Project:

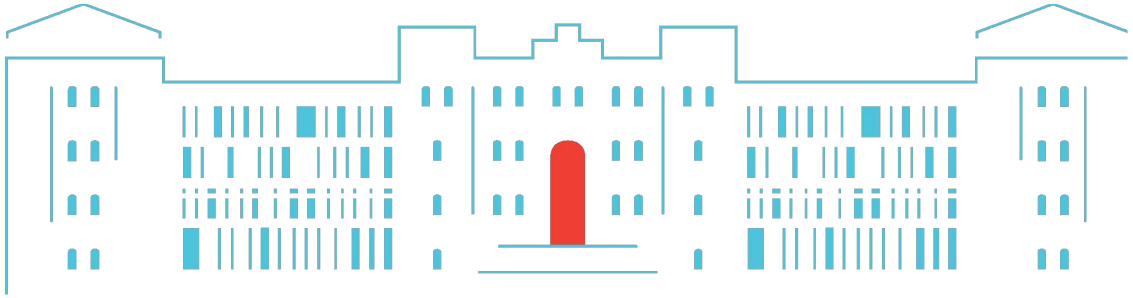
Artist Collaborations GNN.

Manoj Nethenahalli Dhanpal -611794

Varad Santosh Kulkarni - 612254

Richart Seel - 598756

TUHH
Technische Universität Hamburg



29-01-25

INTRODUCTION

- **Cleaned edges.csv and nodes.csv dataset.**
- **Generate negative edges.**
- **Perform Node2Vec.**
- **Node Embeddings and Dimensionality Reduction.**
- **Train and Validate using GNN and Compare with GAT**
- **Sample Prediction**

Import DataSet

TUHH

- Importing cleaned dataset.
- Mapping spotify id to indices and constructing source and target attributes with artist Id.

•[3]:

```
# Mapping Spotify IDs to indices for graph construction
node_index_map = {spotify_id: idx for idx, spotify_id in enumerate(nodes_df['spotify_id'])}
edges_df['Source'] = edges_df['id_0'].map(node_index_map)
edges_df['Target'] = edges_df['id_1'].map(node_index_map)


# Creating edge index (two rows: source and target nodes)
edge_index = torch.tensor(edges_df[['Source', 'Target']].to_numpy().T, dtype=torch.long)
```

[13]:

```
node_keys = list(graph.nodes)
print(len(node_keys))
```

```
135057
```

Perform Node2Vec

TUHH

- **RGraph Creation:** Convert the edge list (`edge_index`) into a NetworkX graph.
- **Node2Vec Setup:** Initialize Node2Vec with specific parameters to learn node embeddings.
- **Training:** Fit the Node2Vec model to generate embeddings for each node based on random walks.
- **Handle Missing Nodes:** Provide zero-vector embeddings for nodes not present in the model.

Node Embedding dimension reduction.

TUHH

- **Perplexity_value**: 5 for t-SNE to balance local and global data patterns, ensuring it's suitable for the subset size.

- **Dimensionality Reduction**: Use t-SNE to effectively reduce high-dimensional node embeddings to 2D for visualization.

