# Operation Analytics and Investigating Metric Spike

**Name: Kaila Sai Pranava Karthik**

## Project Description:

The project called "Operational Analytics and Metric Spike Investigation" aims to analyze all aspects of a companys operations. It focuses on identifying fluctuations or spikes, in metrics. The main goal is to pinpoint areas that need improvement. We will use SQL queries to extract insights, which will then be presented as information to different departments within the organization. The project involves analyzing metrics such, as the number of jobs reviewed, language distribution, throughput duplicate data rows, user engagement, weekly retention, user growth, weekly engagement and email engagement.

## Approach:

The project's data analysis primarily relies on SQL Server. It begins with a thorough understanding of the provided dataset. A dedicated "operation" database is created, housing essential tables constructed using data from provided links. Sample records are incorporated as recommended in the forum. Utilizing SQL queries, the project derives valuable insights.
Case Study 1 focuses on metrics like daily job reviews for November 2020, 7-day rolling throughput averages, language distribution in the last 30 days, and the identification of duplicate rows.
Case Study 2 centers on investigating metric spikes, analyzing user engagement, growth, retention, weekly engagement per device, and email engagement metrics through queries on relevant tables (users, events, email_events). The objective is to identify patterns and trends that can enhance overall operational performance.

## Tech Stack Used:

For this project I utilized SQL Server and MS Excel as my tech stack. SQL Server was employed to interact with the database, including tasks such, as creating tables based on the provided data executing queries and effectively visualizing the results. To begin with I added records, for the dataset 1 in a file before importing them into SQL Server.

**Execution:**
**Case Study 1 (Job Data):**

A. **Number of jobs reviewed:** Amount of jobs reviewed over time.
**My task:** Calculate the number of jobs reviewed per hour per day for November 2020?

```
select
count(distinct job_id)//(30*24) as num_jobs_reviewedfrom job_data
where
ds between '2020-11-01' and '2020-11-30'
```

B. **Throughput:** It is the no. of events happening per second.
**My 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?

```
select ds, jobs_reviewed,
avg(jobs_reviewed)over(order by ds rows between 6 preceding and current row)as
throughput_7_rolling_avg
from(
select ds, count(distinct job_id) as jobs_reviewedFrom job_data
where ds between '2020-11-01' and '2020-11-30'group by ds
order by ds
)a;
```

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

```
select language, num_jobs,
100.0* num_jobs/total_jobs as pct_share_jobsfrom
(
select language, count(distinct job_id) as num_jobsfrom job_data
group by language
)a
cross join(
select count(distinct job_id) as total_jobsfrom job_data
)b;
```

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

```
select * from(
select *,
row_number()over(partition by job_id) as rownumfrom
job_data
)a
where rownum>1;
```

**Case Study 2 (Investigating metric spike):**

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

```
select
    extract(week from occurred_at) as num_week, count(distinct
    user_id) as no_of_distinct_user
from tutorial.yammer_eventsgroup
by num_week;
```

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

```
select year, num_week, num_active_users,
sum(num_active_users) over(order by year, num_week rows between unboundedpreceding and
current row)
as cumm_active_users
from
(select
      extract(year from a.activated_at) as year, extract(week from
      a.activated_at)as num_week,count(distinct user_id) as
      num_active_users
from tutorial.yammer_users a
where state='active' group by
year, num_weekorder by year,
num_week
)a;
```

C. **Weekly Retention:** Users getting retained weekly after signing-up for a product.
**My task:** Calculate the weekly retention of users-sign up cohort?

```
select count(user_id),
          sum(case when retention_week = 1 then 1 else 0 end) as
per_week_retention
from(
select a.user_id,
          a.sign_up_week,
          b.engagement_week,
          b.engagement_week - a.sign_up_week as retention_week
from(
(select distinct user_id, extract(week from occured_at) as sign_up_weekfrom
tutorial.yammer_events
where event_type = 'signup_flow' and
event_name = 'complete_signup'
and extract(week from occured_at)=18)aleft join
(select distinct user_id, extract(week from occured_at) as engagement_weekfrom
tutorial.yammer_events
where event_type = 'engagement')bon
a.user_id = b.user_id
)
group by user_id order
by user_id;
```

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

```
select
extract(year from occured_at) as year_num,extract(week
```

```
from occured_at) as week_num,device,
count(distinct user_id) as no_of_usersfrom
tutorial.yammer_events
where event_type = 'engagement'group
by 1,2,3
order by 1,2,3;
```

E. **Email Engagement:** Users engaging with the email service.
**My task:** Calculate the email engagement metrics?

```
select
100.0 * sum(case when email_cat = 'email_opened' then 1 else 0 end)
            /sum(case when email_cat = 'email_sent' then 1 else 0 end)as
email_opening_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_clicking_ratefrom
(
select *,
case when action in ('sent_weekly_digest', 'sent_reengagement_email')then 'email_sent'
        when action in ('email_open')then
        'email_opened'
        when action in ('email_clickthrough')then
        'email_clicked'
end as email_cat
from tutorial.yammer_events
)a;
```

# Insights:

**Case Study 1 (Job Data):**

- In November 2020, an average of 83% of distinct jobs were reviewed per hour per day.
- We used the 7-day rolling average of throughput because it provides a more accurate representation of the average throughput over time than the daily metric, which only considers the throughput for a single day.
- The Persian language has the highest percentage share (37.5%).
- There are two duplicate rows if we partition the data by job_id, but all rows are unique if we consider all columns.

**Case Study 2 (Investigating Metric Spike):**

- Weekly user engagement increased from week 18 to week 31, but then started declining. This suggests that some users may not be finding the product or service as valuable in the later weeks.
- There are a total of 9381 active users from the 1st week of 2013 to the 35th week of 2014.
- MacBook and iPhone users have the highest overall weekly engagement count per device used.
- The email opening rate is around 34% and the email clicking rate is around 15%. This indicates that users are engaging with the email service, which is positive for the company's expansion plans.

## Result:

This project has provided me with valuable practical experience in performing operational analytics and investigating metric spikes. I have also gained a deeper understanding of data analysis techniques and the application of SQL queries for extracting valuable insights. This practical experience has helped me to better understand how analytics and data-driven decision-making are implemented in real-world scenarios.

In addition, this project has helped me to master my SQL skills and learn how to apply advanced SQL concepts such as window functions. I have also gained a better understanding of how the real-world industry works and how to ask the right questions given the circumstances. I have learned how to identify the relevant columns in a given dataset to answer a specific question and extract valuable insights that can help businesses grow. Finally, I have learned how companies identify and improve different areas of their operations. Overall, this project has been a valuable learning experience that has helped me to develop the skills and knowledge necessary to be a successful data analyst.