Subgraph Pattern Matching with Deep Representations

Team Member: Joe Lou, Yue Zhang, Ziyi Yang
Mentor: Rex Ying, Jiaxuan You, Jure Leskovec
Subgraph Matching
Method: Graph Convolutional Networks (GCNs)
Single Query Matching
Single Query Matching

train_AUROC_overall

test_AUROC_
Family Classification
Family Classification

Train AUROC Overall

Test AUROC
Generalizing
Generalizing

train_AUROC_overall

test_AUROC_
Random Training (In Progress)
Order embedding

- Query graph node embedding (Y), Test graph node embedding (X)

- Comparing embeddings directly reduces computational cost.

- Penalizing Order Violations: $E(x, y) = ||\max(0, y - x)||^2$

- Max-margin loss

$$\sum_{(u, v) \in P} E(f(u), f(v)) + \sum_{(u', v') \in N} \max\{0, \alpha - E(f(u'), f(v'))\}$$

- Hyperparameters introduced: Margin $\alpha$, dimension threshold $b$
Order embedding on multi-query experiment

![Graphs showing train AUROC and test AUROC over a range of values.]
Order embedding Results

- Compare the node embeddings for common subgraph

- 1 is subgraph of 2 and 2 is subgraph of 3.
- Node embeddings have dimension 60.
- Out of 60 elements,
  - 0 elements in embedding 1 > 2 (expect < 12)
  - 9 elements in embedding 2 > 3 (expect < 12)
  - 27 elements in embedding 4 > 5 (expect > 12)
  - 36 elements in embedding 5 > 6 (expect > 12)
Order embedding Results

1

2

3
Conclusions & Future Work

- Delivered a Graph Convolutional Neural Network model for subgraph matching.
- Order embedding method was used to train node embeddings that can be compared directly to reduce computational cost.