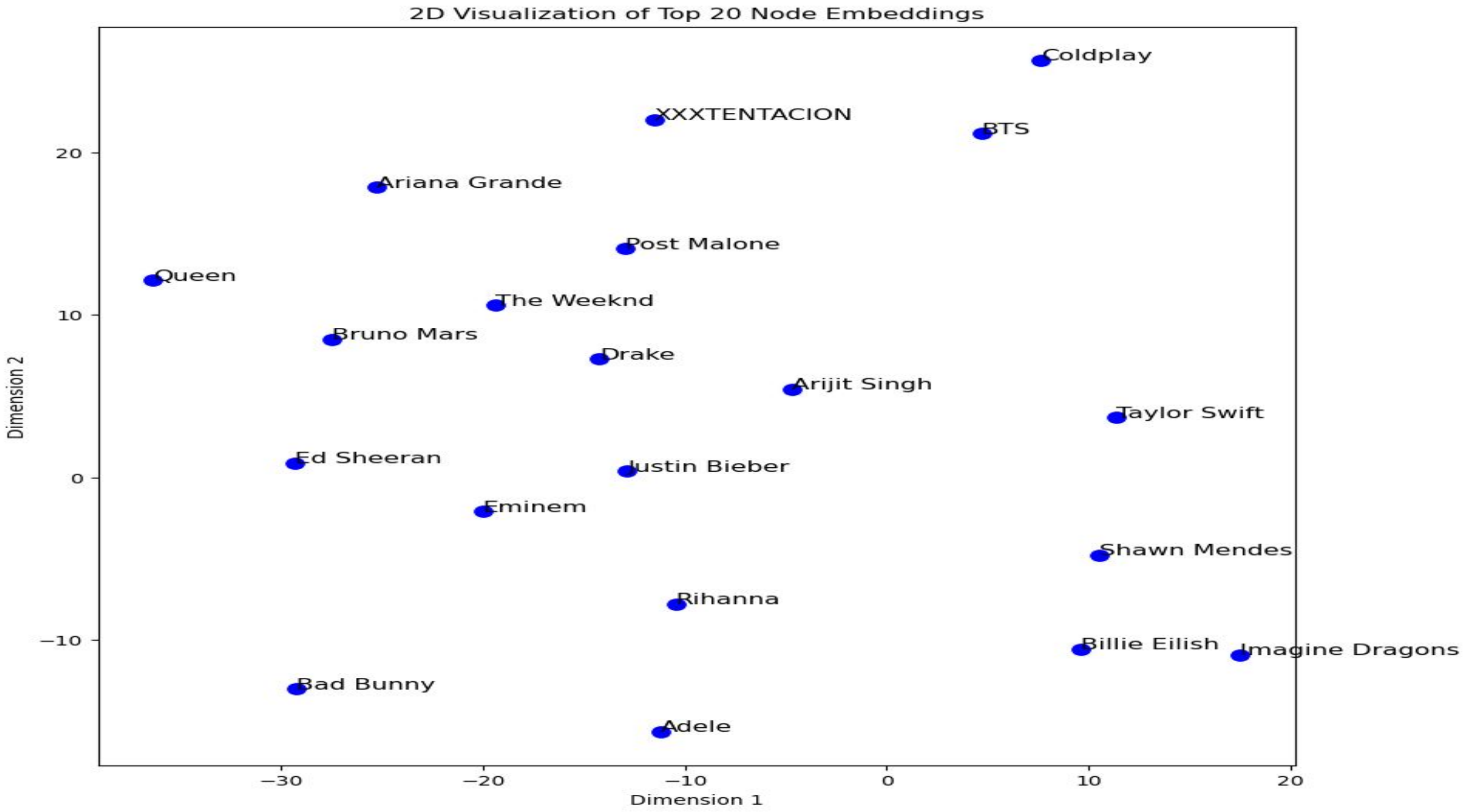
- **Subset Selection**: Select the top 20 nodes.

- **Apply t-SNE**: Perform dimensionality reduction on the selected node embeddings to extract meaningful 2D representations.

- **Scatter Plot**: Create a 2D scatter plot to visually represent the reduced embeddings of the selected nodes.

