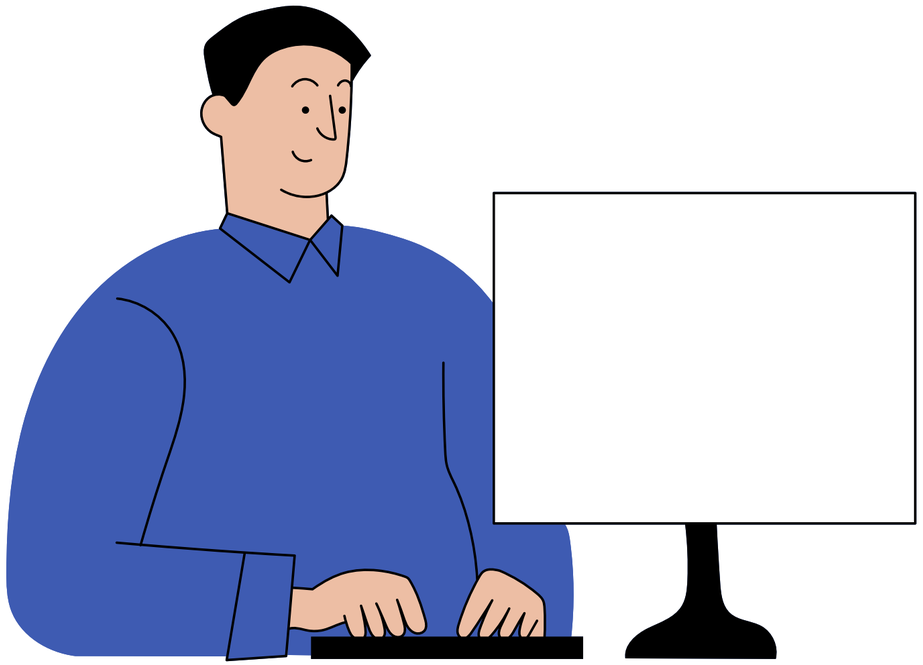
Operation and Metric Analysis

Agenda



- Project Description and Approach
- Approach
- Case Study -I on Job Data
- Case Study - II on Investgating metric spike

Project Description

As a Data Analyst working for Microsoft, I am required to provide the insights from various datasets that has been provided and answer the different questions asked by different departments.

As a part of this, I am assigned with two case studies as mentioned below:

- Job data
- Investigating Metric spike

Approach

- We have been given datasets for each case study.
- For the first case study, since the data is very little, I have used excel to create more random values and imported the file to MySQL to accomplish the tasks assigned and MySQL Command Line Client to share query results.
- For the second case study, since the datasets are huge and already on mode.com, I used that platform for querying Used basic, aggregate and window functions.
- Took some time to understand the tables and the data that is stored within them before moving forward.
- Used Canva to make this presentation.

Case Study -I : Job Data

Note : The dataset that has been imported from CSV has 300 rows after random insertion of data.

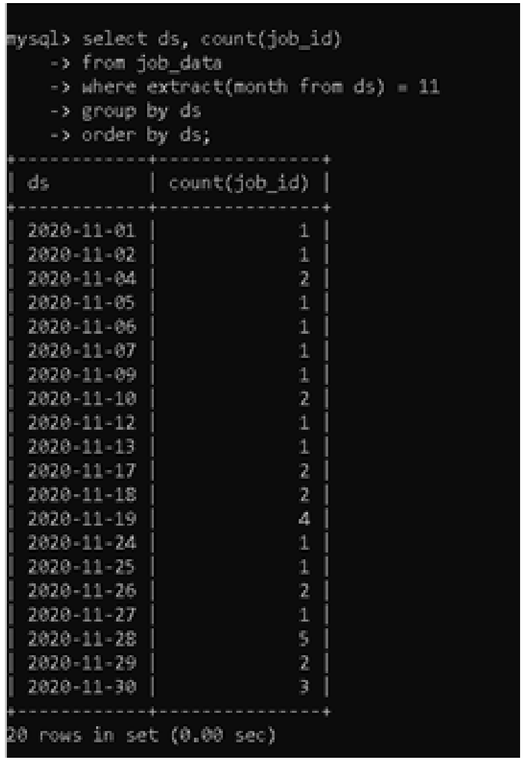
- Table : job_data
  - job_id: unique identifier of jobs
  - actor_id: unique identifier of actor
  - event: decision/skip/transfer
  - language: language of the content
  - time_spent: time spent to review the job in seconds
  - org: organization of the actor
  - ds: date in the yyyy/mm/dd format. It is stored in the form of text and we use presto to run. no need for date function

