

Explanation of the Code

This code builds a graph with six users and trains a GraphSAGE model to classify each user as benign or malicious based on how they are connected in the network.

1. Creating Node Features

Each node represents a user.

To keep things simple, every user is given just two features: 1) [1, 0] (benign user) & 2) [0, 1] (malicious user)

So nodes 0, 1, and 2 are benign, and nodes 3, 4, and 5 are malicious.

2. Defining the Edges

- Benign users (0, 1, 2) are fully connected to each other.
- Malicious users (3, 4, 5) are also fully connected to each other.
- There is only one connection between the two groups:
node 2 is connected to node 3.

This forms two tight clusters with a single weak link between them. GraphSAGE can easily detect this separation.

3. Adding Labels

Each node gets a label: 0 = benign , 1 = malicious. These are the ground-truth values the model must learn.

4. Building the GraphSAGE Model

A simple 2-layer GraphSAGE model is created:

1. **First SAGE layer**
 - Reads each node's neighbors
 - Aggregates their information
 - Passes it through a ReLU activation

2. Second SAGE layer

- Produces the final output (2 numbers per node)
- These represent scores for benign and malicious

The model ends with a log-softmax to produce log-probabilities.

5. Training the Model

The model is trained for 50 epochs using the Adam optimizer.

At each step:

1. The graph is passed through the model.
2. The loss is calculated by comparing predictions with true labels.
3. The model adjusts its weights to reduce the loss.

Because the graph is very clean (two clear clusters), the model learns quickly.

6. Making Predictions

After training, the model predicts the label for each node.

Output: Predicted labels: [0, 0, 0, 1, 1, 1]

This means:

- Nodes 0, 1, 2 → predicted as benign
- Nodes 3, 4, 5 → predicted as malicious

The model achieves accuracy.

Summary

This code builds a tiny network of users and trains a GraphSAGE graph neural network to classify each user as benign or malicious based on both their features and their connections in the graph. Since benign users mostly connect to each other and malicious users do the same, the model quickly learns to separate the two groups and correctly labels all nodes.