Group 2 Members:

Carreon, Ma. Addine Anne T.

Lucas, Sydney Anne V.

Peralta, Nathalya O.

Quipit, Dale Russel

Dataset: https://snap.stanford.edu/data/ego-Facebook.html

Data-Driven Strategies to Increase User Engagement on Facebook

This dataset comprises anonymized "circles" or "friends lists" sourced from Facebook, obtained through a dedicated app used by survey participants. It offers a rich array of node features representing user profiles, circles, and ego networks. Through anonymization, original user IDs were replaced with new identifiers, and feature vectors were obscured for privacy preservation. Despite this anonymization, the dataset retains its value for analyzing shared characteristics among users. Additionally, key statistics underscore its scale, with 4039 nodes, 88234 edges, an average clustering coefficient of 0.6055, and a diameter of 8.

Variables	Description
fb	DataFrame containing the Facebook social network data with columns "start_node" and "end_node".
G	NetworkX graph created from the DataFrame fb representing the social network.
pos	Dictionary containing the node positions for the network visualization, generated using Fruchterman-Reingold layout.
fig, ax	Matplotlib figure and axis objects for creating plots.

plot_options	Dictionary containing options for plotting the network graph, including node size, label presence, color, and edge properties.
shortest_path_length	Float variable storing the shortest path length between two nodes in the graph G.
longest_path_length	Float variable storing the longest path length between two nodes in the graph G.
shortest_path_node	Tuple containing the nodes with the shortest path in the graph G.
longest_path_node	Tuple containing the nodes with the longest path in the graph G.
degrees	List containing the degrees of all nodes in the graph G.
g_min_graph	Integer variable storing the minimum degree in the graph G.
g_max_graph	Integer variable storing the maximum degree in the graph G.
g_avg_graph	Float variable storing the average degree in the graph G.
degree_distribution	List containing the degree distribution of the graph G.
total_nodes	Integer variable storing the total number of nodes in the graph G.
degree_distribution_percentage	List containing the percentage of nodes for each degree in the graph G.
degree_centrality	Dictionary containing the degree centrality values of nodes in the graph G.

sorted_degrees	List of tuples containing nodes sorted by degree centrality in descending order.
betweenness_centrality	Dictionary containing the betweenness centrality values of nodes in the graph G.
sorted_betweenness	List of tuples containing nodes sorted by betweenness centrality in descending order.
closeness_centrality	Dictionary containing the closeness centrality values of nodes in the graph G.
sorted_closeness	List of tuples containing nodes sorted by closeness centrality in descending order.

Case/Scenario: As a data scientist employed by Facebook, you've been assigned a project by the marketing department. Your task is to examine a dataset containing anonymized information about user interactions on the Facebook platform.

Question: How can we enhance engagement and identify influential users on Facebook?