Reddit Data Analysis: Coursework Report

Dataset:

- comments.json - This file contains data about the comments posted by users on the Reddit platform from subreddit InvestmentClub.
- submissions.json - This file contains data about the submissions (posts) made by users on the Reddit platform from subreddit InvestmentClub.

Task - 1: Data Aggregation

Describe how did you aggregate data from two files and organized the data for further processing.

We began by reading two key files, comments.json and submissions.json, which contain the data about the comments and submissions, respectively. These JSON files were loaded into two separate pandas DataFrames: commentDf and submissionDf.

Data Aggregation Process:

1. After loading the data, the next step involved sorting both commentDf and submissionDf based on the created_ts (derived from created_utc column) column, which contains timestamps for each comment and submission
2. We created a new column replied_to to classify whether each comment is replying to a post or another user. This was determined by examining the parent_id field of each comment:
    a. If parent_id starts with t3_, the comment is a reply to a post (classified as 'post').
    b. If parent_id starts with t1_, the comment is a reply to another comment (classified as 'user').
3. We then created two dictionaries: one for mapping comment IDs to comment authors and another for mapping submission IDs to submission authors.
4. Next, we created two functions to extract the user being replied to:
    a. **For replies to comments:** We used the comment_id_to_author dictionary to find the author of the comment being replied to.
    b. **For replies to posts:** We used the submission_id_to_author dictionary to find the author of the post being replied to.
5. Using, these two functions and two dictionaries, we map every comment to it's parent author, whether the comment is made to reply to another comment or it has been made to reply to a submission, and store the values in column **replied_to_user**.
6. Finally, the userInteractionDf DataFrame was created to capture the relationships between comments, users, and submissions. We extracted the relevant columns such as comment_id, comment_author, parent_id, replied_to, replied_to_user, and submission_author.

Data Modelling

After we create userInteractionDf, we remove rows with comment_author or replied_to_user is [deleted]. Because if we were to keep these rows, [deleted] will start to behave like an actual user and might distort the network graph. As there are 2691 rows where comment_author was found be [deleted], and even after removing these rows, there were 1596 rows with replied_to_user equal to [deleted]. All of these rows were removed from userInteractionDf.

We also filled any missing values in the submission_author column. In cases where a comment's submission_author was not already populated (i.e., it was NaN), we used a dictionary (commentSubmissionDict) to look up the submission author.

Explain the data organization and rationale

The data was organized into three separate dataframes to facilitate a more focused analysis for different tasks:

- **userInteractionDf**: This dataframe helps analyze interactions between users in the context of network analysis. By aggregating comment and submission data, it provides insight into how users interact with one another, including the frequency and type of comments, replies, and submissions. This organization is essential for understanding user behavior in terms of network theory.

Columns	Description
comment_id	A unique identifier for each comment, directly mapped from the id column in comments.json.
comment_author	The author of the comment, directly mapped from the author column in comments.json.
parent_id	The unique ID of the comment or post being replied to, with prefixes t1_ for posts or t3_ for comments. Directly mapped from the parent_id in comments.json.
replied_to	Indicates whether the comment was made in response to a post or another comment (values include 'post' or 'user').
replied_to_user	Indicates whether the comment was made in response to a post or another comment (values include 'post' or 'user').
submission_author	The author of the submission, derived by merging information from submissions.json, identifying the user who created the submission the comment belongs to.

- **commentActivityData** and **submissionActivityData**: These two dataframes were designed to allow for an in-depth analysis of user activity. They provide data points related to the volume of comments and submissions, the time of their occurrence, and the authorship. This setup enables us to track user behavior over time and identify patterns in how users post and comment.

  1. **commentActivityData**

Columns	Description
id	The unique identifier for each comment
author	The author of the comment.
created_ts	The timestamp when the comment was created.
body	The content of the comment.
week_number	The week number when the comment was made (used to analyze the comments over time).

  2. **submissionActivityData**

Columns	Description
id	The unique identifier for each submission.
author	The author of the submission
created_ts	The timestamp when the submission was made.
title	The title of the submission.
body	The content of the submission.
num_comments	The number of comments made on the submission.
week_number	The week number when the submission was made (used to analyze the comments over time).

This structure makes it easier to analyze user activity separately for each task, allowing for better focus and interpretation of data relevant to specific questions (e.g., network interactions or submission/comment behavior).

Summarize the data:

- Number of Unique Users: 12918
- Number of Users who both posted & commented: 1292
- Comments to Posts Ratio: 22863/18971 ~ 1.2
- Comments

Property	Value
Total Number of Comments	22863
Number of Deleted Comments	2691
Time Span	2012-02-01 to 2022-12-31
Days/Weeks	3986/570
Number of users who Commented (%)	7225 (55.93%)
No of Users who never Commented (%)	5693 (44.07%)
No of Users who only Commented (%)	5933 (45.92%)
Most Active User	Zurevu (with 2388 Comments)

- Submissions

Property	Value
Total Number of Submissions	18971
Number of Deleted Submissions	2062
Time Span	2012-02-01 to 2022-12-31
Days/Weeks	3986/570
No of Posts with Reply (%)	6251 (32.95%)
No of Posts with no Reply (%)	12720 (67.05%)
No of Users who Posted (%)	6985 (54.07%)
No of Users who never Posted (%)	5933 (45.93%)
No of Users who Posted more than 1 Submission (%)	1553 (12.02%)
Most Active User	Zurevu (with 2064 Submissions)

Task - 2: Graph Creation & Visualization

Describe the way you use the data for building graphs – what are origin nodes and what are destination nodes. Justify the approach.