2D Visualization of Top 20 Node Embeddings

TUHH



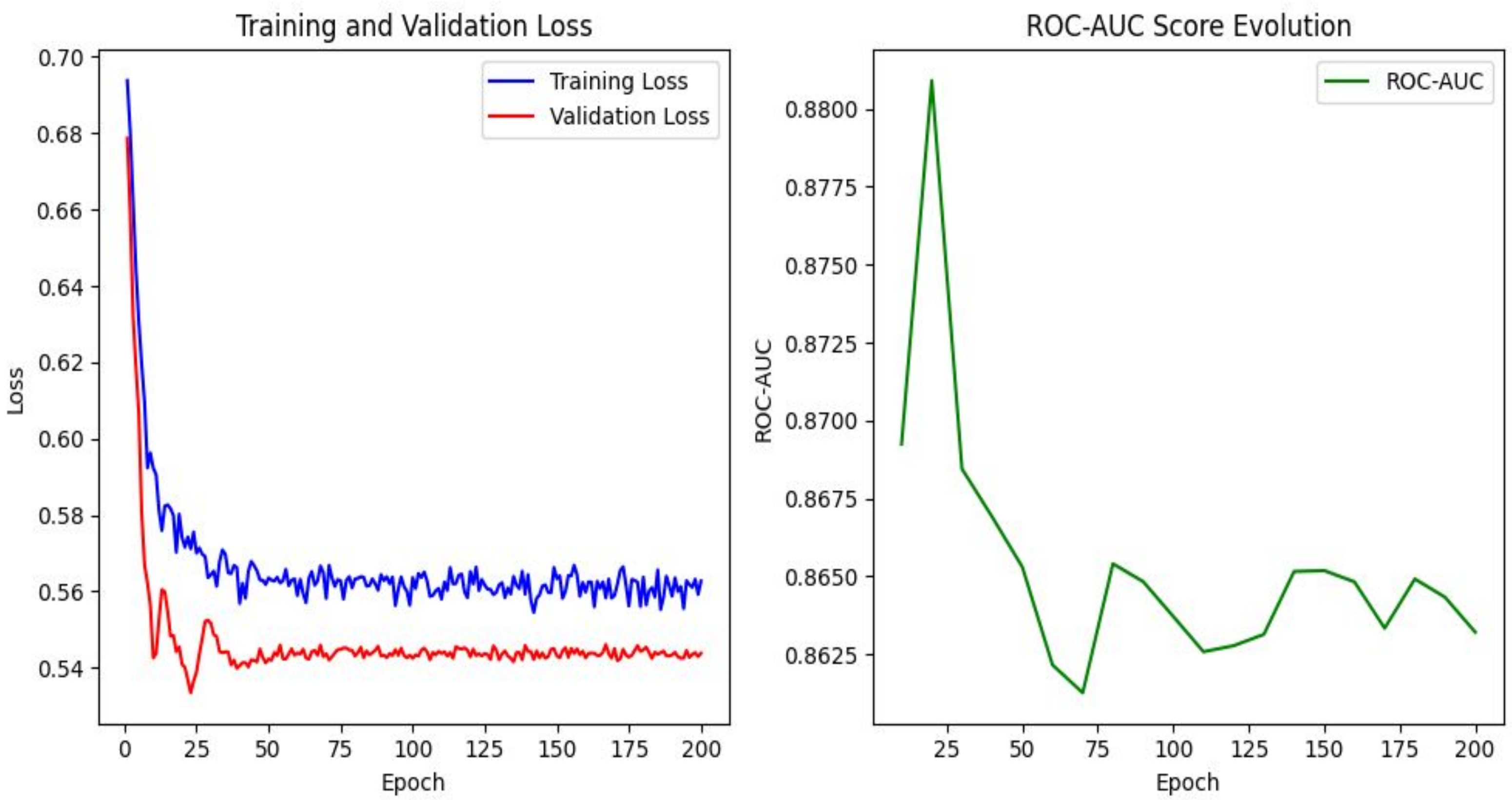
Train and Validate GNN Model

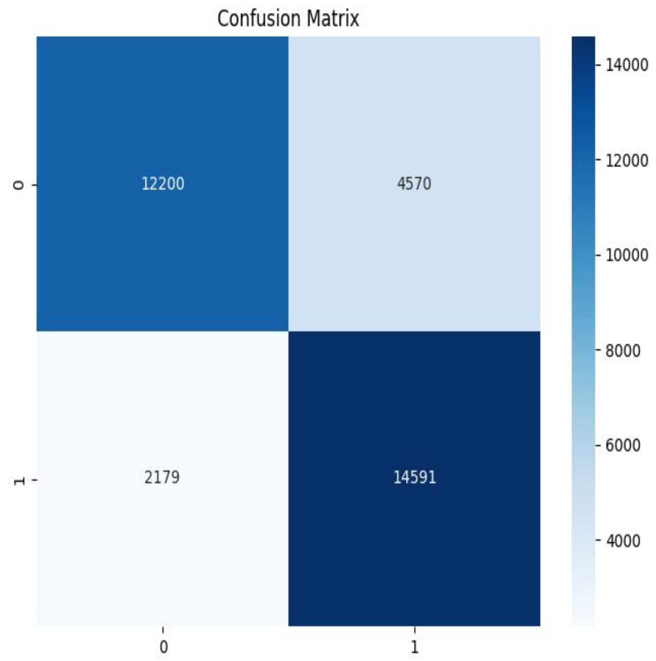
TUHH

```
Epoch 200, Loss: 0.2010, Test Accuracy: 0.9348

Confusion Matrix:
True Negatives (TN): 47555
False Positives (FP): 5812
False Negatives (FN): 1131
True Positives (TP): 51921

Interpretation:
TN: Predicted no collaboration correctly (actual no collaboration)
FP: Predicted collaboration incorrectly (actual no collaboration)
FN: Predicted no collaboration incorrectly (actual collaboration)
TP: Predicted collaboration correctly (actual collaboration)

Classification Report:
                  precision    recall  f1-score   support

No Collaboration       0.98      0.89      0.93     53367
   Collaboration       0.90      0.98      0.94     53052

        accuracy                           0.93    106419
       macro avg       0.94      0.93      0.93    106419
    weighted avg       0.94      0.93      0.93    106419
```

