
-In the first study, I analyzed the job related dataset and derive reviewal insights necessary for the ops team to understand the division of jobs reviewed and their attributes
-In the second study, I worked mainly to investigate the metric spikes and find out answers to questions like, why are the weekly retention rates low, or why is there a weekly dip in the user engagement.

Tech-Stack Used

Microsoft SQL Server

I created a relational schema and ran some SQL queries on the software to find insights about the scope of my study. The dataset was relatively huge and had around 300k records owing to which SQL server was the best option to load and execute the data quickly



Case Study 1- Job Review Analysis

This case study analyzes jobs that were reviewed during a period of time, breaking the events down to hourly and weekly data. The objective was to derive insights about the reviewal of each job and to find areas of improvement.

Case Study 2 - Investigating Metric Spike

This case study focuses on understanding user engagement trends and retention rates for the cohort to find out answers to some of the crucial questions like why is there a dip in daily engagement or why are the retention rates so low. The objective of the study is to find out if the users are finding quality in the product/service and also identifying white space opportunities for future product development plans.

Case Study 1

Jobs Table

ds	job_id	actor_id	event	language	time_spent	org
30-11-2020	21	1001	skip	English	15	A
30-11-2020	22	1006	transfer	Arabic	25	B
29-11-2020	23	1003	decision	Persian	20	C
28-11-2020	23	1005	transfer	Persian	22	D
28-11-2020	25	1002	decision	Hindi	11	B
27-11-2020	11	1007	decision	French	104	D
26-11-2020	23	1004	skip	Persian	56	A
25-11-2020	20	1003	transfer	Italian	45	C

1.A Number of jobs reviewed

Objective: Calculate the number of jobs reviewed per hour per day for November 2020

Query

```
SELECT COUNT(distinct job_id)/(30*24) as job_cnt
FROM op_acs.jobs
WHERE ds BETWEEN '01-11-2020' AND '30-11-2020';
```

Output

job_cnt
0.0083

Insights

Less than 0.01 jobs were reviewed each hour of the day throughout the month of November.

1.B Throughput (Number of events happening per second)

Objective: Calculate 7 day rolling average of throughput

Query

```
Select ds,
job_cnt,
ROUND(AVG(job_cnt) over (order by ds ROWS BETWEEN 6 preceding and current row),2) as sevenDayAVG
FROM (
SELECT ds, COUNT(distinct job_id) as job_cnt
FROM op_acs.jobs
WHERE ds BETWEEN '01-11-2020' AND '30-11-2020'
group by ds
order by ds) sub
```

Output

ds	job_cnt	sevenDayAVG
25-11-2020	1	1.00
26-11-2020	1	1.00
27-11-2020	1	1.00
28-11-2020	2	1.25
29-11-2020	1	1.20
30-11-2020	2	1.33

Insights

For throughput, a 7-day rolling average is better than the daily metric as it makes it easier to understand a trend over a period of time

1.C Percentage share of each language

Objective: Calculate the percentage share of each language in the last 30 days

Query

```
with cte1 as (
select *,
count(job_id) over () as total_jobs
from jobs
WHERE ds BETWEEN '01-11-2020' AND '30-11-2020'
),
cte2 as (select language, count(job_id) as num_jobs, total_jobs
from cte1
group by language)

Select language, num_jobs, round(100.0 * (num_jobs/total_jobs),2) as perc
from cte2
```

Output

language	num_jobs	perc
English	1	12.50
Arabic	1	12.50
Persian	3	37.50
Hindi	1	12.50
French	1	12.50
Italian	1	12.50

Insights

Persian Language had the highest share among other languages

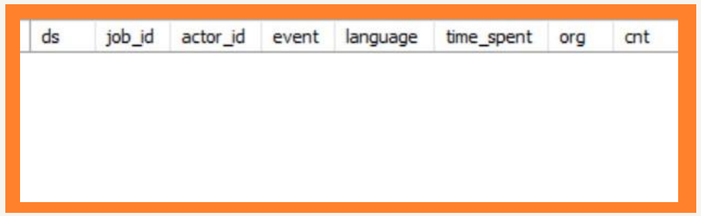
1.D Duplicate Rows

Objective: Display duplicate rows if any

Query

```
SELECT ds, job_id, actor_id, event, language, time_spent, org, COUNT(*) as cnt
FROM op_acs.jobs
GROUP BY ds, job_id, actor_id, event, language, time_spent, org
HAVING COUNT(*) >1
```