Task-I : Jobs Reviewed per day

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

Query:

```
select ds, count(job_id)
from job_data
where extract(month from ds) = 11
group by ds
order by ds;
```



```
mysql> select ds, count(job_id)
    -> from job_data
    -> where extract(month from ds) = 11
    -> group by ds
    -> order by ds;
+------------+---------------+
| ds         | count(job_id) |
+------------+---------------+
| 2020-11-01 |             1 |
| 2020-11-02 |             1 |
| 2020-11-04 |             2 |
| 2020-11-05 |             1 |
| 2020-11-06 |             1 |
| 2020-11-07 |             1 |
| 2020-11-09 |             1 |
| 2020-11-10 |             2 |
| 2020-11-12 |             1 |
| 2020-11-13 |             1 |
| 2020-11-17 |             2 |
| 2020-11-18 |             2 |
| 2020-11-19 |             4 |
| 2020-11-24 |             1 |
| 2020-11-25 |             1 |
| 2020-11-26 |             2 |
| 2020-11-27 |             1 |
| 2020-11-28 |             5 |
| 2020-11-29 |             2 |
| 2020-11-30 |             3 |
+------------+---------------+
20 rows in set (0.00 sec)
```

Task-II : Throughput (It is number of events happening per second)

Calculate 7 day rolling average of throughput? For throughput, do you prefer daily metric or 7-day rolling and why?

Query:

```
select ds, event, count(event), avg(count(event)) over() as
avg_eventcount, avg(count(event)) over (partition by event
rows between 6 preceding and current row) as
7_day_rolling_avg
from job_data
group by ds;
```

