# Assignment 2

**Name: Ahmed Nabil Nour Ahmed ID: 2205245**

## Introduction

Social networks often contain malicious automated accounts bots that attempt to mimic normal user behavior and Detecting these bots depends heavily on graph-based structural features such as degree, clustering coefficient, betweenness, and closeness centrality.

**1. Generate a synthetic social network**
**2. Extract structural graph features**
**3. Train a machine learning classifier to distinguish bots from real users**
**4. Evaluate the impact of two adversarial attacks:**
a. Structural Evasion Attack
b. Graph Poisoning AttackIntroduction

### 1 .Dataset

The data from the file facebook combined.txt

## 2. Graph statistics

• Nodes (users): 4,039
• Edges (friendships): 88,234

## 3.Feature Extraction

1. Degree: number of neighbors. Bots can have unusually small or unusually large degree.
2. clustering coefficient: fraction of neighbor pairs that are connected. Humans often sit in highly clustered friend circles; bots typically do not.
3. Betweenness centrality: how often the node appears on shortest paths between other nodes.
4. Eigenvector centrality: another centrality measure that gives more weight to important neighbors.

5. Community ID: community assignment from the Louvain algorithm.

```
Calculating graph features...
Features calculated!
DataFrame created!
   node  degree  clustering   betweenness
0     0     347    0.041962  1.354765e-01
1     1      17    0.419118  4.033360e-06
2     2      10    0.888889  0.000000e+00
3     3      17    0.632353  2.242632e-06
4     4      10    0.866667  1.238850e-07
```

## 4.Label Assignment

I labeled 5% of all nodes as bots (chosen randomly).
Total labels: Humans: 3838
Bots: 201

## 5. Baseline Bot Detection Model

I trained a Random Forest Classifier using: Features: degree, clustering coefficient, betweenness
Target: bot/human label
Train-test split: 70/30 (stratified)

## Baseline Results

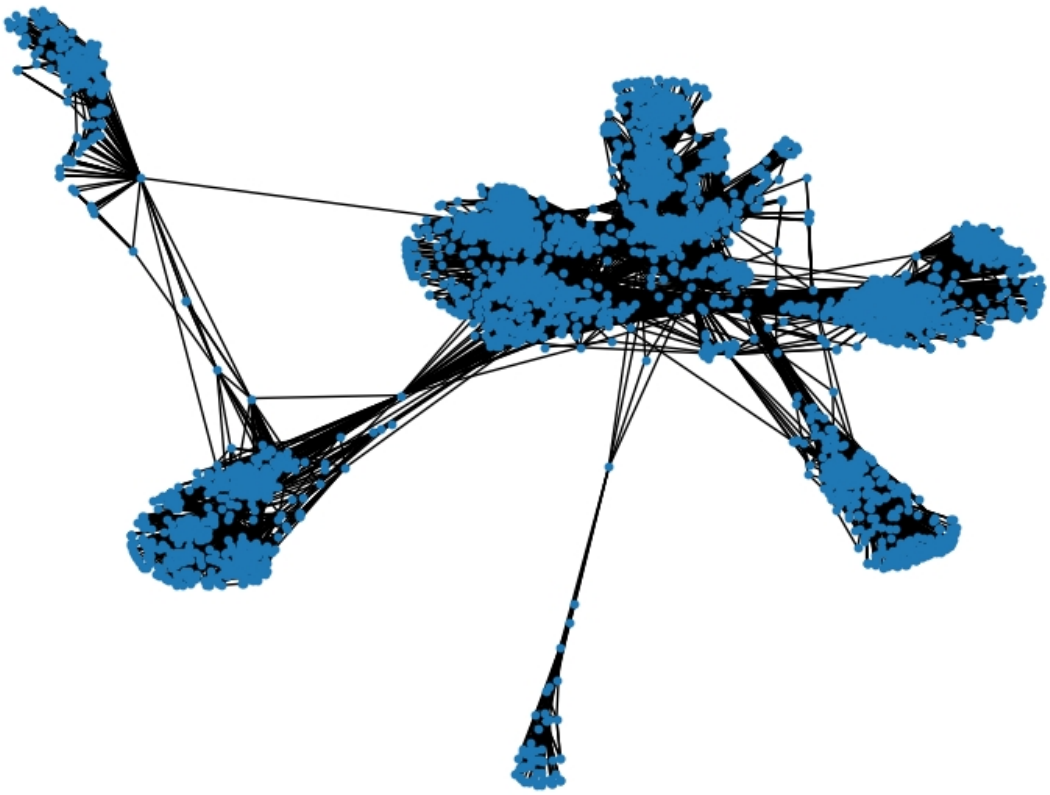
Accuracy: 0.950
Bot Recall: 0.02 (very low)
Bot Precision: 0.50

```
Training baseline bot detection model...

Baseline Model Performance:
Accuracy: 0.9504950495049505
              precision    recall  f1-score   support

           0       0.95      1.00      0.97      1152
           1       0.50      0.02      0.03        60

    accuracy                           0.95      1212
   macro avg       0.73      0.51      0.50      1212
weighted avg       0.93      0.95      0.93      1212
```



## 6. Structural Evasion Attack

**Attack Description**

bots rewire parts of their neighborhood to look more human-like.
The idea is that bots change their connections—without changing their labels—to look more similar to benign nodes in terms of graph features.
For every bot node, I removed 3 edges to reduce its connectivity and distort its structural footprint.
main effects:
1. The degree of bots increases.
2. Bots become more embedded in dense, central parts of the network, which can increase their clustering and change other centrality-based features.

