Graph-Based Analysis of Youth Smoking and Drug Experimentation Patterns

Data Set: https://www.kaggle.com/datasets/waqi786/youth-smoking-and-drug-dataset

This project explores graph algorithms and their application to analyze behavioral patterns in youth smoking and drug experimentation. The dataset includes various demographic and behavioral attributes of individuals, such as socioeconomic status, age group, and peer influence, providing a rich foundation for examining relationships within a network. The project offers insights into how network connectivity shapes individual behaviors The project investigates how shared attributes between individuals (peer influence, age group, and socioeconomic status) influence smoking prevalence and drug experimentation, evaluates degree distributions, analyzes distance-2 neighbors, and explores correlations between node connectivity and behavioral outcomes.

Objective

The primary objective of the project is to leverage graph-based methods to:

1. Identify degree distributions and their relationship to individual behaviors.
2. Analyze the impact of second-order connections (distance-2 neighbors) on behavioral patterns.
3. Determine whether specific connectivity patterns align with real-world phenomena, such as increased exposure to risky behaviors in highly connected individuals.

Key Features

- Construction of an undirected graph where nodes represent individuals and edges represent relationships defined by shared attributes (e.g., similar socioeconomic status, age group, and peer influence).
- Calculation of degree distributions to assess how node connectivity varies across the network.
- Identification of nodes with the highest number of distance-2 neighbors to understand indirect influence in the network.
- Behavioral analysis correlating node degrees to smoking prevalence and drug experimentation, reveals how connectivity might influence behavior.

The codebase is divided into three distinct files—main.rs, graph.rs, and analysis.rs—to ensure modularity and ease of understanding. Each file focuses on specific functionalities, including the main program flow, graph construction, and data analysis. Here's an in-depth explanation of how the code operates.

main.rs

The execution begins in main.rs, where the dataset is first loaded using the load_dataset function. This function parses a CSV file containing data about individuals, each represented as a Person struct. The struct includes attributes such as peer_influence (a numerical measure of social influence), age_group (a categorical variable for age), socioeconomic_status, smoking_prevalence (percentage of smokers), and drug_experimentation (percentage experimenting with drugs).

The create_graph function is then invoked to build a graph where nodes correspond to individuals, and edges represent meaningful social connections. This graph is undirected, meaning connections are mutual. The function returns both the graph and a mapping between dataset indices and node identifiers. The program uses the mapping to link analytical results back to individuals in the dataset.

The compute_degree_distribution function is called next. It calculates the degree (number of connections) for each node in the graph and tallies the frequency of each degree across the graph. This provides insights into the connectivity of the network, indicating whether most individuals have a few or many connections. The results are stored as a HashMap, with degrees as keys and their respective counts as values.

Another critical step in main.rs is the call to compute_distance_2_neighbors. This function computes the number of unique nodes reachable from each node within two hops. The computation involves iterating over each node's immediate neighbors and their respective neighbors, ensuring that the original node itself is excluded. The results indicate the extent of an individual's indirect influence within the network.

The analyze_behavior_by_degree function is also invoked. It calculates the average smoking prevalence and drug experimentation for nodes grouped by their degree. This is achieved by iterating through all nodes, summing the smoking and drug values for nodes with the same degree, and then dividing by the count of nodes in each group. The output allows the program to analyze behavioral trends concerning connectivity levels.

Finally, the program prints a summary report. The degree distribution is sorted to display the top 10 degrees with the highest frequencies, providing an overview of the network's structure. The top 10 nodes with the most distance-2 neighbors are identified, showcasing nodes with extensive indirect influence. The behavioral analysis results are sorted by average smoking prevalence, emphasizing the relationship between connectivity and behavior.

graph.rs

The create_graph function is implemented in graph.rs. It uses the petgraph library to construct an undirected graph. Nodes are added for each individual, and edges are established based on criteria defined in the should_connect function. This function determines whether two nodes should be connected by checking the similarity of their attributes:

1. The absolute difference in peer_influence must be ≤ 2.
2. The age_group of both individuals must match.
3. The socioeconomic_status must be the same.