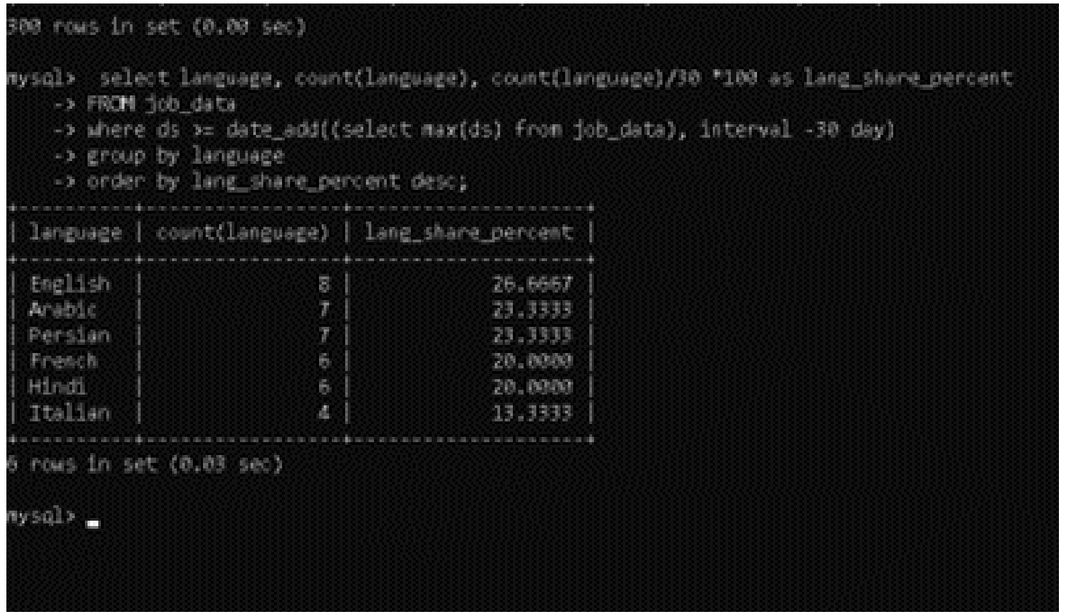
```
mysql> select ds, event, count(event), avg(count(event)) over() as avg_eventcount, avg(count(event)) ov
    -> from job_data
    -> group by ds;
+------------+----------+--------------+----------------+------------------+
| ds         | event    | count(event) | avg_eventcount | 7_day_rolling_avg |
+------------+----------+--------------+----------------+------------------+
| 2020-06-17 | decision |            2 |         1.7647 |           2.0000 |
| 2020-11-29 | decision |            2 |         1.7647 |           2.0000 |
| 2020-11-10 | decision |            2 |         1.7647 |           2.0000 |
| 2020-11-27 | decision |            1 |         1.7647 |           1.7500 |
| 2020-09-07 | decision |            4 |         1.7647 |           2.2000 |
| 2020-12-09 | decision |            1 |         1.7647 |           2.0000 |
| 2020-06-12 | decision |            1 |         1.7647 |           1.8571 |
| 2020-08-19 | decision |            1 |         1.7647 |           1.7143 |
| 2020-08-08 | decision |            3 |         1.7647 |           1.8571 |
| 2020-12-14 | decision |            2 |         1.7647 |           1.8571 |
| 2020-12-31 | decision |            1 |         1.7647 |           1.8571 |
| 2020-05-28 | decision |            2 |         1.7647 |           1.5714 |
| 2020-06-06 | decision |            1 |         1.7647 |           1.5714 |
| 2020-07-07 | decision |            1 |         1.7647 |           1.5714 |
| 2020-10-08 | decision |            2 |         1.7647 |           1.7143 |
| 2020-06-21 | decision |            2 |         1.7647 |           1.5714 |
| 2020-12-16 | decision |            1 |         1.7647 |           1.4286 |
| 2020-06-02 | decision |            1 |         1.7647 |           1.4286 |
| 2020-10-18 | decision |            2 |         1.7647 |           1.4286 |
| 2020-11-09 | decision |            1 |         1.7647 |           1.4286 |
| 2020-08-26 | decision |            1 |         1.7647 |           1.4286 |
| 2020-07-25 | decision |            1 |         1.7647 |           1.2857 |
| 2020-11-07 | decision |            1 |         1.7647 |           1.1429 |
| 2020-05-30 | decision |            2 |         1.7647 |           1.2857 |
| 2020-07-12 | decision |            5 |         1.7647 |           1.8571 |
| 2020-10-21 | decision |            3 |         1.7647 |           2.0000 |
| 2020-12-17 | decision |            2 |         1.7647 |           2.1429 |
| 2020-05-04 | decision |            2 |         1.7647 |           2.2857 |
| 2020-05-09 | decision |            1 |         1.7647 |           2.2857 |
| 2020-08-09 | decision |            2 |         1.7647 |           2.4286 |
| 2020-06-25 | decision |            1 |         1.7647 |           2.2857 |
| 2020-09-05 | decision |            2 |         1.7647 |           1.8571 |
| 2020-12-13 | decision |            2 |         1.7647 |           1.7143 |
```

Task-III : Percentage share of each languauge

Calculate the percentage share of each language in the last 30 days?