GAT (Graph Attention Network) Model

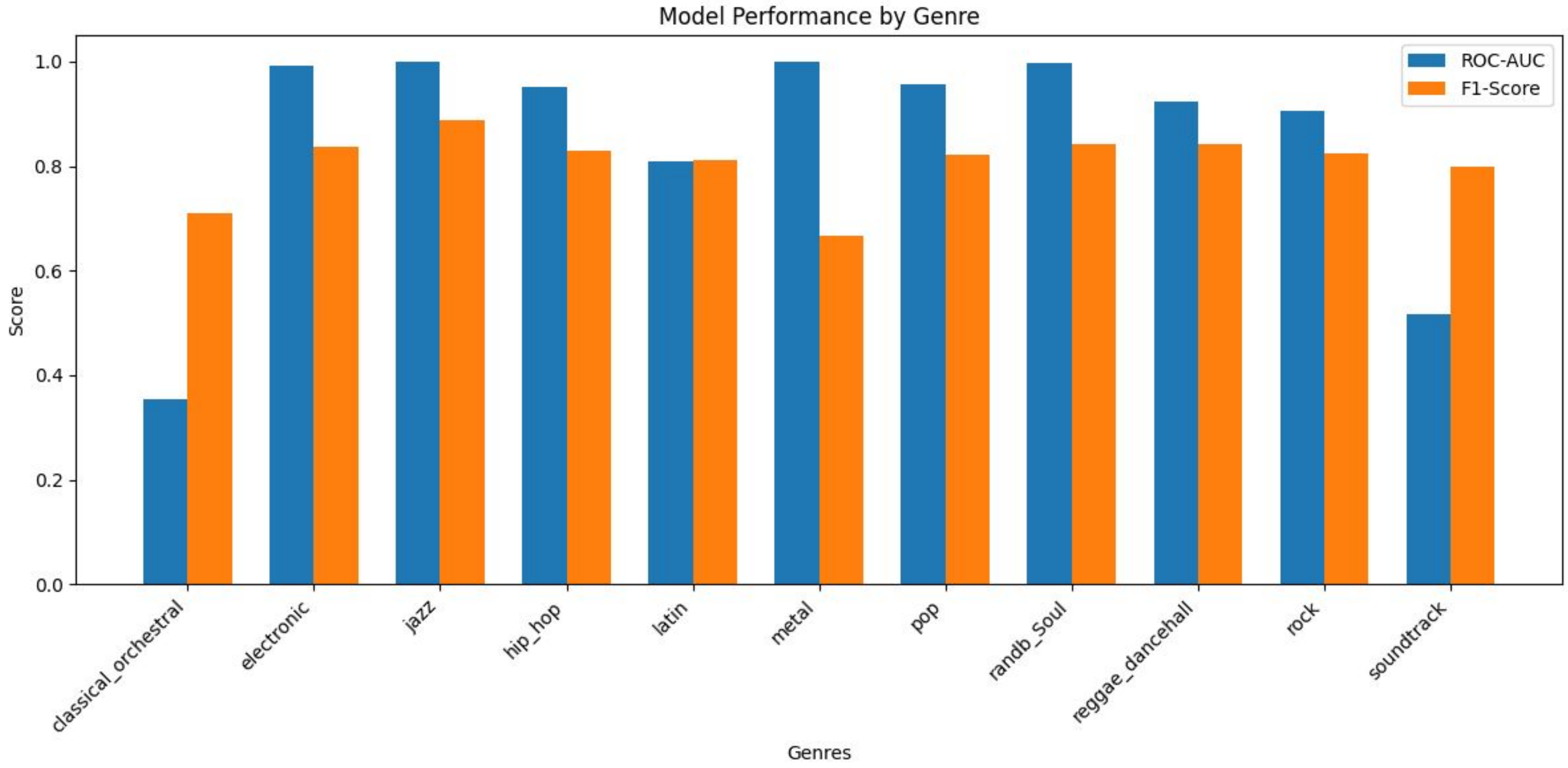
TUHH



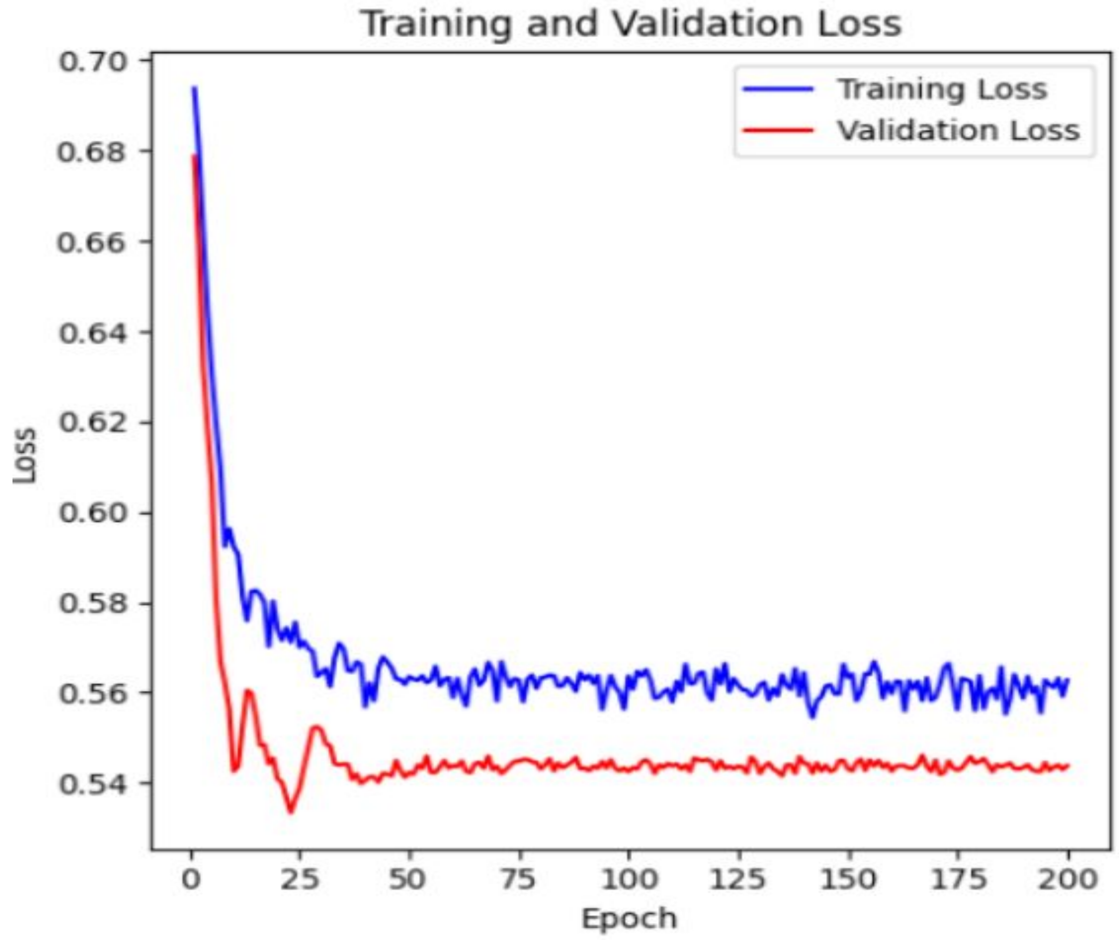


