BDA Assignment-1

Overview

Dataset: Wikipedia voting network ("wiki-Vote.txt") — directed graph of who votes for whom in Wikipedia admin elections. It comprises **7,115 nodes** and **103,689 edges** (Stanford Network Analysis Project, Wolfram Data Repository).

You're given **ground-truth metrics**:

- Nodes: 7,115
- Edges: 103,689
- Nodes in largest WCC: 7,066 (0.993)
- Edges in largest WCC: 103,663 (1.000)
- Nodes in largest SCC: 1,300 (0.183)
- Edges in largest SCC: 39,456 (0.381)
- Average clustering coefficient: 0.1409
- Number of triangles: 608,389
- Fraction of closed triangles: 0.04564
- Diameter: 7
- 90-percentile effective diameter: 3.8

Students will compute these characteristics using Spark, then compare with the ground truth.

Assignment Structure

1. Data Loading & Preprocessing (Spark)

- Load the edge list (text file with "from to") into a Spark DataFrame or RDD.
- Cast node IDs to appropriate types and handle duplicates if any.

2. Basic Graph Statistics

- **Count of nodes and edges**:
  - **Nodes**: unique IDs from both columns.
  - **Edges**: count of rows.
- Validate against ground truth (7115 nodes, 103689 edges).

3. Weakly Connected Components (WCC)

- Use graph processing libraries available in Spark (e.g. GraphFrames or GraphX).
- Compute size (node & edge counts) of the largest WCC; compare with ground-truth (7066 nodes, 103663 edges).

4. Strongly Connected Components (SCC)

- Similarly, compute largest SCC component sizes; compare with ground-truth (1300 nodes, 39,456 edges).

5. Clustering Metrics & Triangles

- Compute **average clustering coefficient**.
- Count **total number of triangles** present.
- Compute **fraction of closed triangles**: (triangles) / possible triplets; compare to 0.04564.

6. Distance-based Metrics

- Compute **diameter** (longest shortest path) and **effective diameter** (90th percentile of shortest-path distances).
- Compare with ground-truth (diameter = 7, effective = 3.8).

7. Comparison Report

- Tabulate your computed values vs ground truth.
- For each metric, discuss:
    - Accuracy of your computation.
    - Potential sources of discrepancies (e.g., parallel rounding, sampling, numerical approximations, data loading issues).

Suggested Assignment Format

You might ask students to:

1. **Implement each metric** with Spark in a well-documented notebook or script.
2. **Visualize** results where feasible (e.g., distribution of component sizes, path-length histogram for effective diameter).
3. **Write a summary report** that includes:
    - A results table comparing computed vs ground truth.
    - Short explanations (e.g. “Our WCC nodes count is 7068 vs. 7066—discrepancy due to isolated nodes or data parsing difference.”).

Example Template Snippet (Scala/Python with GraphFrames)

```
from graphframes import GraphFrame
edges = spark.read.csv("wiki-Vote.txt", sep=" ", schema="src INT, dst INT")
vertices = edges.selectExpr("src AS id").union(edges.selectExpr("dst AS
id")).distinct()
```

```
g = GraphFrame(vertices, edges)

# Basic counts
num_nodes = vertices.count()
num_edges = edges.count()

# WCC & SCC
wcc = g.stronglyConnectedComponents(maxIter=10)  # for SCC; use
g.weaklyConnectedComponents for WCC
component_sizes = wcc.groupBy("component").count().orderBy("count",
ascending=False)
largest_scc_nodes = component_sizes.first()['count']

# Triangles & clustering
triangles = g.triangleCount()
avg_clustering = triangles.selectExpr("avg(count)").collect()[0][0]  #
approximate

# Distance metrics: use sample or breadth-first approach
...
```

Students can extend this template to compute all required metrics.

Summary Table (Draft)

Metric	Ground Truth	Your Compute	Notes on Difference
Nodes	7,115	…	…
Edges	103,689	…	…
Largest WCC (nodes)	7,066 (0.993)	…	…
Largest WCC (edges)	103,663 (1.000)	…	…
Largest SCC (nodes)	1,300 (0.183)	…	…
Largest SCC (edges)	39,456 (0.381)	…	…
Avg. clustering coefficient	0.1409	…	…
Number of triangles	608,389	…	…
Fraction of closed triangles	0.04564	…	…
Diameter	7	…	…
Effective diameter (90-percentile)	3.8	…	…