In this analysis, we focus on the interactions between users based on their submissions, comments to submissions and responses to comments. The goal is to build a graph where the nodes represent users, and the directed edges represent interactions (one user replying to another).

- **Data Preparation for Building Graphs**:
    - **Origin Nodes**: These are the users who initiate interactions by replying to other users' comments. We get the origin nodes from the comment_author column of the userInteractionDf DataFrame.
    - **Destination Nodes**: These are the users who are being replied to, or the targets of the interaction. We get the destination nodes from the replied_to_user column of the userInteractionDf DataFrame.

  We start by filtering the DataFrame to include only non-null interactions and then remove any self-loops, i.e., instances where a user replies to their own submission, as this does not represent a meaningful interaction.

- **Graph Construction**:
    - A **directed graph** (DiGraph) is created where each edge represents a reply made by one user to another. In this setup:
        - The **origin node (source)** is the comment_author who is replying to someone.
        - The **destination node (target)** is the replied_to_user who is the recipient of the reply.

  This structure of the graph reflects the flow of communication between users, making it useful for analyzing user interactions such as who replies to whom and the overall communication network within the community.

- **Node and Edge Representation**:
    - **Nodes**: Each user is represented as a node in the graph.
    - **Edges**: The directed edges are created from the user who replied (comment_author) to the user who was replied to (replied_to_user). This interaction is treated as an edge directed from the replying user to the replied-to user.

  The reason for this directed approach is that it captures the direction of communication: replies indicate a response or follow-up action from one user to another. By maintaining this directionality, we can perform analyses such as identifying central users, communities of active reply networks, and the overall structure of the user interactions.

Justification of the Approach:

- **Directed Graph Representation**: The directed nature of the graph is crucial because it allows us to preserve the flow of interactions. For example, if user A replies to user B, we want to capture that flow of information or communication from A to B. A simple undirected graph wouldn't allow us to distinguish between who initiated the interaction and who received it.
- **Use of comment_author and replied_to_user**: These two columns in the data provide the necessary information to establish the relationships between users, enabling the creation of the directed edges.
- **Self-loops Removal**: By removing self-loops, we focus on genuine interactions between distinct users, ensuring that our graph represents meaningful interactions rather than isolated self-replies.

Visualisation of the graphs created; Graph for the entire data and zooming in on a part. How do you interpret this data? Can you make any observations about the data?

Complete User Interaction Network (Including Isolated Nodes)

This visualization shows the complete Reddit interaction network with the inclusion of isolated nodes (users with zero interactions). This allows for a better understanding of the overall network, including users who do not participate in discussions.

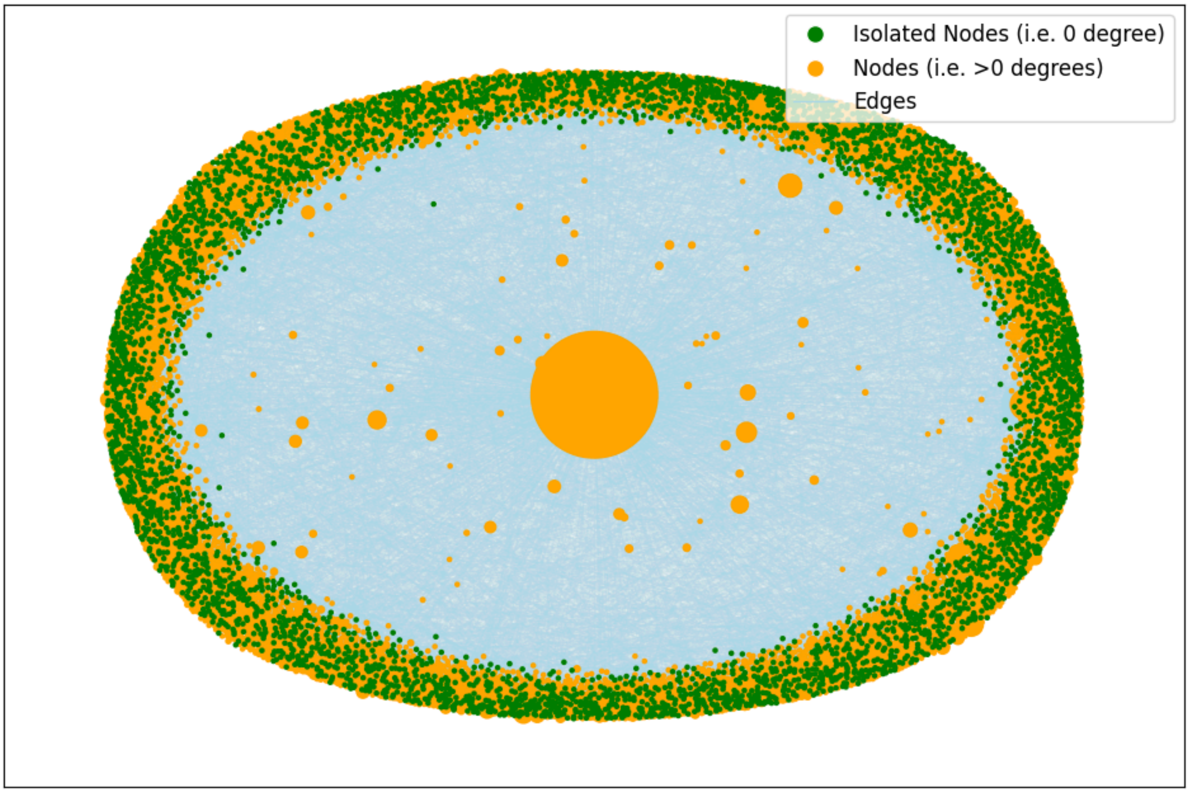
Graph Statistics:

- Number of Total Nodes - 12918
- Number of Isolated Nodes - 4577
- Number of Edges - 14274

Graph Interpretation

- Green nodes represent isolated users (those with zero interactions), while orange nodes represent active users who have at least one interaction. For better clarification, green nodes are those users who posted a submission but didn't receive any comments on their post, and they choose not to interact on any other post as well.
- Light blue edges connect active users in the network. These edges shows a interaction between two users, occured when a user makes a comment on a submission or a user replies to another user's comment.

