OPERATION ANALYTICS & INVESTIGATING METRIC SPIKE

PROJECT DESCRIPTION:

We want to work closely with the operations, support, and marketing teams to produce insights from the data they acquire. Analysing activities of the company is one of the most important components of a business, is used to forecast the overall development or reduction of a company's fortune.

APPROACH:

The study of a company's whole end-to-end activities is known as operation analytics. This data can then be used by the organisation to identify areas for improvement. This form of research, as one of the most important components of a business, is used to forecast the overall development or reduction of a company's fortune. This translates to more efficient workflows, more cross-functional team communication, and enhanced automation.

Understanding metric spikes is also an important part of operational analytics. We must determine whether or not our sales are declining, and if so, why. Also, we want to know how many users are engaging with us, if they are active or if we need to send them emails to maintain interaction active in order to monitor daily or weekly growth in users. Moreover, based on the data, we can make informed decisions for our company and enhance sales.

TECH-STACK USED:

For this project we are using **MySQL Workbench 8.0 CE** which is an open-source, cross-platform relational database tool that enhances functionality, data modelling, SQL creation and numerous tools for configuration.

INSIGHTS:

OPERATIONAL ANALYSIS CASE STUDY

1. **Number of jobs reviewed:**

- To keep record of our daily activities and to improve productivity we are reviewing jobs on hourly basis for each day.
- With following query, we can get data for jobs reviewed for each day

```
-- jobs reviewed per hour per day
SELECT ds AS Date,
    ROUND(COUNT(DISTINCT job_id)/ SUM(time_spent)*3600) as jobs_reviewed
FROM project
WHERE ds BETWEEN '2020-11-01' AND '2020-11-30'
GROUP BY ds;
```

2. **Throughput**

- Throughput is the number of units of information that a system can process in a given length of time. It is generally used to systems spanning from computer and network systems to organisations.
- We are comparing the outcomes of average weekly throughput with daily throughput to assist us make better decisions.
- Using following query, we can compare the throughput.

```
-- daily throughput
SELECT ds AS Date,
    ROUND(COUNT(event)/ SUM(time_spent),2) AS Daily_throughput
FROM project
GROUP BY ds
ORDER BY ds;
-- 7 day throughput
SELECT ROUND(COUNT(event)/ SUM(time_spent),2) AS Weekly_throughput FROM project ;
```

3. **Percentage share of each language**

- Users provide us with job data in a variety of languages. We'd like to know how much each language contributes to our data.
- Using following query, we can get this information.

```
-- share of each language
SELECT language,
    ROUND(100.00 * COUNT(*)/total_jobs,2) AS each_share
FROM project
CROSS JOIN
    (SELECT COUNT(*) AS total_jobs FROM project) AS total
GROUP BY language;
```

4. **Duplicate rows**

- Duplicate data can occasionally lead to inaccurate business analysis. We need to know if the data contains any duplicates so that we can correctly treat it for analysis.
- With following query, we can get duplicate values from data.

```
-- duplicate values
SELECT actor_id,
    COUNT(actor_id) AS Duplicates
FROM project
GROUP BY actor_id HAVING COUNT(*) > 1;
```

INVESTIGATING METRIC SPIKE CASE STUDY

1. **User engagement**
   - To assess a user's activity level. Assessing whether a product or service is of high quality in the eyes of the user.
   - We are calculating weekly user engagement with following query

```
--  weekly user engagement
SELECT
    EXTRACT(WEEK FROM occured_at) AS week_number,
    COUNT(DISTINCT(user_id)) AS num_of_users
FROM T_event
GROUP BY week_number;
```

2. **User growth**
   - A product's user base expands with time. We want to calculate number of users growing over time for a product.
   - With following query, we can calculate the growth.

```
-- user growth for product
SELECT year, month, user_count,
   ROUND(((user_count/LAG(user_count,1) OVER (ORDER BY Month) - 1)*100),2) AS 'user_%_growth'
FROM
(SELECT
   EXTRACT(year FROM created_at) AS year,
   EXTRACT(month FROM created_at) AS Month,
   COUNT(activated_at) AS user_count FROM users
   GROUP BY Year,Month) a Order By year;
```