Query:

```
select language, count(language), count(language)/30
*100 as lang_share_percent
from job_data
where ds >= date_add((select max(ds) from job_data),
interval -30 day)
group by language
order by lang_share_percent desc;
```



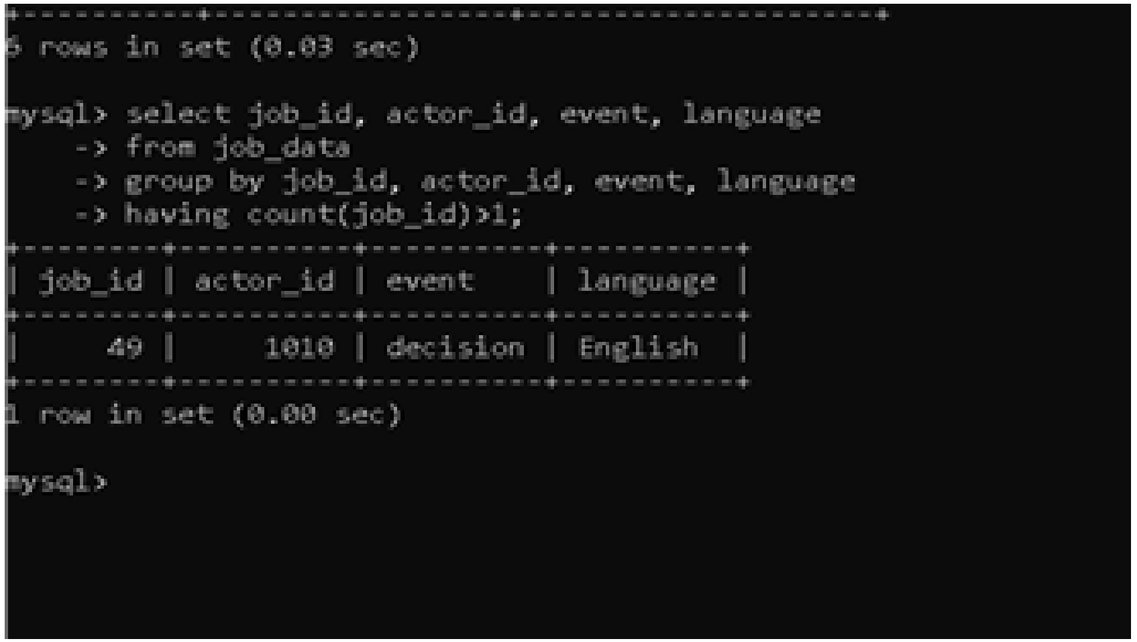
```
300 rows in set (0.00 sec)

mysql>  select language, count(language), count(language)/30 *100 as lang_share_percent
    -> FROM job_data
    -> where ds >= date_add((select max(ds) from job_data), interval -30 day)
    -> group by language
    -> order by lang_share_percent desc;
+----------+-----------------+---------------------+
| language | count(language) | lang_share_percent  |
+----------+-----------------+---------------------+
| English  |               8 |             26.6667 |
| Arabic   |               7 |             23.3333 |
| Persian  |               7 |             23.3333 |
| French   |               6 |             20.0000 |
| Hindi    |               6 |             20.0000 |
| Italian  |               4 |             13.3333 |
+----------+-----------------+---------------------+
6 rows in set (0.03 sec)

mysql>
```

Task-IV : Duplicate Rows

Rows that have the same value present in them.

Query:

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



Case Study II: Investigating Metric Spike

Questions asked by different departments

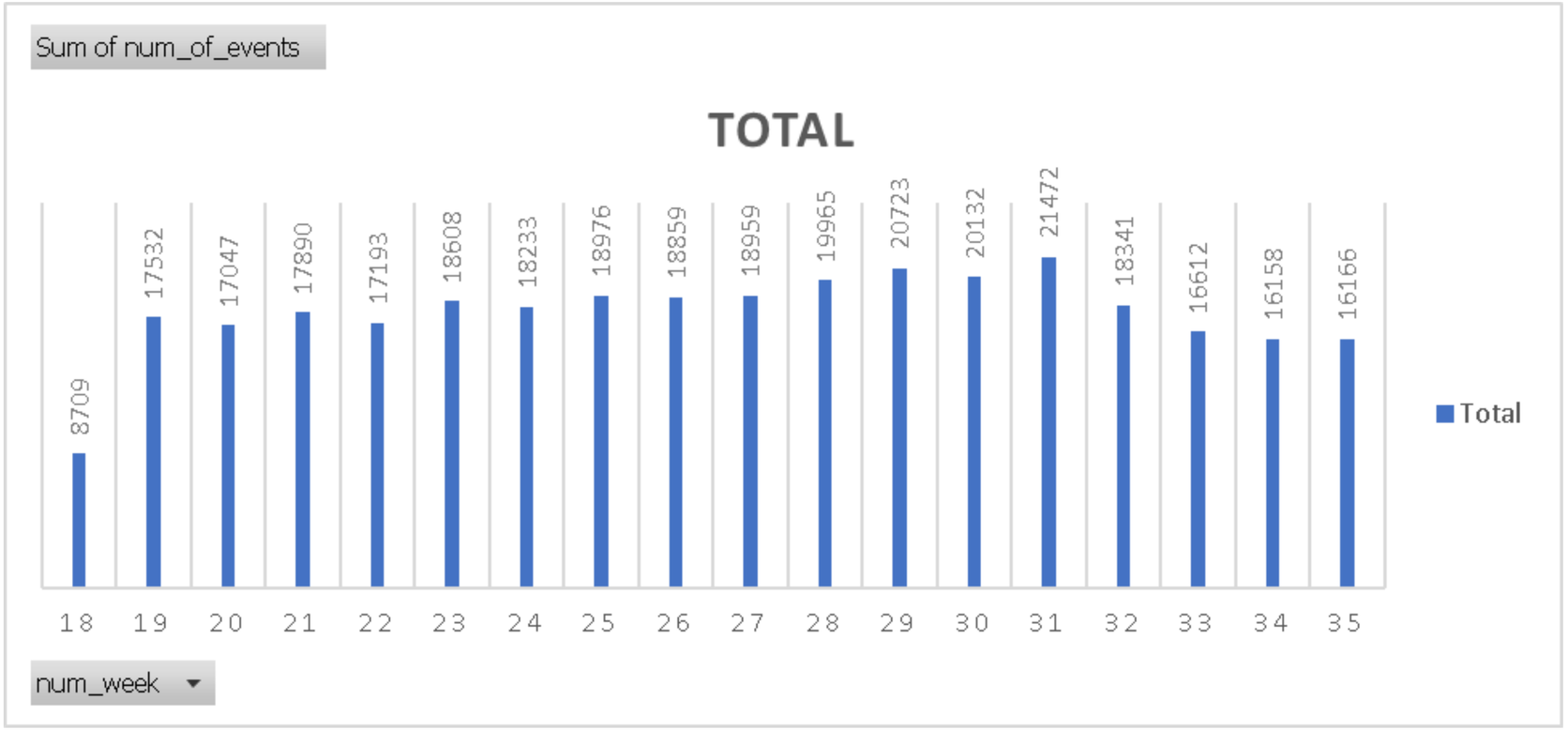
01 Weekly User Engagement

02 User Growth for product