InvestmentClub Reddit - Complete Interaction Network



Observations:

- **Isolated Nodes**: A considerable number of nodes are **isolated** (green). These nodes have **zero interactions** and do not participate in any discussions within the subreddit. They are placed on the outer edge of the network, confirming their lack of involvement.

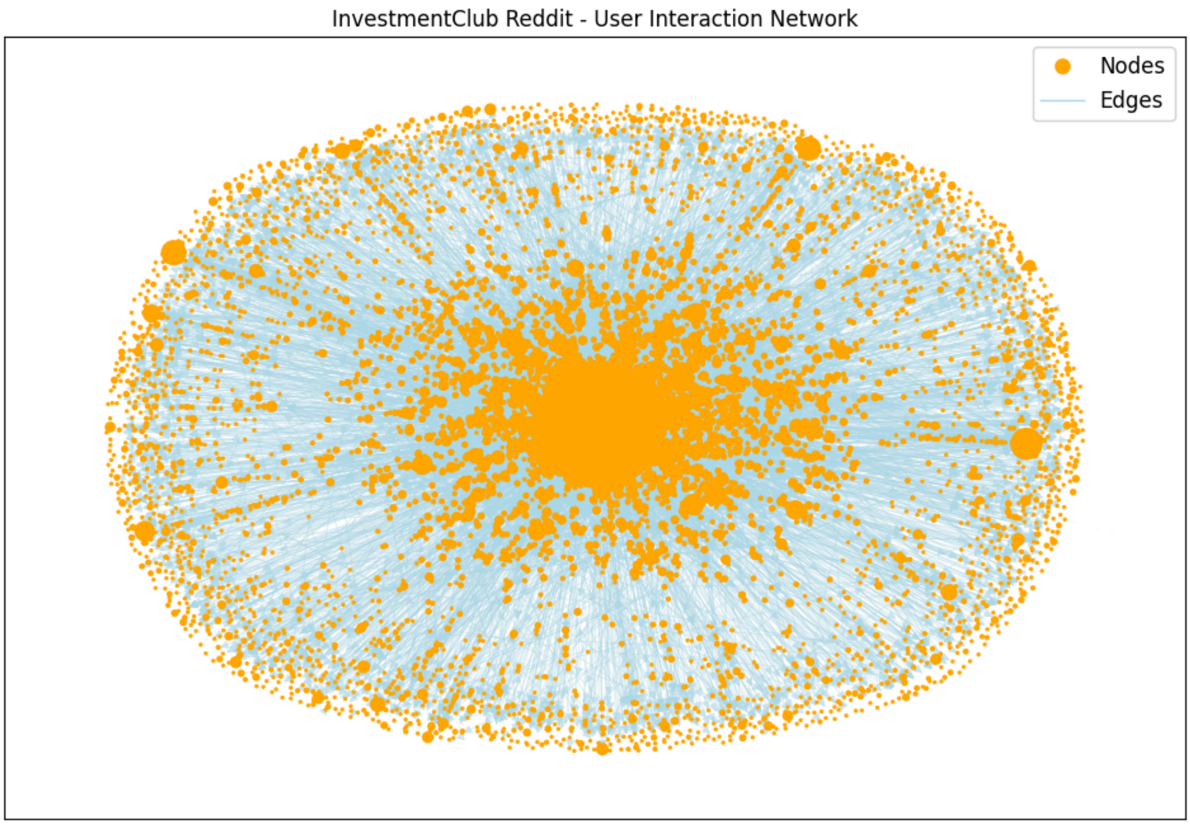
- **Majority of Active Nodes**: The **orange nodes** represent the majority of the active users in the network. These users have at least one interaction and are at the core of the network structure.
- **Low Degree Users on the Outer Edge**: Many users with fewer interactions are located on the periphery of the graph, implying that these users are relatively less engaged in conversations.

Complete User Interaction Network (without Isolated Nodes)

This visualization shows the complete user interaction network without the inclusion of isolated nodes (users with zero interactions). This allows for a better understanding of the overall network, including users who do not participate in discussions.

Graph Statistics:

- Number of Total Nodes - 8341
- Number of Edges - 14274

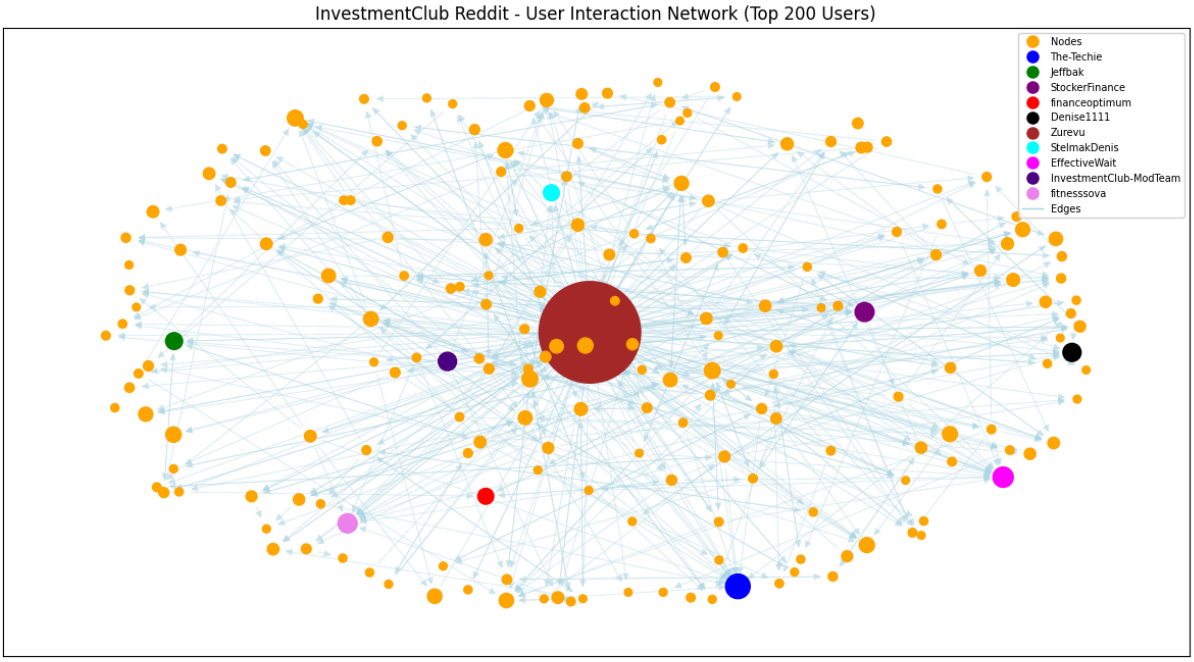


Observations:

- **High Interaction Concentration**: The central area of the graph is **denser** in terms of node sizes, indicating a cluster of users with **higher degrees** (more interactions).
- **Centralization of Discussions**: A few nodes (often larger) in the middle have **high degrees**. These central users are interacting with a lot of other users and are more involved in discussions. These users might include moderators or highly influential members.
- **Overall Sparse Connectivity**: The overall structure shows a **sparse connectivity** with thin edges connecting the nodes, especially on the outer parts. The edges are more concentrated near the center, suggesting that most discussions occur between a smaller group of users.
- **Subgroup Formation**: The graph hints at possible **subgroups or communities** based on the concentrated areas of interactions. This suggests that there might be certain topic-based clusters or sub-communities formed by a core group of users engaging with each other.

Top 200 User Interaction Network

The visualization above illustrates the **top 200 users' interaction network** within the **InvestmentClub Reddit community**. This graph represents the relationships and communication between the most active users based on their degree centrality, which indicates the number of interactions they have within the community.



Observations:

- **Highly Active Users**: A small set of users dominate the network, characterized by large, brightly colored nodes. These users interact with multiple others, likely contributing significantly to the discussions and overall activity in the subreddit. "Zurevu" user seems to highly important user in InvestmentClub subreddit community.
- **Peripheral Users**: The majority of users are located on the outer edges of the network, indicating limited interactions. These users are not as central to the community's discussions but still participate.
- **Clusters of Activity**: The graph shows some clustering, where specific users are more directly connected to each other, forming tight-knit subgroups within the community. This suggests that certain topics or discussions may attract specific groups of users, creating sub-communities based on shared interests or engagement patterns.

Super Users:

The list of top 10 super users, identified based on **degree centrality**, highlights the most influential individuals within the community. These users have a significantly higher number of connections (edges) compared to others, making them central to the network's structure. And as seen from the above graph and now confirmed with degree centrality of top users, Zurevu is definitely the most important user in the community.

User	Degree Centrality
Zurevu	0.3080335731414868
The-Techie	0.017985611510791366
EffectiveWait	0.01223021582733813
fitnesssova	0.01091127098321343
StockerFinance	0.01079136690647482

User	Degree Centrality
InvestmentClub-ModTeam	0.009952038369304557
Denise1111	0.009832134292565948
Jeffbak	0.008633093525179856
StelmakDenis	0.007673860911270983
financeoptimum	0.007553956834532374

Task - 3: Network Analysis and Other Important Properties

Study the role of super users in the community. How central are they to the cohesiveness and functioning of the community?

Metric: Size of the Largest Strongly Connected Component (LCC) after Node Removal

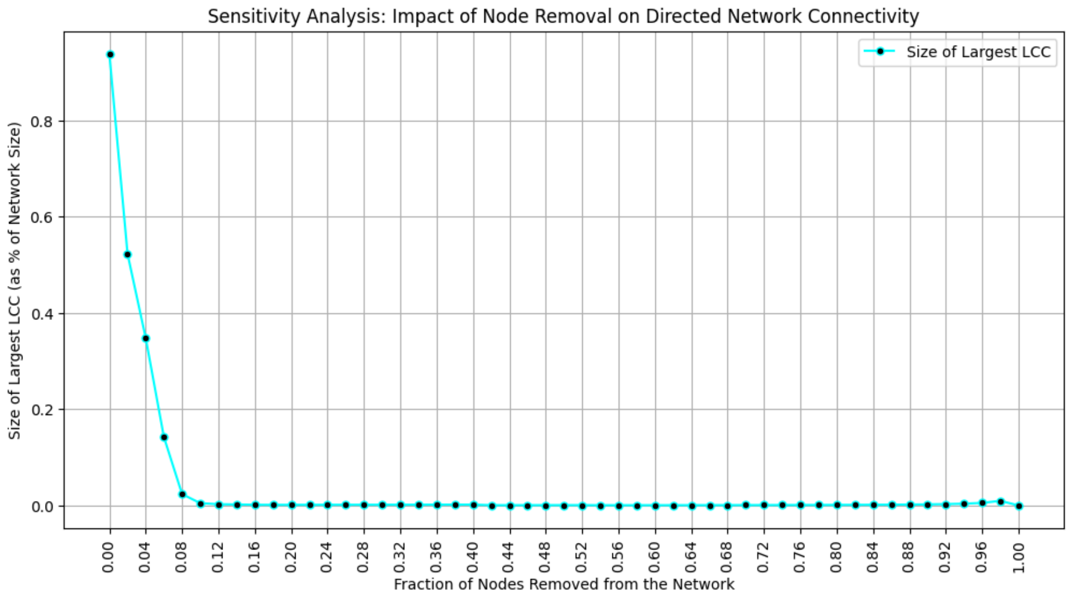
Description:
The "Size of the Largest Strongly Connected Component (LCC)" measures the connectivity of the network. It calculates the largest subgraph in which every node is reachable from every other node. In the context of this analysis, we focus on how the removal of super users (highly connected users) impacts the cohesiveness of the network, as seen in the plot provided.