3. **Weekly retention**
   - Users are retained on a weekly basis after signing up for a product.
   - We can gain a better picture of if we are losing users by calculating the weekly retention of users.

```sql
-- Weekly retention of users
SELECT first_login AS Week_no,
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,
        n.first_login ,
        m.login_week,
        m.login_week - n. first_login AS week_num
FROM
(SELECT user_id,
        extract(week FROM occured_at) AS login_week
        FROM T_event
        GROUP BY user_id, login_week) m,
(SELECT user_id,
        MIN(extract(week FROM occured_at)) AS first_login
        FROM T_event
        GROUP BY user_id) n
    WHERE m.user_id = n.user_id) AS with_week_num
GROUP BY first_login ORDER BY first_login;
```

4. **Weekly engagement**
   - We want to determine a user's level of activity. For that we are weekly assessing whether a user perceives a product/service to be of high quality.
   - To display weekly engagement, we are executing following query

```sql
-- weekly engagement
select EXTRACT(week FROM occured_at) AS week_num,
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('acer aspire notebook') then user_id else null end) as 'acer aspire notebook',
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('asus chromebook') then user_id else null end) as 'asus chromebook',
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('dell inspiron notebook') then user_id else null end) as 'dell inspiron notebook',
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('htc one') then user_id else null end) as 'htc one',
count(distinct case when device in('ipad air') then user_id else null end) as 'ipad air',
count(distinct case when device in('ipad mini') then user_id else null end) as 'ipad mini',
count(distinct case when device in('iphone 4s') then user_id else null end) as 'iphone 4s',
count(distinct case when device in('iphone 5') then user_id else null end) as 'iphone 5',
count(distinct case when device in('iphone 5s') then user_id else null end) as 'iphone 5s',
count(distinct case when device in('kindle fire') then user_id else null end) as 'kindle fire',
count(distinct case when device in('lenovo thinkpad') then user_id else null end) as 'lenovo thinkpad',
count(distinct case when device in('mac mini') then user_id else null end) as 'mac mini',
count(distinct case when device in('macbook air') then user_id else null end) as 'macbook air',
count(distinct case when device in('macbook pro') then user_id else null end) as 'macbook pro',
count(distinct case when device in('nexus 10') then user_id else null end) as 'nexus 10',
 count(distinct case when device in('nexus 5') then user_id else null end) as 'nexus 5',
 count(distinct case when device in('nexus 7') then user_id else null end) as 'nexus 7',
 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('samsumg galaxy tablet') then user_id else null end) as 'samsumg galaxy tablet',
 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('samsung galaxy s4') then user_id else null end) as 'samsung galaxy s4',
 count(distinct case when device in('windows surface') then user_id else null end) as 'windows surface'
 FROM T_event
 where event_type = 'engagement'
 group by 1
 order by 1;
```

5. **Email Engagement**
   - The level of interaction our users have with our email campaigns is measured by email engagement. We classify them according to how frequently they click and how long they have been a part of our audience.
   - With following query, we can understand how are we engaging with our customers through emails.

```sql
-- emeail engagement
SELECT
    week_num,
    ROUND((email_sent/Total*100),2) AS sent_email_rate,
    ROUND((email_opened/Total*100),2) AS opened_email_rate,
    ROUND((email_clicked/Total*100),2) AS clicked_email_rate
FROM (
    SELECT EXTRACT(week FROM occured_at) AS week_num,
           COUNT(CASE WHEN action in ('sent_weekly_digest','sent_reengagement_email' ) THEN user_id ELSE NULL END) AS 'email_sent',
           COUNT(CASE WHEN action = 'email_open' THEN user_id ELSE NULL END) AS email_opened,
           COUNT(CASE WHEN action = 'email_clickthrough' THEN user_id ELSE NULL END) AS email_clicked,
           COUNT(user_id) AS Total
    FROM email_events
    Group BY 1
    ORDER BY 1) AS cal
GROUP BY 1;
```

RESULTS

OPERATIONAL ANALYSIS CASE STUDY

1. Number of Jobs reviewed

As per results, we are seeing daily growth in productivity. On November 28, we have reviewed highest number of jobs.

