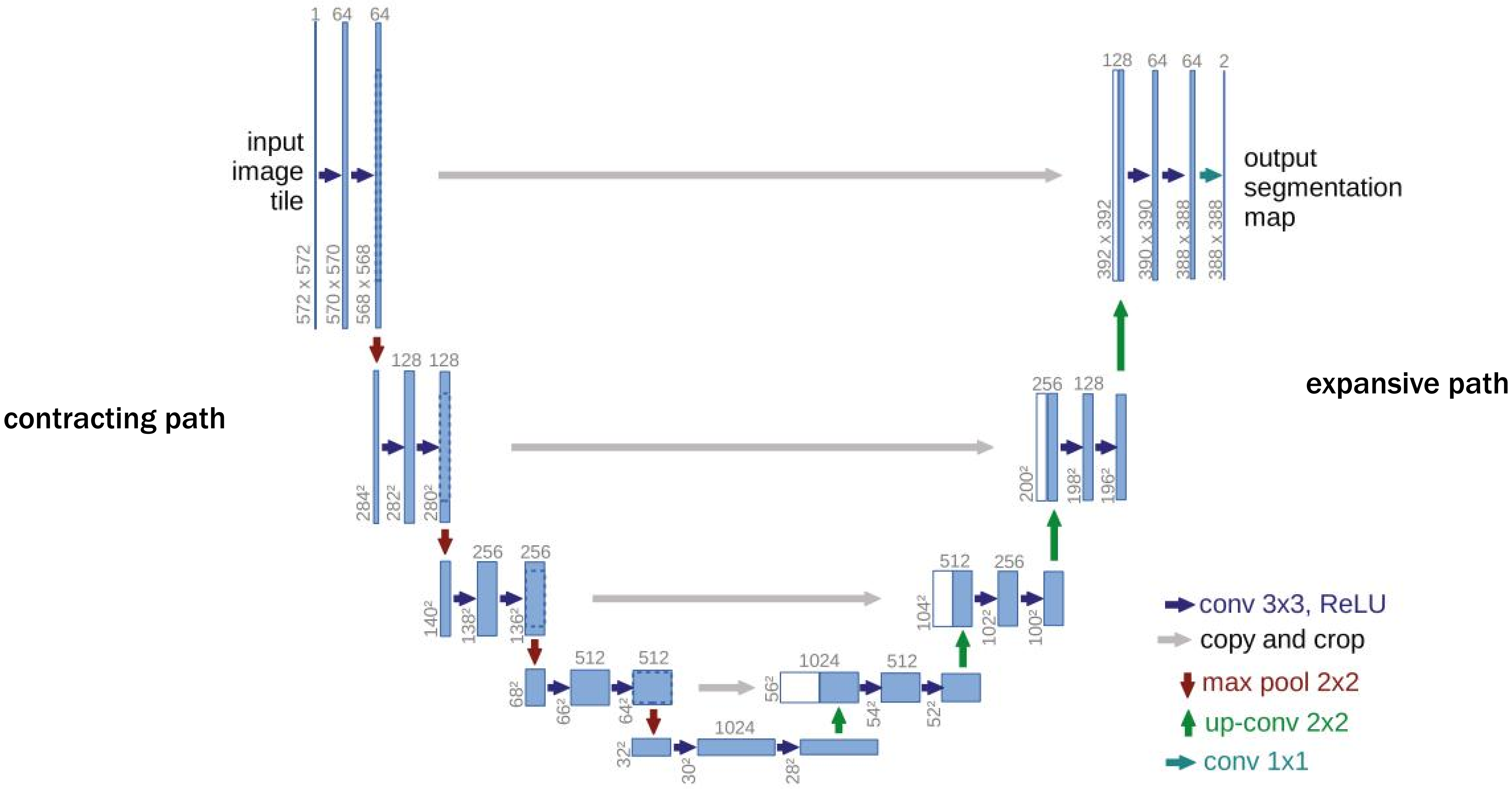
Graph U-Nets

Hongyang Gao   Shuiwang Ji

ICML-2019

Tao Lei

2019/09/29



U-Net: Convolutional Networks for Biomedical Image Segmentation

Gap

1. Given images are special cases of graphs withnodes lie on 2D lattices, graph embedding tasks have a natural correspondence with image pixel-wise prediction tasks such as segmentation.

2. While encoder-decoder architectures like U-Nets have been successfully applied on many image pixel-wise prediction tasks, similar methods are lacking for graph data.