- Interpretation:
    - If a super user is removed, and the LCC size significantly decreases, it suggests that the super user is crucial for maintaining the connectivity and cohesiveness of the network.
    - If the LCC size remains relatively constant despite node removal, it implies that the network is more resilient and doesn't rely heavily on a few super users.

Pseudo - Code:

```
Sort nodes by degree in descending order.

Initialize an empty list to store the sizes of the largest connected component (LCC).

Calculate the LCC size for the full graph and append it to the list.

Define the fraction of nodes to remove (e.g., 2%).

For each iteration, repeat the following steps:
    a. Remove the top nodes (based on degree).
    b. Calculate the LCC size of the remaining graph.
    c. Append the LCC size to the list.

Return the list of LCC sizes after node removal.
```

Plotted Graph:



Observations:

1. **Initial Network Connectivity:**
    - When **0% of the nodes are removed**, the size of the largest strongly connected component (LCC) is **close to 100%** of the total network size. This indicates that the network is **highly connected** and cohesive initially, with all nodes being part of a large, well-connected component.
2. **Sharp Decline After Node Removal:**
    - As soon as the **first 5% of nodes** are removed from the network, we observe a **sharp drop in the size of the largest connected component**. The size of the LCC decreases significantly, showing that the **removal of a small proportion of nodes** (particularly high-degree nodes or central users) leads to a **disruption in connectivity**. This indicates that the **super users** or highly connected nodes play a crucial role in maintaining the structure of the network.
3. **Flattening of the LCC Curve:**
    - After approximately **10% of the nodes are removed**, the size of the largest connected component stabilizes and does not decrease significantly with further node removals. This suggests that once a critical number of key nodes are removed, the remaining network becomes **fragmented**, and the LCC reaches a point where it no longer shrinks rapidly, implying that most of the cohesive structure has already been destroyed

The graph shows that the **super users** (the most influential nodes in the network) hold a critical position in maintaining the network's **cohesiveness**. **Removing a small percentage** of these central nodes (the top 5-10%) leads to a **dramatic decrease in network connectivity**, indicating their **critical role** in keeping the network intact. The sharp decline in the size of the LCC suggests that **super users** are responsible for much of the **connectivity** in the network. Once these users are removed, the network's **larger connected component** disintegrates, and the network becomes fragmented.

This indicates that **highly active users** or **moderators** (those with the most interactions) are **essential for network cohesion**. Their removal can lead to **disconnection** within the community, disrupting interactions and causing the network to become much more **sparse** and less connected.

How users on support communities, as a group behave over time?

To answer the question **"How users on support communities, as a group behave over time?"**, we can use **Z-Score analysis** as a metric to measure user activity in relation to their average activity and identify outliers or unusual behavior. The Z-Score can give us insights into whether users are behaving typically or exhibiting extreme activity, which could help in understanding how support communities evolve.

Metric: Z-Score (Standardized Score)

The **Z-Score** is a statistical measure that describes how far a data point is from the mean, in terms of standard deviations. It's particularly useful to understand whether users are more active or less active compared to the average behavior of the community.

Description:

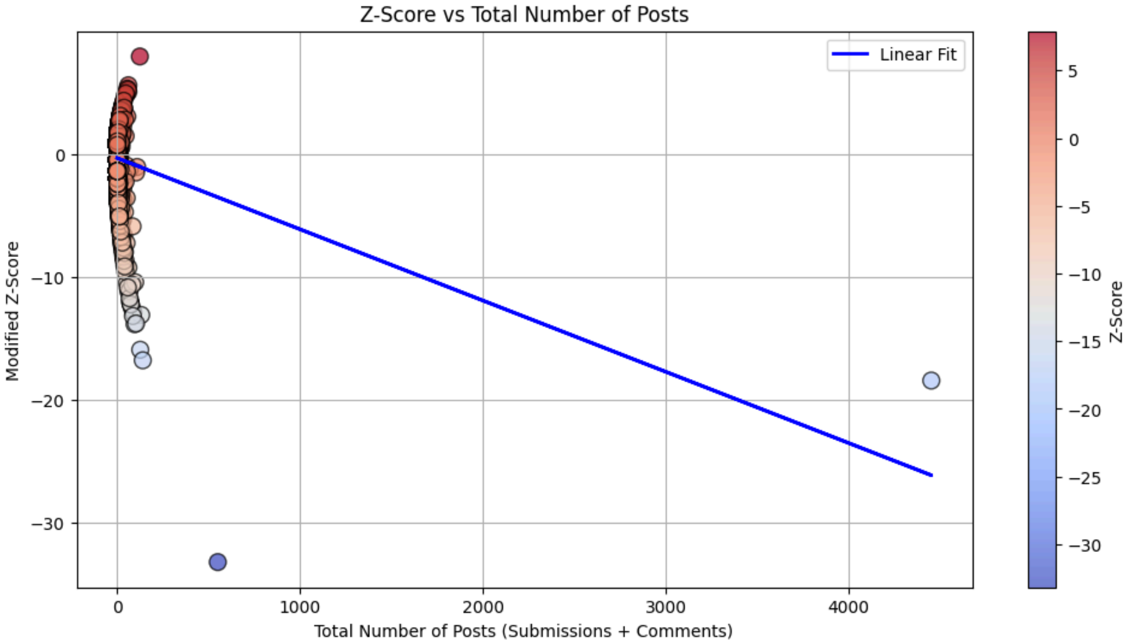
Z-Score helps to standardize the number of interactions (posts, comments) each user has in the community.