Date	jobs_reviewed
2020-11-25 00:00:00	80
2020-11-26 00:00:00	64
2020-11-27 00:00:00	35
2020-11-28 00:00:00	218
2020-11-29 00:00:00	180
2020-11-30 00:00:00	180

2. Throughput

We can track growth with daily throughput, but it is more difficult with weekly throughput.

Date	Daily_throughput
2020-11-25 00:00:00	0.02
2020-11-26 00:00:00	0.02
2020-11-27 00:00:00	0.01
2020-11-28 00:00:00	0.06
2020-11-29 00:00:00	0.05
2020-11-30 00:00:00	0.05

Weekly_throughput
0.03

3. Percentage share of each language

Persian has a larger market share than any other language.

language	each_share
English	12.50
Hindi	12.50
Persian	37.50
Italian	12.50
Arabic	12.50
French	12.50

4. Duplicate Rows

We are having duplicate records for actor with id 1003

actor_id	Duplicates
1003	2

INVESTIGATING METRIC SPIKE CASE STUDY

1. User Engagement

Every week, we have the following number of active users.

week_number	num_of_users
17	740
18	1260
19	1287
20	1351
21	1299
22	1381
23	1446
24	1471
25	1459
26	1509
27	1573
28	1577
29	1607
30	1706
31	1514
32	1454
33	1438
34	1443
35	118

2. User Growth

As you can see, there is a significant decline in user growth, which suggests we are losing people or registering fewer new users. We must take the required steps to increase user growth.

year	month	user_count	user_%_growth
2013	1	161	NULL
2013	2	160	-71.01
2013	3	150	-63.33
2013	4	181	NULL
2013	5	214	NULL
2013	6	213	NULL
2013	7	284	NULL
2013	8	316	NULL
2013	9	330	NULL
2013	10	390	18.18
2013	11	399	2.31
2013	12	486	21.80
2014	1	552	242.86
2014	2	409	155.63
2014	3	0	-100.00
2014	4	0	-100.00
2014	5	0	-100.00
2014	6	0	-100.00
2014	7	0	-100.00

3. Weekly retention

- Because of the results we are currently experiencing, we need to enhance user engagement if we want our business to succeed.
- When retention rates decline week over week, we must implement the necessary business changes to boost engagement.

Week_no	Week_1	Week_2	Week_3	Week_4	Week_5	Week_6	Week_7	Week_8	Week_9	Week_10	Week_11	Week_12	Week_13	Week_14	Week_15	Week_16	Week_17	Week_18
17	472	324	251	205	187	167	146	145	145	136	131	132	143	116	91	82	77	5
18	362	261	203	168	147	144	127	113	122	106	118	127	110	97	85	67	4	0
19	284	173	153	114	95	91	81	95	82	68	65	63	42	51	49	2	0	0
20	223	165	121	91	72	63	67	63	65	67	41	40	33	40	0	0	0	0
21	187	131	91	74	63	75	72	58	48	45	39	35	28	2	0	0	0	0
22	224	150	107	87	73	63	60	55	48	41	39	31	1	0	0	0	0	0
23	219	138	101	90	79	69	61	54	47	35	30	0	0	0	0	0	0	0
24	205	143	102	81	63	65	61	38	39	29	0	0	0	0	0	0	0	0
25	218	139	101	75	63	50	46	38	35	2	0	0	0	0	0	0	0	0
26	181	114	83	73	55	47	43	29	0	0	0	0	0	0	0	0	0	0
27	199	121	106	68	53	40	36	1	0	0	0	0	0	0	0	0	0	0
28	194	114	69	46	30	28	3	0	0	0	0	0	0	0	0	0	0	0
29	186	102	65	47	40	1	0	0	0	0	0	0	0	0	0	0	0	0
30	202	121	78	53	3	0	0	0	0	0	0	0	0	0	0	0	0	0
31	145	76	57	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
32	188	94	8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
33	202	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
34	44	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

