Node Feature Kernels Increase Graph Convolutional Network Robustness

Mohamed El Amine Seddik
Changmin Wu
Johannes Lutzeyer
Michalis Vazirgiannis

Message-Passing Framework

- Graph Convolutional Network (GCN, [3]): $\sigma(AX\Theta)$.
- Input: graph structure $A$ and node features $X$.
- Procedure: Aggregation step $AX$ and Update step $X\Theta$.

Research Problem

**Question 1:** What is the influence of the graph structure information and node feature information on each other in the GCN?

**Question 2:** How will the inference drawn from a GCN be impacted by graph structural noise and is there a way to enhance its robustness to such noise?

Take-away Message

The message-passing step dilutes (or in the extreme case completely ignores) information present in the node features if the underlying graph structure is noisy (or in the extreme case completely random). Adding a node feature kernel addresses this problem.

- Replace trainable weight $\Theta$ with a normally distributed matrix, i.e., $\sigma(AXW)$, where $W_{ij} \sim \mathcal{N}(0,1), A \in \mathbb{R}^{n \times n}, X \in \mathbb{R}^{n \times p}$ and $W \in \mathbb{R}^{p \times d}$.
- Insight from this model: we study the spectral behaviour of Gram matrix $G = \frac{1}{n} \sigma(AXW)\sigma(W^T X^T A^T)$, specifically the eigenvector of $G$ corresponding to its largest eigenvalue (the informative eigenvector).

Why RandomGCN?

- To enable a Random Matrix Theory (RMT) analysis and its powerful tools in the theoretical study of neural networks.
- Empirically, RandomGCNs can achieve comparable results with the vanilla GCN, i.e., no training is needed for the update weights in high dimensions.

Observation If a graph is sufficiently perturbed, then the GCN will fail to benefit from the node features no matter how informative they are.

Intuitive Explanation In a message-passing framework, node features are aggregated over graph neighbourhoods. When these neighbourhoods are random, we are aggregating random subsets of node features, thus destroying potential information.

Proposed Solution

This can be addressed by using the node feature information to directly inform the structure of the GCNs message passing scheme $eA = (1 - e)I$. 

**Figure 1:** In high dimensions: GCN and RandomGCN exhibit equivalent performance.

**Figure 2:** Adding a node feature kernel helps reconstruct meaningful neighbourhoods.

We evaluate the robustness of the GCN model on the node classification task with two structural perturbation schemes: edge deletion of ratio $\alpha$ and edge insertion of ratio $\beta$. Consistent with our theoretical analysis, we use a single-layer GCN model (with MLP readout).

For simplicity we use the linear kernel $K_{ij} = x_i x_j$ and sparsify the dense kernel with the adjacency matrix $K = A$.

**Stochastic Block Models (Community number: 2):**

| $\alpha$ | GCN | Confl GCN | kGCN | k-jk GCN | GCN
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>90.6</td>
<td>89.8</td>
<td>88.5</td>
<td>85.0</td>
<td>87.7</td>
</tr>
<tr>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- The performance of the GCN degrades on SBMs with cluster structure (homophilic/heterophilic) as a result of edge-deletion and edge-insertion noise.
- Addition of the proposed kernel improves GCNs robustness against graph structural noise.

**Citation/Co-purchase/Co-author graphs**

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>GCN</th>
<th>Confl GCN</th>
<th>kGCN</th>
<th>k-jk GCN</th>
<th>GCN</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>88.0</td>
<td>85.4</td>
<td>84.9</td>
<td>81.4</td>
<td>86.3</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Edge-insertion noise seems to have a greater impact on real-world graphs.
- Node feature kernels can largely compensate the performance reduction caused by graph structural noise.

**Deeper GCN model (4 layers):**

<table>
<thead>
<tr>
<th>$\alpha$</th>
<th>GCN</th>
<th>Confl GCN</th>
<th>kGCN</th>
<th>k-jk GCN</th>
<th>GCN</th>
<th>CI</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0</td>
<td>93.9</td>
<td>91.5</td>
<td>88.9</td>
<td>86.5</td>
<td>92.4</td>
<td></td>
</tr>
<tr>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Our proposed kernel performs better or on par with Jumping Knowledge [5] and GCNII [1] under all considered schemes.
- It can be further combined with JK to improve the performance.

We also observed similar behaviour for other GNNs (GIN[6], GraphSage[2] and GAT[4]). More experiments are on our paper.

---