03 Weekly retention of user sign-up cohort

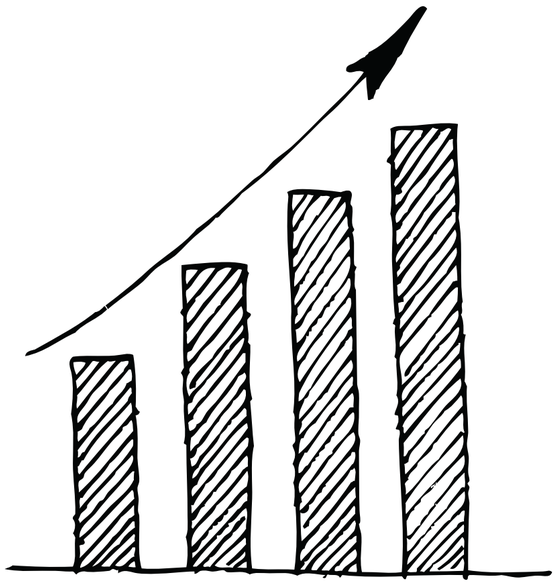
04 Email Engagement metrics

Weekly User Engagement

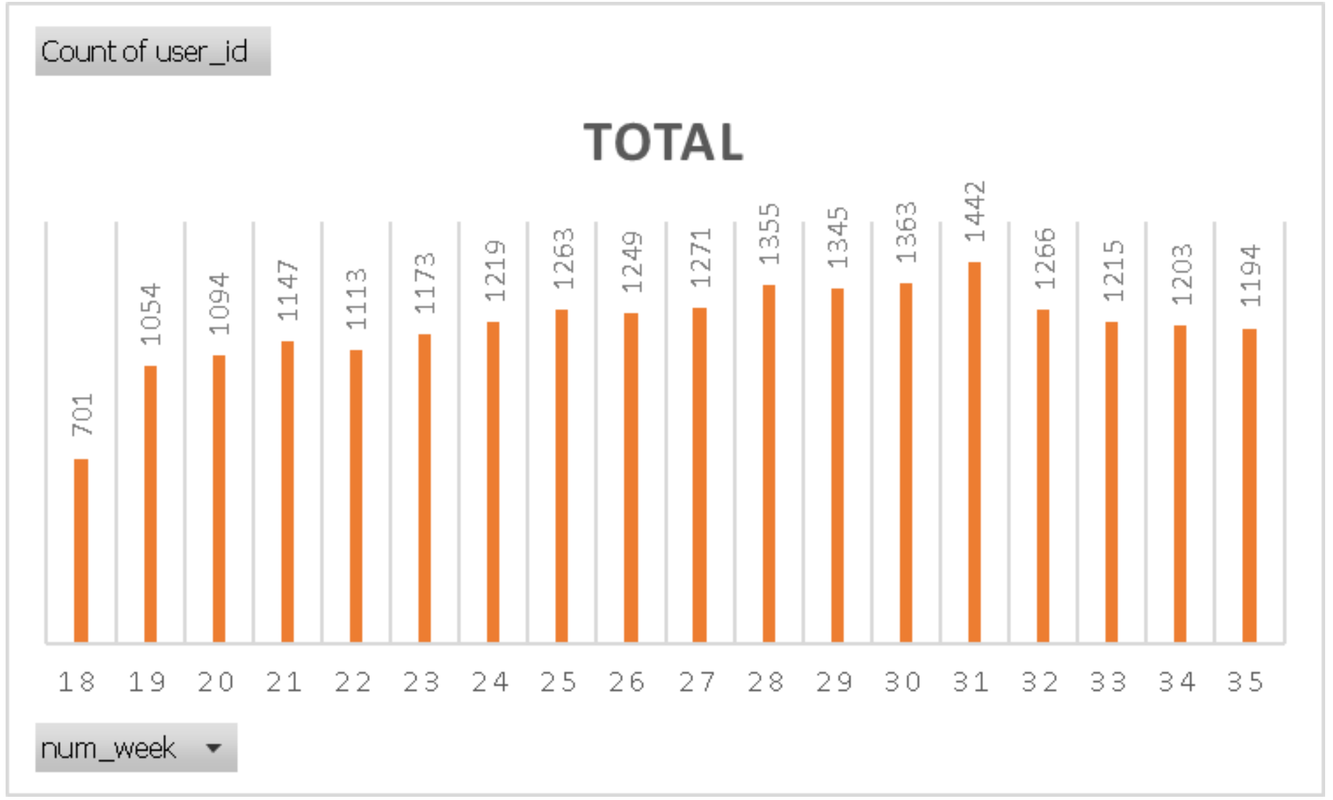


User Growth for product