4.Weekly Engagement

With following results, we can say that most of our customers engaging through iphone 5, Lenovo thinkpad & mackbook air.

week_num	acer aspire desktop	acer aspire notebook	amazon fire phone	asus chromebook	dell inspiron desktop	dell inspiron notebook	hp pavilion desktop	htc one	ipad air	ipad mini	iphone 4s
17	9	20	4	21	18	46	14	16	27	19	21
18	26	33	9	42	58	77	37	19	52	30	46
19	23	41	12	27	36	83	40	30	55	36	44
20	23	40	11	41	52	84	30	29	59	32	55
21	29	47	5	38	41	80	44	21	51	23	45
22	25	41	5	52	52	92	38	24	58	34	45
23	22	43	16	49	53	103	54	20	41	33	53
24	24	40	11	43	59	99	56	20	57	39	53
25	28	47	13	38	52	105	52	21	57	30	40
26	29	35	13	49	60	89	46	23	56	43	50
27	29	49	10	52	53	89	56	27	55	35	67
28	30	49	6	50	56	103	56	26	54	35	61
29	28	53	12	49	54	113	58	31	52	34	60
30	33	60	12	56	54	127	42	31	70	35	65
31	31	55	14	56	44	113	51	13	55	27	56
32	35	55	12	62	57	104	51	18	48	30	34
33	39	46	14	49	37	110	38	19	40	28	35
34	30	63	11	47	49	105	36	25	39	25	50
35	1	3	0	6	1	9	1	2	0	2	6

week_num	iphone 5	iphone 5s	kindle fire	lenovo thinkpad	mac mini	macbook air	macbook pro	nexus 10	nexus 5	nexus 7	nokia lumia 635	samsumg galaxy tablet	samsung galaxy note	samsung galaxy s4	windows surface
17	65	42	6	86	6	54	143	16	40	18	17	8	7	52	10
18	113	73	27	153	13	121	252	30	73	30	33	11	15	82	10
19	115	79	21	178	18	112	266	25	87	41	23	6	11	91	16
20	125	79	23	173	26	119	256	22	103	32	22	9	18	93	21
21	137	74	30	167	18	110	247	25	91	29	25	6	20	84	17
22	125	71	21	176	25	145	251	27	96	45	25	10	19	105	15
23	152	79	25	176	18	124	266	45	88	36	31	14	14	99	14
24	142	79	25	165	29	152	255	38	87	49	35	11	20	101	22
25	137	78	24	197	21	121	275	29	89	51	37	12	14	99	22
26	152	94	26	192	11	134	269	29	87	46	42	12	9	112	21
27	163	83	25	202	15	142	302	37	84	40	31	15	15	116	33
28	151	93	31	220	28	148	295	26	85	39	35	9	10	122	33
29	144	90	37	209	31	148	295	25	77	45	43	13	16	123	28
30	152	103	25	206	23	159	322	36	84	62	34	9	15	103	19
31	135	71	14	207	24	147	321	24	69	38	28	8	14	100	19
32	119	67	12	179	20	125	307	30	67	25	28	6	12	82	10
33	110	65	14	191	32	133	312	23	70	30	27	12	13	80	15
34	101	70	13	193	30	136	292	25	70	33	17	14	13	90	18
35	2	3	3	16	2	10	17	2	4	2	2	0	1	6	3

5.Email Engagement

As you can see, the findings show a weekly decline in email engagement. In order to keep our users, we must increase email engagement.

week_num	sent_email_rate	opened_email_rate	clicked_email_rate
17	62.32	21.28	11.39
18	63.45	22.24	10.49
19	62.16	22.67	11.13
20	61.62	22.64	11.43
21	63.52	22.82	9.97
22	63.59	21.56	10.66
23	62.39	22.34	11.18
24	61.61	22.92	10.99
25	63.77	21.79	10.54
26	62.99	22.22	10.61
27	62.24	22.49	11.37
28	62.92	22.48	10.77
29	63.98	21.71	10.51
30	62.29	23.24	10.59
31	65.27	23.25	7.66
32	66.59	22.85	7.14
33	64.73	23.10	7.91