import pandas as pd

import networkx as nx

import matplotlib.pyplot as plt

def analyze\_and\_plot\_network(csv\_path):

# Load data

data = pd.read\_csv(csv\_path)

# Create a graph from the edge list

G = nx.from\_pandas\_edgelist(data, 'source', 'target')

# Compute the degree of each node

degrees = dict(G.degree())

# Save the degree to a text file

summary\_path = 'network\_analysis\_summary.txt'

with open(summary\_path, 'w') as file:

file.write("Node degrees:\n")

file.write("\n".join(f"{node}: {deg}" for node, deg in degrees.items()))

# Plot the graph

plt.figure(figsize=(8, 6)) # Set the size of the plot

pos = nx.spring\_layout(G) # Positions for all nodes using a spring layout

nx.draw(G, pos, with\_labels=True, node\_color='skyblue', node\_size=[v \* 100 for v in degrees.values()], font\_size=8, edge\_color='gray')

labels = nx.get\_edge\_attributes(G, 'weight')

nx.draw\_networkx\_edge\_labels(G, pos, edge\_labels=labels)

# Save plot to a file

plt.savefig('network\_graph.png')

plt.show()

# Return the path of the saved summary

return summary\_path, 'network\_graph.png'

# Example usage

try:

summary\_file\_path, graph\_image\_path = analyze\_and\_plot\_network(Friendships\_Data.csv')

print(f"Summary file saved at: {summary\_file\_path}")

print(f"Graph image saved at: {graph\_image\_path}")

except Exception as e:

print(f"An error occurred: {e}")

**What This Script Does:**

1. **Creates Graph**: It constructs a graph from your CSV file containing edge pairs.
2. **Computes Degrees**: It calculates the degree for each node.
3. **Plots Graph**: It visually plots the graph using a spring layout, which tries to position the nodes in a way that minimizes edge overlap. Node sizes are proportional to their degrees, enhancing the visualization to easily identify hubs or highly connected nodes.
4. **Saves Plot**: The plot is saved to an image file (network\_graph.png).
5. **Displays Plot**: Besides saving, it also displays the plot in a window.
6. **Reads CSV**: It reads your CSV file into a pandas DataFrame. The CSV should have two columns, source and target, which represent the nodes connected by each edge.
7. **Creates Graph**: It uses nx.from\_pandas\_edgelist to create a graph. This function directly interprets the rows in the DataFrame as edges between nodes listed under source and target.
8. **Computes Degrees**: It calculates the degree of each node (i.e., the number of connections each node has).
9. **Saves Summary**: It writes a summary of the node degrees to a text file.