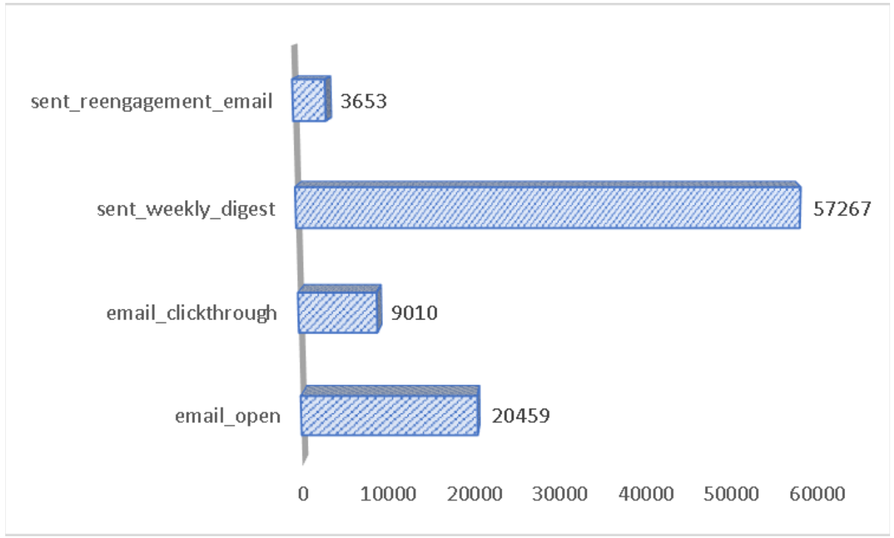
Data shows about 3,680 new users completed the 'sign-up' process

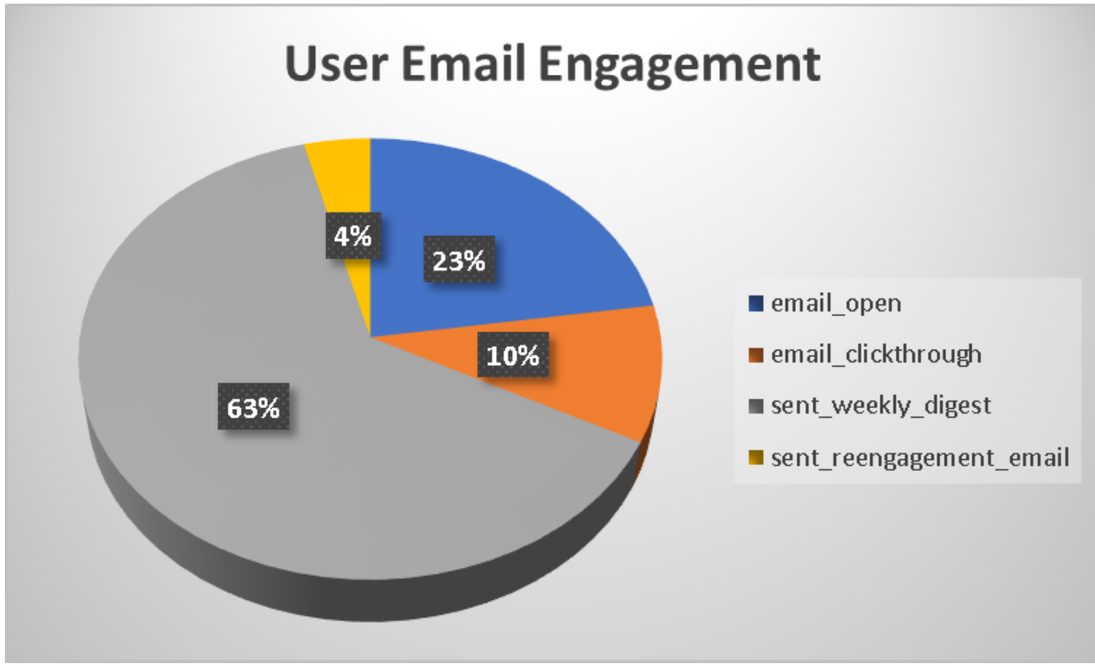


Weekly Retention of user sign-up cohort



Email Engagement metrics





That's a wrap!