- Interpretation:
    - A positive Z-Score indicates above-average activity, while a negative Z-Score indicates below-average activity.
    - In this context, the Z-Score can be used to track a user's engagement over time to observe if they are asking for help or helping other users in the group.

Pseudo - Code:

```
1. Remove unwanted users (i.e., '[deleted]' and NaN):
       filteredSubmissionActivityData = submissionActivityData[~submissionActivityData['author'].isin(['[deleted]', np.nan])]
       filteredCommentActivityData = commentActivityData[~commentActivityData['author'].isin(['[deleted]', np.nan])]

2. Count submissions and comments per user:
       submissionCounts = filteredSubmissionActivityData['author'].value_counts()
       commentCounts = filteredCommentActivityData['author'].value_counts()

3. Merge the submission and comment counts:
       userActivity = pd.DataFrame({
          'submissions': submissionCounts,
          'comments': commentCounts
       }).fillna(0)

4. Calculate total posts:
   userActivity['totalPosts'] = userActivity['submissions'] + userActivity['comments']

5. Calculate Z-score for each user:
   Use the formula: Z = (answers - 2 * questions) / sqrt(2 * (questions + answers))
     - This measures the activity imbalance between submissions (questions) and comments (answers).
     - Higher positive values of Z-score represent users with more comments relative to submissions.
     - Lower or negative values of Z-score represent users with more submissions than comments.

6. Store or return the final result:
       userActivity['zScore'] = (
                  userActivity['comments'] - 2 * userActivity['submissions']) / np.sqrt(2 * (userActivity['submissions'] + userActivity['comments'])
       )
```

Plotted Graph:



Observations:

1. **Help Seekers and Support Givers Distribution:**
    - **6504 users** have a **Z-Score < 0**, categorizing them as **Help Seekers**. These users tend to post more **submissions** (questions) and fewer **comments** (answers), indicating they are seeking help within the community.
    - **6255 users** have a **Z-Score > 0**, categorizing them as **Support Givers**. These users tend to post more **comments** (answers) and fewer **submissions** (questions), indicating they are actively providing support to others by answering questions and sharing knowledge.
2. **Users with Neutral Behavior:**
    - **159 users** have a **Z-Score = 0**, behaving as both **Help Seekers** and **Support Givers**. These users make a balanced amount of **submissions** (questions) and **comments** (answers), indicating they engage in both seeking help and providing assistance to others, thus playing a dual role in the community.

How exclusive super users in the community in their behaviour?

Metric: Rich-Club Coefficient

Description:

The **Rich-Club Coefficient** measures the tendency of high-degree nodes (super users) in a network to form connections with other high-degree nodes. It quantifies the concentration of connections among nodes with high degrees and indicates how exclusive or tightly-knit the "rich" users (high-degree users) are in their behavior. A higher rich-club coefficient suggests that super users tend to form clusters among themselves, making them more exclusive in terms of network connectivity.

The rich-club coefficient is calculated as:

$$\phi(k) = \frac{2E_k}{N_k(N_k - 1)}$$

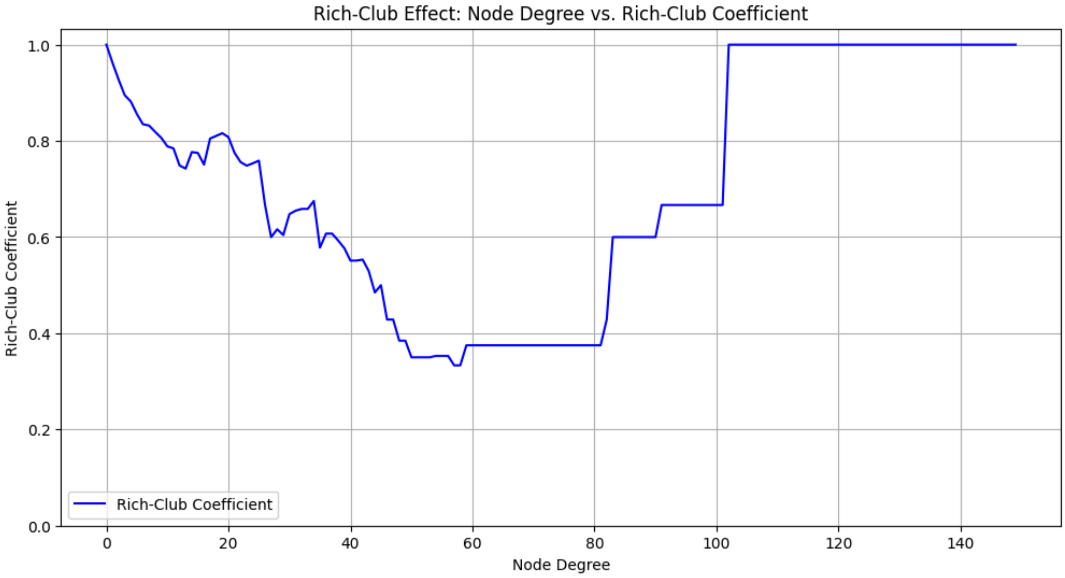
where:

- N_k is the number of nodes with degree greater than k
- E_k is the number of edges among those nodes.

