1 Task

Attack graph models.

- Model types:
  - Inductive Graph Classification (Graph classification): For example, drug molecular classification.
  - Transductive Node Classification (Node classification): Classifying papers in a citation database.

- Means:
  - Add or Delete edges
  - Delete nodes via delete edges

- Restriction on the modification:
  - Explicit semantics (Oracle) will tell whether the modification is valid. Used in synthetic data.

\[
I(G, \tilde{G}, c) = I(f^*(G, c) = f^*(\tilde{G}, c)),
\]

- Small modification. Number of edges is limited to \(N\), and within a neighborhood graph where the distance between nodes are less than \(b\).

\[
I(G, \tilde{G}, c) = I(|E - \tilde{E}| \cup (\tilde{E} - E)| < m) \cdot I(\tilde{E} \subseteq N(G, b)).
\]
2 RL attack Overview

3 Algorithm

3.1 Black-box Hierarchical Reinforcement Learning

- **In one sentence:** Use Deep Q Learning to generate edge modification
- **RL setting:**
  - **State:** State is represented by \((\hat{G}_t, c)\), where \(\hat{G}_t\) is a partially modified graph and \(c\) is the original label.
  - **Action:** Action is add or remove an edge. Because the space is too large, the authors use hierarchical structure to model actions: Pick first node, and pick the second node, with two different Q function.
  - **Terminal State:** When \(m\) edge is modified, the process stops.
  - **Reward:** Non-zero reward is only received at the terminal state.
- **Value iteration:**
  \[
  Q^{1*}(s_t, a_t^{(1)}) = \max_{a_t^{(2)}} Q^{2*}(s_t, a_t^{(1)}, a_t^{(2)})
  \]
  \[
  Q^{2*}(s_t, a_t^{(1)}, a_t^{(2)}) = r(s_t, a_t = (a_t^{(1)}, a_t^{(2)})) + \max_{a_{t+1}^{(1)}} Q^{1*}(s_{t+1}, a_{t+1}^{(1)}).
  \] (2)
- **Parameterization of Q:** The authors find that time and previous modification is not useful in Q, therefore:
  - For \(Q_1\), use a structure2vec weight \(\mu_1\)
    \[
    Q^{1*}(s_t, a_t^{(1)}) = W_{Q_1}^{(1)} \sigma(W_{Q_1}^{(2)\top} [\mu_{a_t^{(1)}}, \mu(s_t)]),
    \] (3)
  - For \(Q_2\), use with an extra consideration of the chosen node \(a\)
    \[
    Q^{2*}(s_t, a_t^{(1)}, a_t^{(2)}) = W_{Q_2}^{(1)} \sigma(W_{Q_2}^{(2)\top} [\mu_{a_t^{(1)}}, \mu_{a_t^{(2)}}, \mu(s_t)])
    \] (4)
3.2 Other attacks

- **Random Sampling**: the simplest baseline.

- **Gradient based attack (white-box attack)**:
  Sort the gradient of all edges, greedily pick the edge with largest gradient.
  
  $$
  \hat{G}_{t+1} = \begin{cases}
  (\hat{V}_t, \hat{E}_t \setminus \{(u_t, v_t)\}) : \frac{\partial L}{\partial \alpha_{u_t, v_t}} < 0 \\
  (\hat{V}_t, \hat{E}_t \cup \{(u_t, v_t)\}) : \frac{\partial L}{\partial \alpha_{u_t, v_t}} > 0
  \end{cases}
  \tag{5}
  $$

- **Genetic programming**:
  - **Fitness function**: $L(f(\hat{G}_j^{(r)}, c), y)$, similar to reward.
  - **Selection**: Do a weighted sampling/greedy selection to select the ‘breeding’ population $P^{(r)}_b$.
  - **Crossover**: After selection, randomly pick two candidates and mixing the edges together:
    $$
    \hat{G}' = (V, (\hat{E}_1 \cap \hat{E}_2) \cup \text{rp}(\hat{E}_1 \setminus \hat{E}_2) \cup \text{rp}(\hat{E}_2 \setminus \hat{E}_1)).
    \tag{6}
    $$
  - **Mutation**: Pick a solution $(u_t, v_t)$, have a certain probability to change it to either $(u_t, v'_t)$ or $(u'_t, v_t)$.

4 Experiment

- **Tasks**:
  - **Graph Level Attack**:
    * Synthesize 15K graphs using Erdos-Renyi graph model
    * Predict number of connected components (1, 2, 3)
  - **Node Level Attack** Core, Citeseer, Pubmed and Finance dataset.

- **Target**: Structure2vec model

- Result on graph classification attack: Genetic programming give the best result in comparison.
Result on node classification attack: RL gets a result that closes to the exhaust result.

<table>
<thead>
<tr>
<th>Settings</th>
<th>Methods</th>
<th>15-20 nodes</th>
<th>40-50 nodes</th>
<th>90-100 nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>K = 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K = 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K = 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>K = 5</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>RBA</td>
<td>RandSampling</td>
<td>89.6%</td>
<td>99.2%</td>
<td>98.7%</td>
</tr>
<tr>
<td>WBA</td>
<td>GradArgmax</td>
<td>87.9%</td>
<td>98.2%</td>
<td>97.6%</td>
</tr>
<tr>
<td>PBA-C</td>
<td>GeneticAlg</td>
<td>91.4%</td>
<td>92.7%</td>
<td>98.9%</td>
</tr>
<tr>
<td>PBA-D</td>
<td>RL-S2V</td>
<td>93.9%</td>
<td>95.3%</td>
<td>97.3%</td>
</tr>
</tbody>
</table>

Restricted black-box attack on test set II

<table>
<thead>
<tr>
<th>Method</th>
<th>Citeseer</th>
<th>Cora</th>
<th>Pubmed</th>
<th>Finance</th>
</tr>
</thead>
<tbody>
<tr>
<td>(unattached)</td>
<td>71.60%</td>
<td>81.00%</td>
<td>79.90%</td>
<td>88.67%</td>
</tr>
<tr>
<td>RBA, RandSampling</td>
<td>67.60%</td>
<td>78.50%</td>
<td>79.00%</td>
<td>87.44%</td>
</tr>
<tr>
<td>WBA, GradArgmax</td>
<td>63.00%</td>
<td>71.30%</td>
<td>72.4%</td>
<td>86.33%</td>
</tr>
<tr>
<td>PBA-C, GeneticAlg</td>
<td>63.70%</td>
<td>71.20%</td>
<td>72.30%</td>
<td>85.96%</td>
</tr>
<tr>
<td>PBA-D, RL-S2V</td>
<td>62.70%</td>
<td>71.20%</td>
<td>72.80%</td>
<td>85.45%</td>
</tr>
<tr>
<td>Exhaust</td>
<td>62.50%</td>
<td>70.70%</td>
<td>71.80%</td>
<td>85.22%</td>
</tr>
</tbody>
</table>

(a) pred = 2 (b) pred = 1 (c) pred = 0