This ensures that edges represent meaningful social connections rather than random associations. The mapping between dataset indices and node identifiers returned alongside the graph, is essential for linking individuals back to analytical results.

analysis.rs

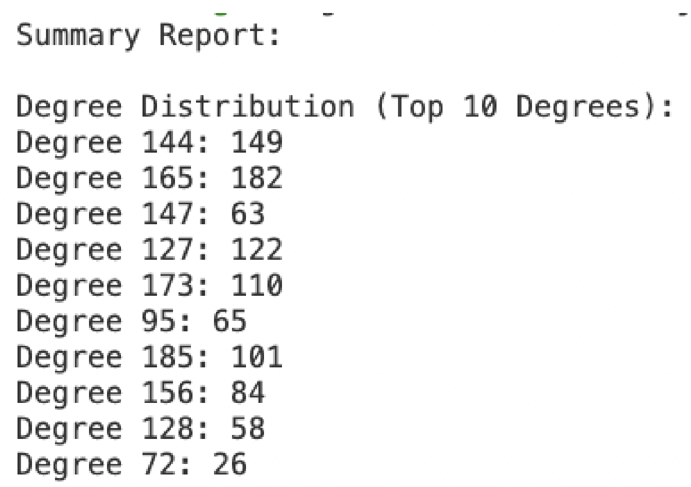
This file houses functions for analyzing the graph structure and individual behaviors.

1. compute_degree_distribution:
   This function iterates over all nodes in the graph and counts the number of edges (connections) for each node. These counts are grouped into a HashMap, where keys are degrees, and values are the number of nodes with that degree. The distribution reveals whether the network exhibits characteristics of a power-law distribution, commonly seen in social networks.
2. compute_distance_2_neighbors:
   This function calculates the number of unique nodes that can be reached within two hops from each node. It iterates over all neighbors of a node and then over their neighbors, using a HashSet to avoid duplicates. The results highlight nodes with significant indirect influence, which can be critical in understanding how behaviors like smoking and drug experimentation might spread.
3. analyze_behavior_by_degree:
   This function examines how behaviors vary with connectivity. It groups nodes by their degree and calculates the average smoking prevalence and drug experimentation for each group. The calculation involves summing the values for all nodes in a group and dividing by the group size. The results help identify trends, such as whether individuals with higher connectivity exhibit different behavioral patterns compared to those with lower connectivity.

Example Analysis and Code Details

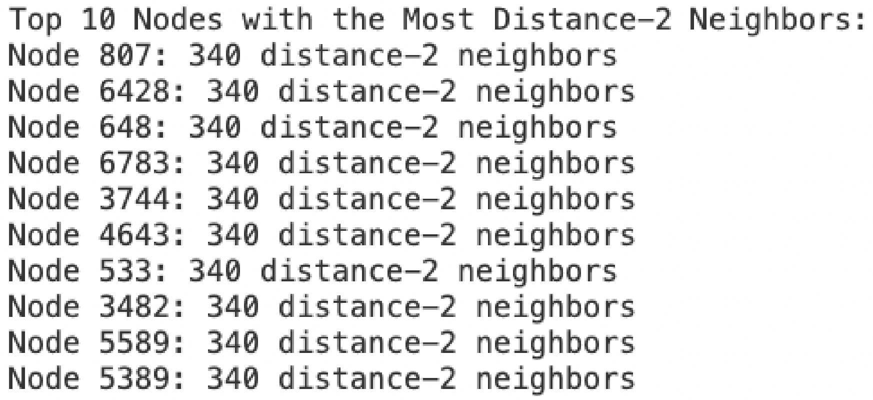
- The load_dataset function in main.rs ensures the input data is clean and ready for analysis, parsing each row into a structured Person object.
- In create_graph (graph.rs), edges are added selectively to avoid creating a fully connected graph, maintaining realistic social connections.
- The compute_distance_2_neighbors function provides a nuanced understanding of network reach by capturing secondary connections, which can influence the spread of behaviors indirectly.
- The analyze_behavior_by_degree function’s output, sorted by smoking prevalence, directly links connectivity to health-related behaviors, offering actionable insights for interventions.