Output



ds	job_id	actor_id	event	language	time_spent	org	cnt

Insights

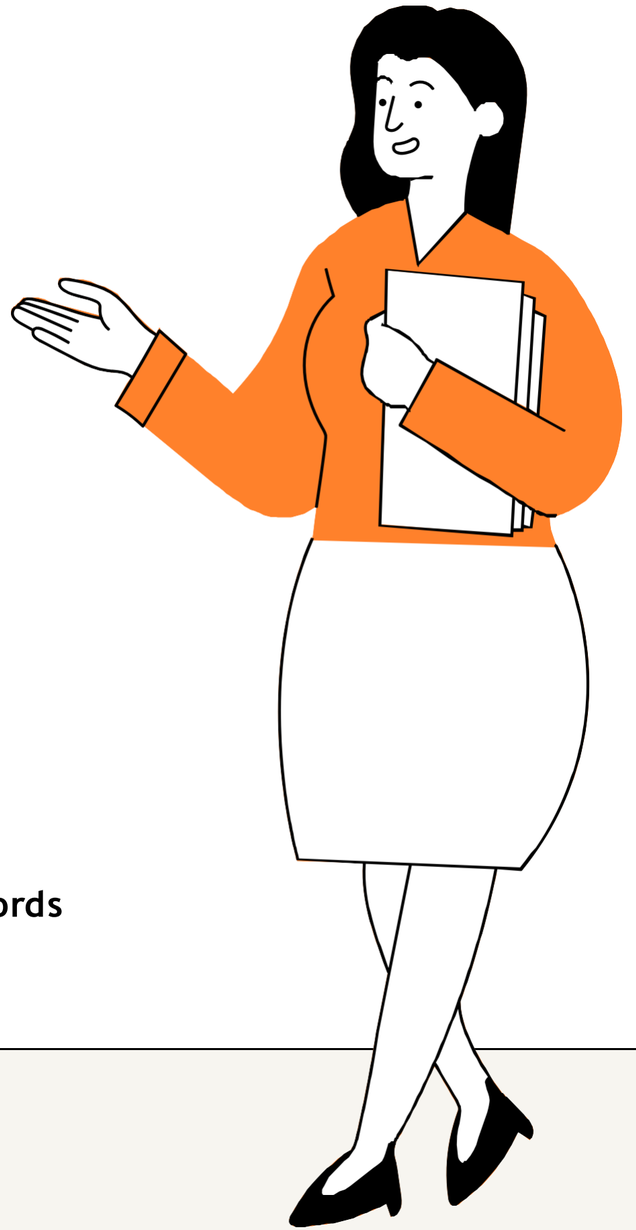
The output showed no records as there were no duplicates in the dataset

Case Study 2

Events Table

user_id	occurred_at	event_type	event_name	location	device	user_type
14573	21-06-2014 09:08	signup_flow	create_user	Canada	acer aspire desktop	
14573	21-06-2014 09:08	signup_flow	enter_email	Canada	acer aspire desktop	
14573	21-06-2014 09:09	signup_flow	enter_info	Canada	acer aspire desktop	
14573	21-06-2014 09:09	signup_flow	complete_signup	Canada	acer aspire desktop	3
14573	21-06-2014 09:09	engagement	login	Canada	acer aspire desktop	3
14573	21-06-2014 09:10	engagement	home_page	Canada	acer aspire desktop	3
14573	21-06-2014 09:10	engagement	like_message	Canada	acer aspire desktop	3
14573	21-06-2014 09:11	engagement	home_page	Canada	acer aspire desktop	3
14573	21-06-2014 09:12	engagement	like_message	Canada	acer aspire desktop	3
14573	21-06-2014 09:12	engagement	home_page	Canada	acer aspire desktop	3
14574	21-06-2014 07:11	signup_flow	create_user	Japan	lenovo thinkpad	
14574	21-06-2014 07:11	signup_flow	enter_email	Japan	lenovo thinkpad	