Pseudo - Code:

```
# Calculate the rich-club coefficient
richClubValues = nx.rich_club_coefficient(graphUndirected, normalized = True, seed = 40)

# Extract the degree values and corresponding rich-club coefficients
degrees = sorted(richClubValues.keys())  # Degree values
richClubCoeffs = [richClubValues[k] for k in degrees]

# Use degrees and richClubCoeffs to plot the graph
```

Plotted Graph:

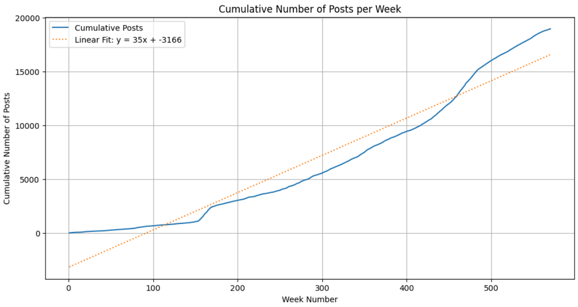


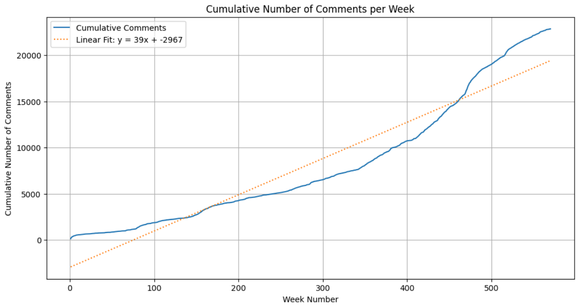
Observations:

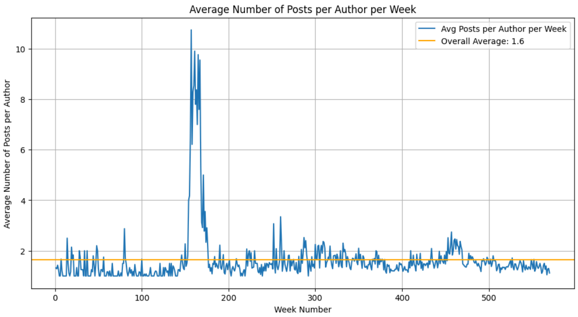
1. **Declining Rich-Club Coefficient:**
   - The graph shows that the **rich-club coefficient** tends to decrease as the **node degree** increases. This indicates that the **most connected nodes** (super users) have fewer exclusive connections among themselves, which implies that they **interact with lower-degree users** more frequently than with other super users.
2. **Sharp Increase at Higher Degrees:**
   - There is a noticeable **increase in the rich-club coefficient** for users with **higher degrees** (e.g., above degree 100). This suggests that, beyond a certain threshold, super users **tend to form more exclusive communities** with other high-degree users, making the behavior more **exclusive and clustered** within that group.
3. **Exclusive Behavior of Super Users:**
   - The **super users** (with high degrees) are observed to be more exclusive in their interactions. Initially, the network is highly interconnected, but as the degree increases, these users show a preference for connecting within their **own clique**, further isolating themselves from the rest of the network.

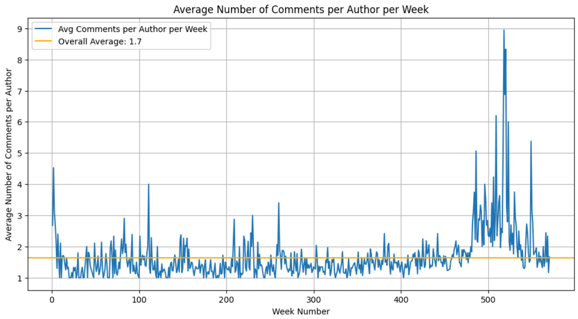
Task - 4: Additional Research Question

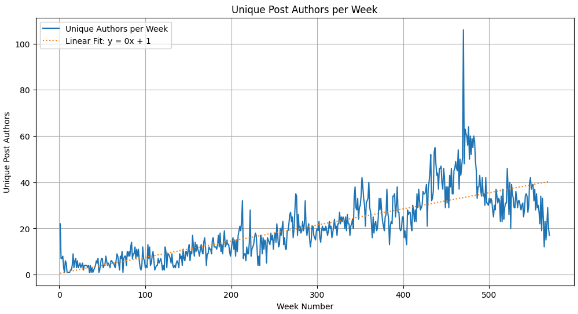
How users in the community behave over time ?