Output

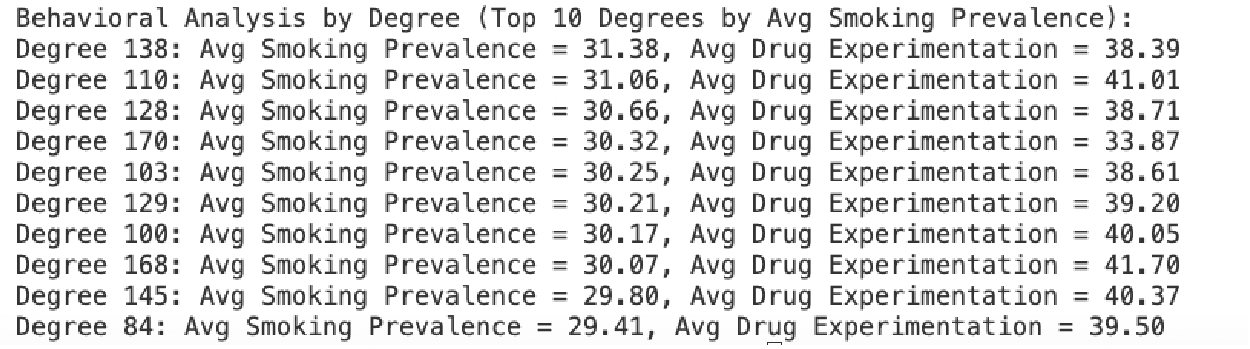


```
Summary Report:

Degree Distribution (Top 10 Degrees):
Degree 144: 149
Degree 165: 182
Degree 147: 63
Degree 127: 122
Degree 173: 110
Degree 95: 65
Degree 185: 101
Degree 156: 84
Degree 128: 58
Degree 72: 26
```



```
Top 10 Nodes with the Most Distance-2 Neighbors:
Node 807: 340 distance-2 neighbors
Node 6428: 340 distance-2 neighbors
Node 648: 340 distance-2 neighbors
Node 6783: 340 distance-2 neighbors
Node 3744: 340 distance-2 neighbors
Node 4643: 340 distance-2 neighbors
Node 533: 340 distance-2 neighbors
Node 3482: 340 distance-2 neighbors
Node 5589: 340 distance-2 neighbors
Node 5389: 340 distance-2 neighbors
```



```
Behavioral Analysis by Degree (Top 10 Degrees by Avg Smoking Prevalence):
Degree 138: Avg Smoking Prevalence = 31.38, Avg Drug Experimentation = 38.39
Degree 110: Avg Smoking Prevalence = 31.06, Avg Drug Experimentation = 41.01
Degree 128: Avg Smoking Prevalence = 30.66, Avg Drug Experimentation = 38.71
Degree 170: Avg Smoking Prevalence = 30.32, Avg Drug Experimentation = 33.87
Degree 103: Avg Smoking Prevalence = 30.25, Avg Drug Experimentation = 38.61
Degree 129: Avg Smoking Prevalence = 30.21, Avg Drug Experimentation = 39.20
Degree 100: Avg Smoking Prevalence = 30.17, Avg Drug Experimentation = 40.05
Degree 168: Avg Smoking Prevalence = 30.07, Avg Drug Experimentation = 41.70
Degree 145: Avg Smoking Prevalence = 29.80, Avg Drug Experimentation = 40.37
Degree 84: Avg Smoking Prevalence = 29.41, Avg Drug Experimentation = 39.50
```

To run the program, clone the repo and do cargo run --release in the project folder.

The first section, Degree Distribution, shows the top 10 degrees in the graph and their respective counts. The degree of a node represents the number of direct connections it has in the social network. For example, 182 nodes have a degree of 165, meaning these individuals are directly connected to 165 others. This suggests that certain individuals act as central figures in the network, potentially amplifying their influence on behaviors like smoking and drug experimentation. In contrast, nodes with lower degrees (e.g., Degree 72 with 26 occurrences) represent individuals with fewer direct social ties. Understanding the degree distribution helps characterize the network's structure, such as whether it follows patterns like a power-law distribution. The graph exhibits some characteristics of a power-law distribution, such as the presence of high-degree nodes (hubs) and a larger number of low-degree nodes, which align with typical patterns seen in scale-free networks. However, the degree counts for higher degrees are more uniform than expected in a strict power-law

The Top 10 Nodes with the Most Distance-2 Neighbors section highlights the nodes with the largest number of unique neighbors within two hops. For instance, Node 807 and others listed in this section have 340 distance-2 neighbors, indicating that their indirect reach is substantial. This suggests these nodes are embedded in dense regions of the network, making them key players in disseminating behaviors or information. Practically, this insight identifies individuals who could amplify public health campaigns or interventions. If a person with high distance-2 connectivity adopts a positive behavior, such as quitting smoking, their indirect influence could inspire a large subset of the population to follow suit.

The Behavioral Analysis by Degree provides a deeper look at how connectivity correlates with health-related behaviors. It lists the top 10 degrees sorted by average smoking prevalence. Nodes with a degree of 138 exhibit the highest smoking prevalence (31.38%) and moderate drug experimentation (38.39%), while nodes with a degree of 110 show slightly lower smoking prevalence (31.06%) but higher drug experimentation (41.01%). Interestingly, nodes with lower degrees, like 84, show reduced smoking prevalence (29.41%) but similar levels of drug experimentation. This suggests that individuals with moderate to high connectivity might be more exposed to smoking due to social influence, while drug experimentation may not correlate as strongly with direct connections. These patterns provide actionable insights for public health professionals. For instance, individuals with high degrees could be prioritized for smoking cessation programs, leveraging their influence to propagate healthier behaviors across the network.

Overall, the program's output describes the social network's connectivity, identifies key influencers, and examines how behaviors vary with connectivity. These insights can guide real-world interventions, such as targeting central figures for public health campaigns or focusing efforts on dense network clusters to maximize the spread of positive behaviors.