7 fields
340832 records



Case Study 2

Users Table

user_id	created_at	company_id	language	activated_at	state
0	01-01-2013 20:59	5737	english	01-01-2013 21:01	active
1	01-01-2013 13:07	28	english		pending
2	01-01-2013 10:59	51	english		pending
3	01-01-2013 18:40	2800	german	01-01-2013 18:42	active
4	01-01-2013 14:37	5110	indian	01-01-2013 14:39	active
5	01-01-2013 13:39	2463	spanish		pending
6	01-01-2013 18:37	11699	english	01-01-2013 18:38	active
7	01-01-2013 16:19	4765	french	01-01-2013 16:20	active
8	01-01-2013 04:38	2698	french	01-01-2013 04:40	active
9	01-01-2013 08:04	1	french		pending
10	01-01-2013 09:36	10	arabic		pending
11	01-01-2013 08:07	3745	english	01-01-2013 08:09	active

6 fields
19066 records

Case Study 2

Email Events Table

user_id	occurred_at	action	user_type
0.0	2014-05-06 09:30:00	sent_weekly_digest	1.0
0.0	2014-05-13 09:30:00	sent_weekly_digest	1.0
0.0	2014-05-20 09:30:00	sent_weekly_digest	1.0
0.0	2014-05-27 09:30:00	sent_weekly_digest	1.0
0.0	2014-06-03 09:30:00	sent_weekly_digest	1.0
0.0	2014-06-03 09:30:25	email_open	1.0
0.0	2014-06-10 09:30:00	sent_weekly_digest	1.0
0.0	2014-06-10 09:30:24	email_open	1.0
0.0	2014-06-17 09:30:00	sent_weekly_digest	1.0
0.0	2014-06-17 09:30:23	email_open	1.0
0.0	2014-06-24 09:30:00	sent_weekly_digest	1.0
0.0	2014-07-01 09:30:00	sent_weekly_digest	1.0

4 fields
19066 records

2.A User Engagement

Objective: Calculate the weekly user engagement

Query

```
select extract(week from occurred_at) as weeknum, count(distinct user_id) as cnt
from tutorial.yammer_events
group by weeknum
```

Output

weeknum	cnt
18	791
19	1244
20	1270
21	1341
22	1293

*this above table is not the full output

Insights

Week 31 posted the highest user engagement and week 18 posted the minimum user engagement

2.B User Growth

Objective: Calculate the user growth for product

**User growth have been calculated on a weekly basis.

Query

```
with cte as (select
extract(year from activated_at)||'-'||extract(week from activated_at) as weeknum,
count(distinct user_id) as activeUsers
from tutorial.yammer_users
where state = 'active'
group by weeknum )

select weeknum, activeUsers,
sum(activeUsers) over (order by weeknum rows between unbounded preceding and current row) as userGrowth
from cte
```

Output

weeknum	activeusers	usergrowth
2013-1	67	67
2013-10	43	110
2013-11	33	143
2013-12	32	175
2013-13	33	208

*this above table is not the full output

Insights

Week 35 of the year 2014 had the highest number of active users and week 2 of the year 2013 had the lowest number of active users

2.C Weekly Retention

Objective: Calculate the weekly retention of the users sign-up cohort

****assuming retention for 18th week only for the users who signed up**

Query

```
with cte1 as (
select distinct user_id,
Extract(week from occurred_at) as signup_week
from tutorial.yammer_events
where event_type = 'signup_flow'
and event_name = 'complete_signup' and extract (week from occurred_at) = 18 )

,cte2 as (select distinct user_id,
Extract(week from occurred_at) as engagement_week
from tutorial.yammer_events
where event_type = 'engagement')

select count(user_id) total_engaged_users,
sum(case when retention_week > 0 then 1 else 0 end) as retained_users
from
  (select a.user_id, a.signup_week,
    b.engagement_week, b.engagement_week-a.signup_week as retention_week
    from cte1 a
    LEFT JOIN cte2 b
    on a.user_id = b.user_id
    order by a.user_id ) sub
```