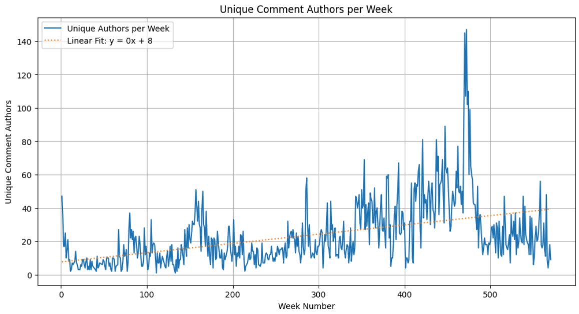

Cumulative Number of Posts per Week


Cumulative Number of Comments per Week


Average Number of Posts per Author per Week


Average Number of Comments per Author per Week


Unique Post Authors per Week


Unique Comment Authors per Week

Observations:

- Both posts and comments exhibit a **steady upward trend** over time, reflecting the growth in community engagement.
- As seen in both cumulative graphs, there is a **significant spike around weeks 460-470**, where both submissions and comments show a sharp increase in activity. This could indicate a period of heightened interest or a specific event that drove user engagement. This can also be explained by Unique Post/Comment Authers per week graphs, as there's a significant increase of unique users between 460-470 week.
- The growth in comments generally aligns with the increase in posts, suggesting an ongoing, proportional relationship between the two metrics.
- Both the average number of posts and comments per author remain low for most weeks, with the majority of authors contributing **fewer than 2 posts/comments per week**, signifying a few highly engaged users are driving the activity, while the majority of users remain less active.

What is the community talking about ?

Metric: Topic Modeling using Latent Dirichlet Allocation (LDA)

Description:
Latent Dirichlet Allocation (LDA) is a probabilistic model used for topic modeling, which helps in extracting **hidden topics** from a collection of text documents. It assumes that each document is a mixture of topics, and each word in the document is attributable to one of the document's topics. The key goal of LDA is to find the **distribution of topics** across documents and the **distribution of words** for each topic.

In this case, LDA is used on **high engagement submissions** (those with 5 or more comments) from the dataset to identify common themes (or topics) discussed by the community. This can help to understand **what the community is talking about** based on the contents of the submissions.

Pseudo - Code:

```
Step 1: Filter Submissions with more than 5 comments
Step 2: Combine 'title' and 'selftext' into one text column 'full_text'
Step 3: Text Vectorization using TF-IDF to convert text to numerical features

Step 4: Apply Latent Dirichlet Allocation (LDA) for Topic Modeling
n_topic = 5
lda = LatentDirichletAllocation(n_components = n_topic , random_state = 42)
lda.fit(X)

Step 5: Extract and print the top words for each topic
```

Observations:

- Topics Detected:

  ```
  Topic 0:  ['amp', 'com', 'hi', 'will', 'type', 'deflation', 'investmentclub', 'stock', 'removed', 'mcdonald']
  Topic 1:  ['money', 'just', 'buy', 'can', 'market', 'investment', 'investing', 'invest', 'stocks', 'stock']
  Topic 2:  ['stock', 'investing', 'apple', '2020', 'billion', 'buffett', 'warren', 'user', 'deleted', 'removed']
  Topic 3:  ['oil', 'amazon', 'buy', 'bitcoin', 'recession', 'crash', 'favorite', 'analysis', 'market', 'stock']
  Topic 4:  ['new', 'next', 'electric', 'stock', 'apple', 'buy', 'removed', 'market', 'peter', 'bitcoin']
  ```

- **Stock and Market Focus:** A significant number of topics revolve around stock investments, trading strategies, and market trends.
- **High-Profile Figures:** There is a strong presence of discussions about famous investors, such as Warren Buffett, and major companies like Apple, suggesting the community is focused on well-known stocks.
- **Emerging Trends:** Topics also reflect interest in emerging sectors, such as electric vehicles and cryptocurrencies like Bitcoin, with conversations around future market developments.
- **Economic Impact:** The conversation often ties into broader economic issues like recessions, stock crashes, and deflation, indicating that users are aware of external market influences

Focusing on Zurevu:

Zurevu is the user driving the majority of engagement in the **InvestmentClub** subreddit, as evidenced by his high number of comments (2388) and submissions (2064), along with his degree centrality of approximately **0.31**. In comparison, the second most influential user, **The-Techie**, has a degree centrality of only **0.018**.

Upon further investigation, the likely reason for Zurevu's high engagement is that he started the subreddit and is most likely its **admin**. This assumption is supported by his first submission, titled "**Welcome to /r/InvestmentClub!**". And that's also the first submission from submissions.json file sorted on created_utc in ascending order.

Conclusion:

In this analysis of the **InvestmentClub** subreddit, several key observations were made about the behavior and interactions of users within the community. By aggregating and processing the comment and submission data, we were able to track user activities and identify influential users, topics of discussion, and overall trends.

1. **User Interaction Dynamics:** The dataset revealed that a significant portion of users were either predominantly asking questions (Help Seekers) or answering them (Support Givers), with a smaller subset acting as both. A few highly engaged users played a central role in maintaining the network's cohesiveness.
2. **Super Users and Network Cohesion:** Through the use of **Degree Centrality** and **Largest Strongly Connected Component (LCC) analysis**, we found that a small group of super users were crucial to the community's interaction structure. Their removal led to a rapid fragmentation of the network, highlighting their importance in maintaining connectivity.
3. **Community Engagement Over Time:** Both posts and comments followed a generally upward trend, with notable spikes around specific periods, suggesting events or topics that significantly impacted user engagement. Despite the overall increase in activity, the majority of users contributed only a small number of posts or comments, with a few active users driving the majority of the interactions.
4. **Topic Analysis: Latent Dirichlet Allocation (LDA)** was used to identify the topics most discussed in high-engagement submissions. Key topics included stock investments, major figures like Warren Buffett, and emerging market trends such as electric vehicles and cryptocurrencies.

Overall, this analysis demonstrates the dynamics of a large online community and the factors driving engagement. The most influential users shape the discussions, while trending topics guide the direction of conversations within the community.