## After Feature Recalculation

Model performance slightly improved for bot detection:
Accuracy: 0.955
Bot Precision: 0.88
Bot Recall: 0.12

```
Applying Structural Evasion Attack...
Structural Evasion Attack applied!

Recalculating features after Structural Evasion...

Evaluating model after Structural Evasion...
Performance After Structural Evasion:
Accuracy: 0.9554455445544554
              precision    recall  f1-score   support

           0       0.96      1.00      0.98      1152
           1       0.88      0.12      0.21        60

    accuracy                           0.96      1212
   macro avg       0.92      0.56      0.59      1212
weighted avg       0.95      0.96      0.94      1212
```

## 7.Graph Poisoning Attack

**Attack Description**

I added 30 fake nodes connected randomly to real users:
Fake nodes were incorrectly labeled as humans
Goal: contaminate training data
This attack tries to mislead the classifier during learning
Graph Statistics After Poisoning
Original nodes: 4039
Added fake nodes: 30
Final total: 4069 nodes
Evaluation After Poisoning
Performance dropped:

Accuracy: 0.944
Bot Precision: 0.00
Bot Recall: 0.00

The model completely fails to detect bots after poisoning because the mislabeled fake nodes distort the learned decision boundaries.

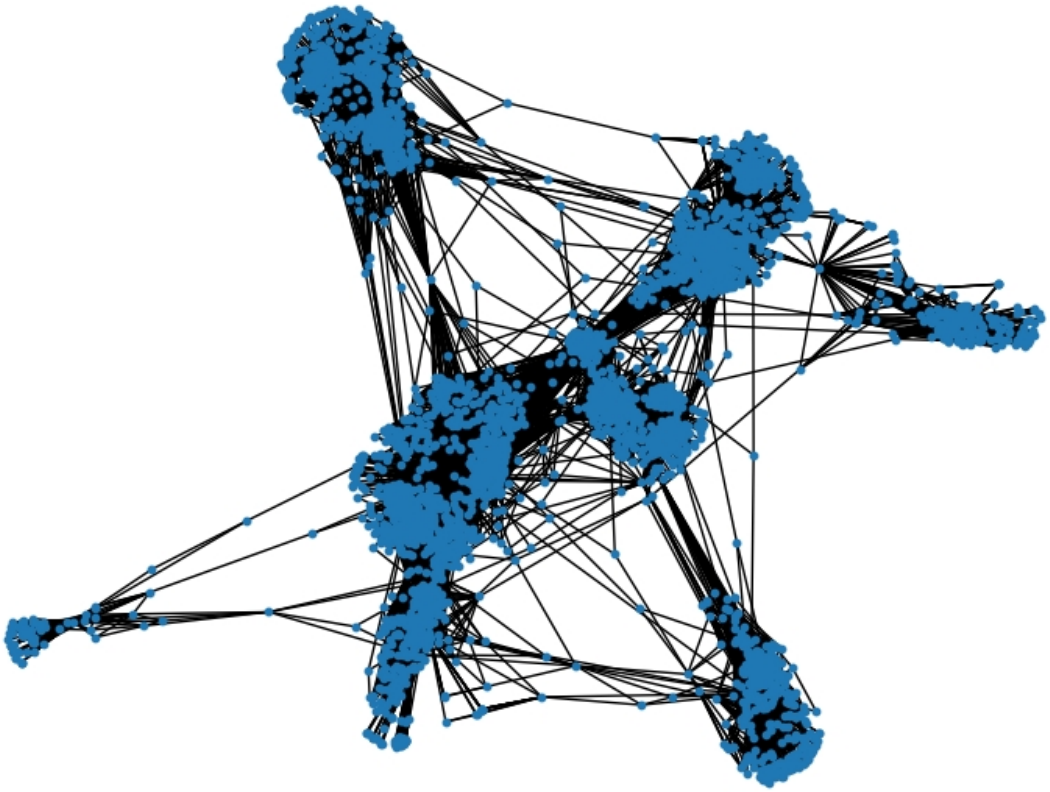
```
Applying Graph Poisoning Attack (robust)...
Fake nodes added: 30
Total nodes after poisoning: 4069

Recalculating features after Graph Poisoning...
Feature calc done in 18.4s
df_p created. Shape: (4069, 5)
  node  degree  clustering   betweenness  label
0    0     348    0.041720  1.438012e-01    0.0
1    1      17    0.419118  1.032532e-06    0.0
2    2      10    0.888889  1.024758e-07    0.0
3    3      17    0.632353  5.556466e-06    0.0
4    4      10    0.866667  0.000000e+00    0.0

Evaluating model after Graph Poisoning...
Performance After Graph Poisoning:
Accuracy: 0.9443079443079443
              precision    recall  f1-score   support

         0.0       0.95      0.99      0.97      1161
         1.0       0.00      0.00      0.00        60

    accuracy                           0.94      1221
   macro avg       0.48      0.50      0.49      1221
weighted avg       0.90      0.94      0.92      1221
```



| Scenario | Accuracy | Bot Precision | Bot Recall |
|---|---|---|---|
| **Baseline** | 0.950 | 0.50 | 0.02 |
| **Structural Evasion** | 0.955 | 0.88 | 0.12 |
| **Graph Poisoning** | 0.944 | 0.00 | 0.00 |