--> **Initial thoughts and progress**
Instagram, TikTok, LinkedIn, Quora, Reddit

-->Instagram

First, we decided to explore Instagram which is the most used social network platform globally. After looking deeper into the application, we came up with users being nodes and if they follow each other then there will be an edge between them. Although this is a nice strategy, after reading the Meta API documentation we got to know that we cannot get the random user data, however if we have the username associated to a profile, we can obtain the userid and then get the necessary information such as list of followers. Also, we cannot get the followers list if the user is a private account.

Ref: Here's a short youtube video on getting started with the Meta-API.

From google we got the top 100 list of instagram users and their usernames.

Ref: Medium article about data extraction. sheets --> from this website

Google sheets' IMPORTJSON function with the Insta URL is supposed to extract the profile data if username or url of that instagram page is known.

Next step is to check the userids of the users.

Ref: Query the instagram API to get user id based on their username

From this repo we referenced a piece of code to check the userids of the usernames with us.

```
/**
* Get a user ID from username via API
* @param String $username username to search
* @return Int Instagram userid
*/
private function getUserID($username) {
   $ch = curl_init();
   curl_setopt($ch, CURLOPT_TIMEOUT, 20);
   curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
   curl_setopt($ch, CURLOPT_TIMEOUT, 20);
   curl_setopt($ch, CURLOPT_URL,
"https://api.instagram.com/v1/users/search?q=$username&client_id=".$this-
>client_id);
   $returned_users = curl_exec($ch);
   curl_close($ch);
   // Loop through the results and find the exact match
   foreach (json_decode($returned_users)->data as $row) {
      if($row->username == $username) {
         $user_id = $row->id;
         continue;
      }
```

```
        }
        return $user_id;
    }
```

We made a simple script in python to get the data from the sheets and iterate over to get the userids. This gives us the user ids in a separate excel file.

```
import pandas as pd
import requests
# Loading Excel file
excel_path = 'file path in computer'
df = pd.read_excel(excel_path)
# Instagram access token
access_token = 'INSTA_ACCESS_TOKEN'
# Function to get user ID from username
def get_user_id(username, token):
    url =
f"https://graph.instagram.com/v11.0/{username}?fields=id&access_token={token}"
    response = requests.get(url)
    if response.status_code == 200:
        data = response.json()
        return data.get("id")  # if id is what we need here
    else:
        return None
# Iterate over usernames and fetch user IDs
user_ids = []
for username in df['username_column_name']:
    user_id = get_user_id(username, access_token)
    if user_id:
        user_ids.append(user_id)
    else:
        print(f"Failed to fetch user ID for username: {username}")
# Optional: Save the user IDs back to the Excel file
df['user_id'] = user_ids
df.to_excel('updated_user_ids.xlsx', index=False)
```

Problems encountered: The idea about Instagram is promising but the rules and regulations set by the company Meta doesn't allow us to do what we intended to do. Additional information can be found here . The approach we deduced after reading the documentation has a lot of manual work like collecting the usernames of the people/pages. So, we decided to move to a different social network platform.

--> TikTok

The tiktok's api won't let us use it unless we request access. To request access, we need to be qualified as a PhD student or a university teaching staff. So, we decided to move on to the next one. Information can be found here.

LinkedIn and Quora also had similar accessibility issues, post which we decided to pursue Reddit which had excellent documentation and support.

--> Reddit

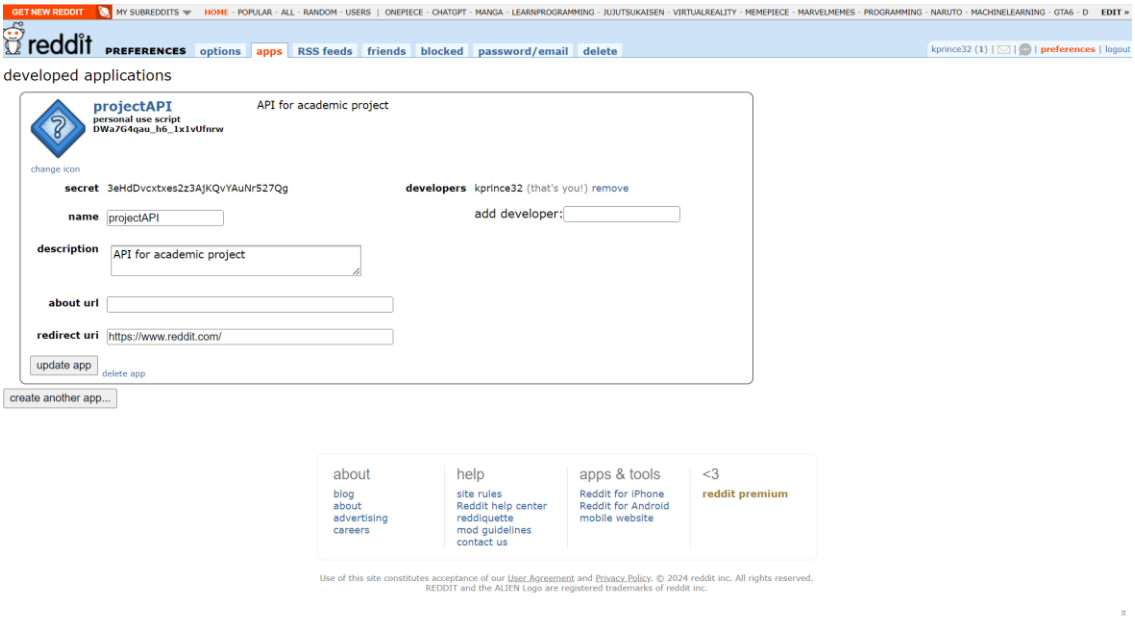
Problem Statement: To identify the most popular and impactful subreddits and users who are well-connected with others, as the success of publicizing anything in these subreddits significantly impacts its reach.