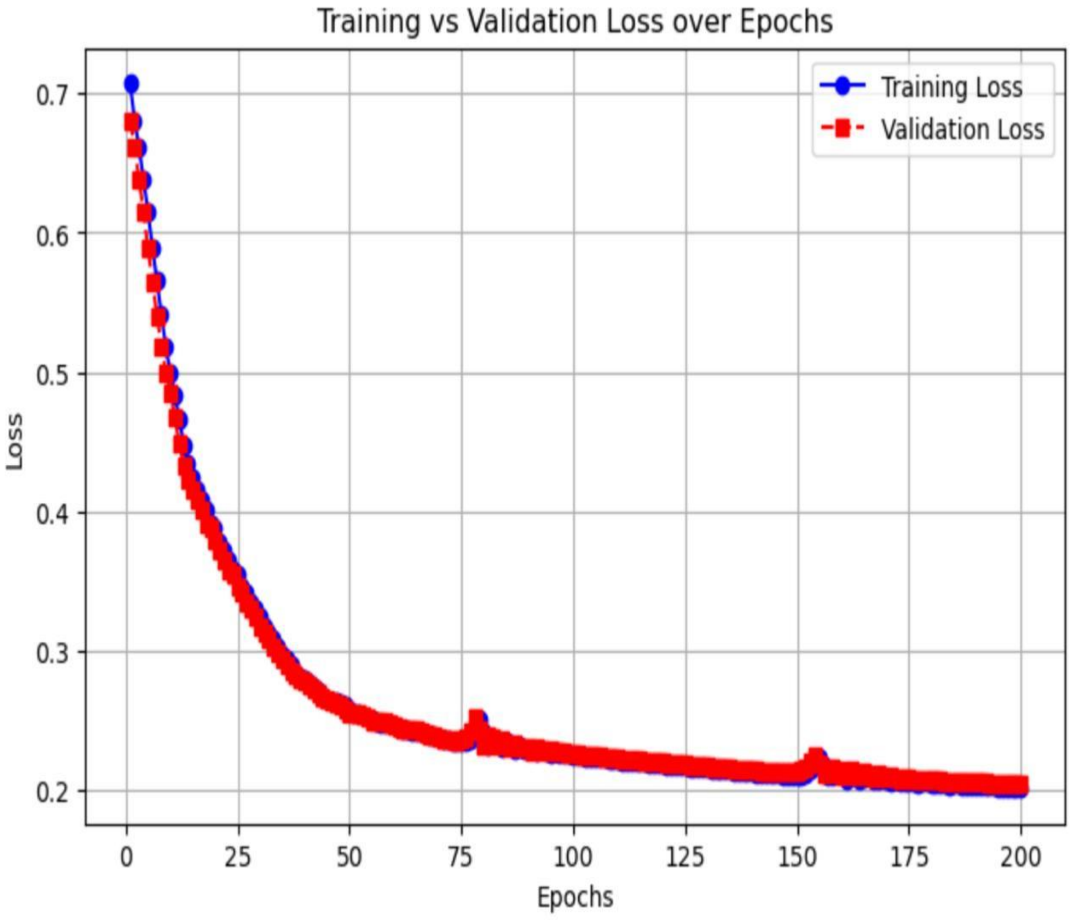

```
Epoch: 180, Loss: 0.5474, Val Loss: 0.5511, Val ROC-AUC: 0.8521
Epoch: 190, Loss: 0.5464, Val Loss: 0.5489, Val ROC-AUC: 0.8505
Epoch: 200, Loss: 0.5514, Val Loss: 0.5501, Val ROC-AUC: 0.8515
```