Output

total_engaged_users	retained_users
317	236

Insights

- 30% of the users retained in week 18 were retained only for the next 7 days.
- User 11816 was retained for the longest duration of 17 weeks

2.D Weekly Engagement

Objective: Calculate the the weekly engagement per device

Query

```
with cte as (select extract(year from occurred_at)||'-'||extract(week from occurred_at) as weeknum,
device, count(distinct user_id) as usercnt
from tutorial.yammer_events
where event_type = 'engagement'
group by weeknum, device
order by weeknum)

select weeknum, device, usercnt
from cte
```

Output

weeknum	device	usercnt
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

*this above table is not the full output

Insights

Weeks 31& 32 of the year 2014 had the highest user engagement of 317 users each week for the product and the device being used was 'MacBook Pro' for both the weeks

2.E Email Engagement

Objective: Calculate the email engagement metrics

Query

```
select
100.0 * 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,
100.0 *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
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 tutorial.yammer_emails ) sub
```

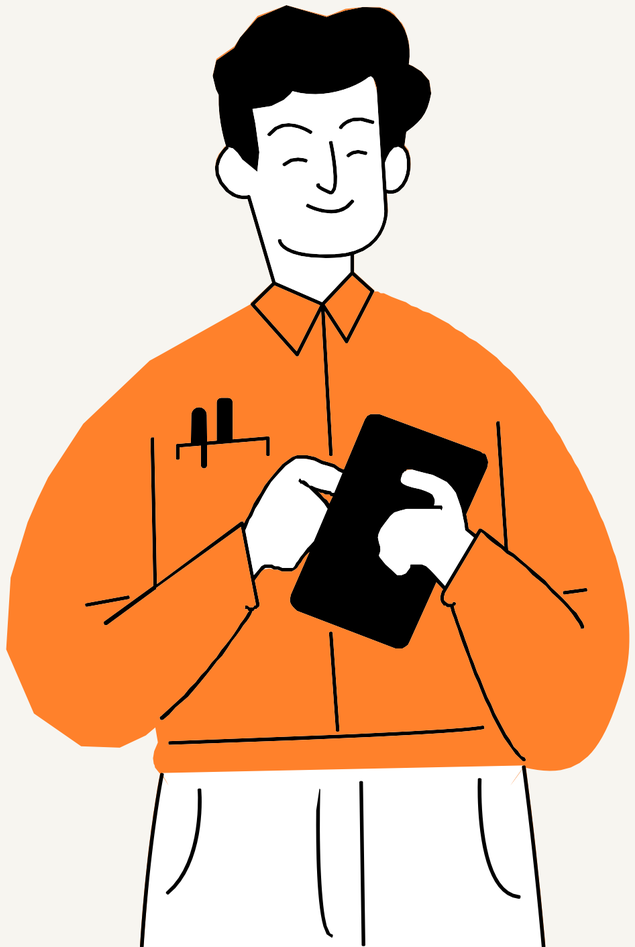
Output

email_open_rate	email_click_rate
35.7256	15.7333

Insights

Out of the total emails sent, around 35.73% of them were opened and only 15.74% of those emails were clicked

Key Insights



- Less than 0.01 jobs were reviewed each hour of the day throughout the month of November.
- Out of the total emails sent, around 35.73% of them were opened and only 15.74% of those emails were clicked
- The Persian Language had the highest share among other languages, accounting to ~38%
- Maximum retained users were only retained for a week, the retention rates dropped week-by- week
- Users who had the highest engagement with the product were operating on 'Macbook Pro'

Result

The key findigs of the project would be the identification of the reviewed jobs and how they are mapped across languages

Further, retention rates were found out and in-depth analysis was drawn upon some predefined assumptions to identify the retained users

End-to-end user engagement analysis was done which stated that ~70% of the users were engaged regularly with the product

I learned that a lot of metrics other than the monetary ones can have a huge impact on the business decision-making process

THANK YOU