We initially believed we would be working on a diffusion network as we considered the comments and replies on a post. As the problem statement evolved, we ended up with a friendship network where the users are nodes, and the edges are the subreddits they've commonly participated in.

Approach:

We first requested access for the Reddit Api using the reddit apps portal where we created an app to get the secret key and client id for the API access.

The details can be found below:



client_id='DWa7G4qau_h6_1x1vUfnrw',

client_secret='3eHdDvcxtxes2z3AjKQvYAuNr527Qg',

user_agent='projectAPI'

User policy for reddit can be found here.

Our approach is to take a set of popular subreddits and get the data from the posts. We're taking the user_data of post author, commentors and other commentors replying to the first level comments.

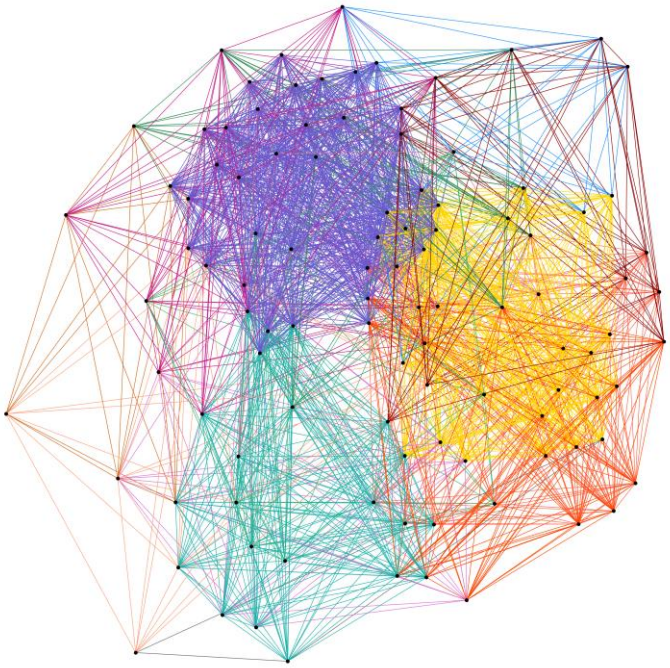
After writing the code for the same, we're able to generate the usernames of the users in a dataframe. And this dataframe is stored and exported in a csv file.

Next step is cleaning and processing the data by storing the necessary usernames in a separate dataframe where the users interacted with multiple subreddits atleast once. This also contains how many occurrences the user was present in and what subreddits the user is associated with.

We used NetworkX to generate the graphs as it had great support for Pythis the language we used to work on the entire project.

From the above data we created a graph where we put users as nodes and subreddits as edges. If two users are associated in same subreddits then an edge is formed between them. Note that we're considering only the users that have interaction in multiple subreddits.

The graph is as follows

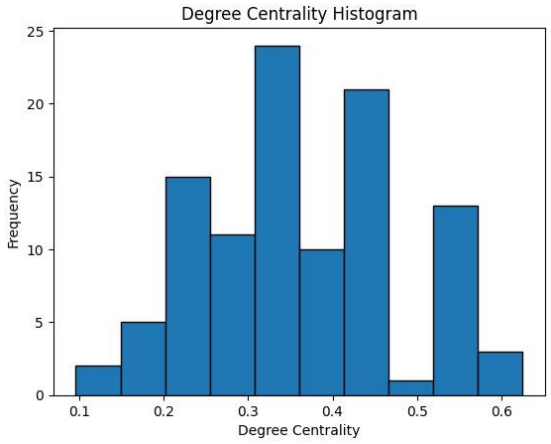


Here the color of edges represents the different subreddits.

Metrics:

We derived the degree centrality, betweenness centrality, and the clustering coefficient. This was done by parsing the CSV in which we stored the previous results, using the NetworkX functions to derive the numbers. We also attributed the corresponding subreddit and the user for these numbers to help derive the final insights.

We represented the degree centrality using a histogram, and the other two by their face value. Degree centrality varies between 0-1 based on the connectivity that the node has; 0 for no edges and 1 for edges connected to all nodes.



Results:

Based on all the generated metrics, we get to identify the subreddits with the most popularity, impact and users who are well connected with other subreddits.

Publicizing anything in these subreddits has better reach than the ones analyzed by us.

This was derived from the degree centrality, betweenness centrality, and clustering coefficient.

With a larger dataset, we could be getting more insights along these lines.

For example, in the most active subreddits, is the activity contributed by only a set of individuals or different individuals? How can the subreddits be made more active – are there any patterns or reasons why some are more active than the others? Etc.