GAT (Graph Attention Network) Model



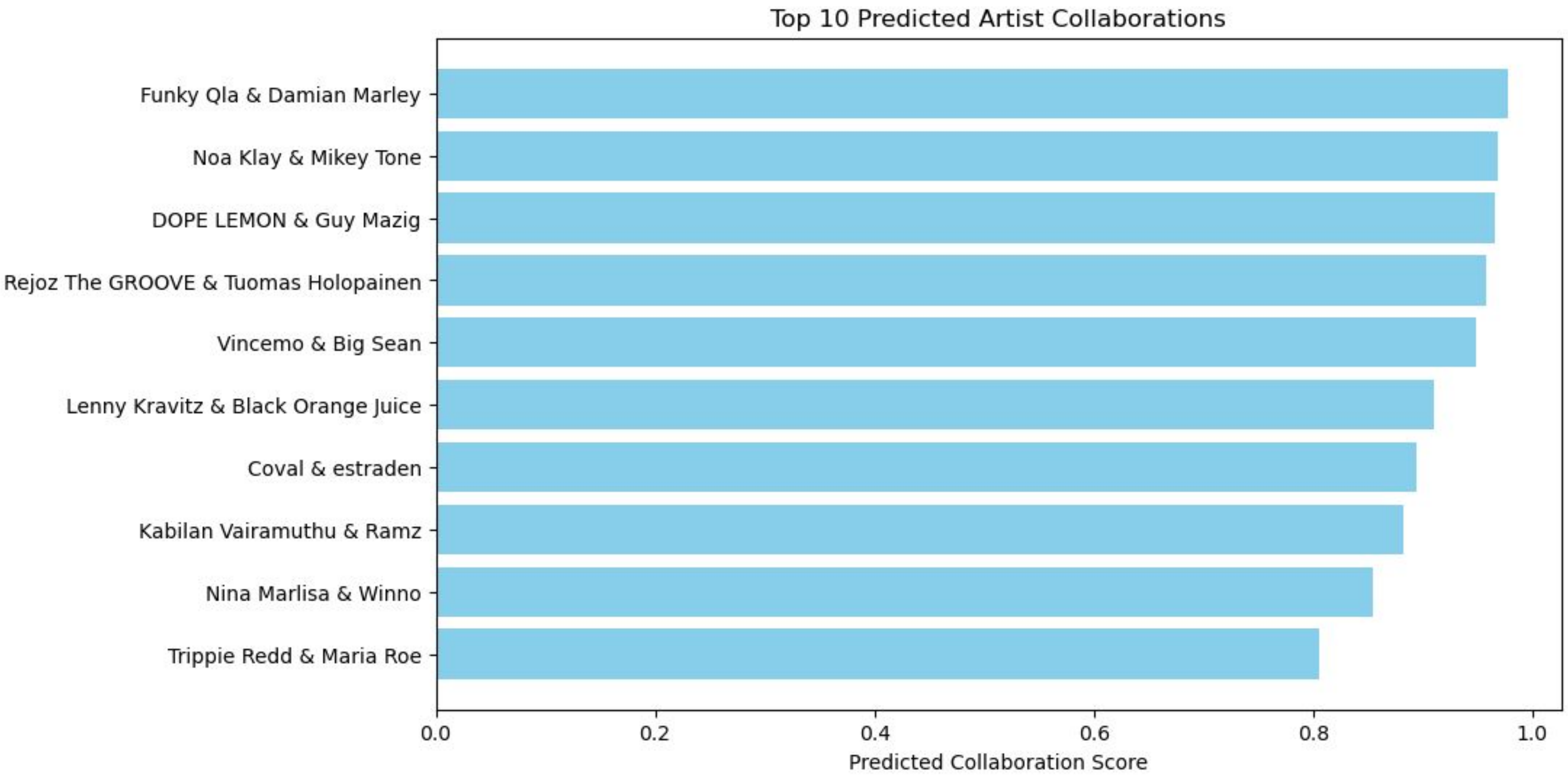
Comparing GAT (Left) and GNN (Right) Training loss.





Prediction for 10 Artist Collaborations.

TUHH



CONCLUSION

- Created equal negative edges to avoid bias.
- Performed Node2Vec.
- Used Node embedding and reduced dimensions for simplicity.
- Trained and Validated model.
- Compared GNN and GAT
- Predicted the top 10 collaborations.

THANK YOU!