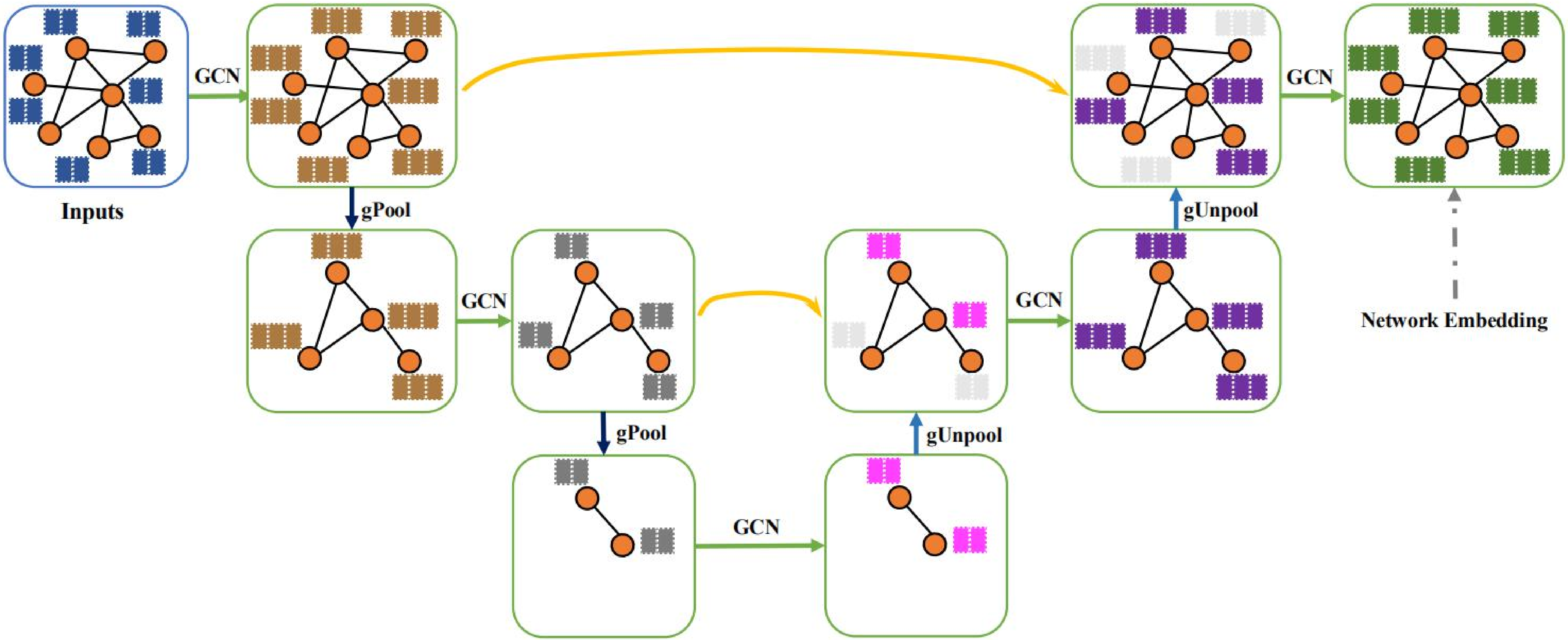
3. This is due to the fact that pooling and up-sampling operations are not natural on graph data.

Graph U-Net

gPool: adaptively selects some nodes to form a smaller graph based on their scalar projection values on a trainable projection vector.

gUnpool: restores the graph into its original structure using the position information of nodes selected in the corresponding gPool layer

Graph U-Net



GCN
Inputs
gPool
GCN
gPool
GCN
gUnpool
GCN
gUnpool
GCN
Network Embedding

Graph Pooling Layer

The layer-wise propagation rule of the graph pooling layer is defined as:

$$
\begin{aligned}
\mathbf{y} &= X^{\ell}\mathbf{p}^{\ell}/\|\mathbf{p}^{\ell}\|, \\
\mathrm{idx} &= \mathrm{rank}(\mathbf{y}, k), \\
\tilde{\mathbf{y}} &= \mathrm{sigmoid}(\mathbf{y}(\mathrm{idx})), \\
\tilde{X}^{\ell} &= X^{\ell}(\mathrm{idx}, :), \\
A^{\ell+1} &= A^{\ell}(\mathrm{idx}, \mathrm{idx}), \\
X^{\ell+1} &= \tilde{X}^{\ell} \odot \left(\tilde{\mathbf{y}}\mathbf{1}_C^T\right),
\end{aligned}
\tag{2}
$$

where k is the number of nodes selected in the new graph.rank(y, k) is the operation of node ranking, which returns indices of the k-largest values in y.

Graph Pooling Layer

Given a node i with its feature vector xi, the scalar projection of xi on p is:

$$y_i = \mathbf{x_i}\mathbf{p}/||\mathbf{p}||$$

yi measures how much information of node i can be retained when projected onto the direction of p.

Graph Pooling Layer

we employ a gate operation to control information flow.

$$\tilde{\mathbf{y}} = \text{sigmoid}(\mathbf{y}(\text{idx}))$$

Notably, the gate operation makes the projection vector p trainable by back-propagation (LeCun et al., 2012). Without the gate operation, the projection vector p produces discrete outputs, which makes it not trainable by back-propagation.

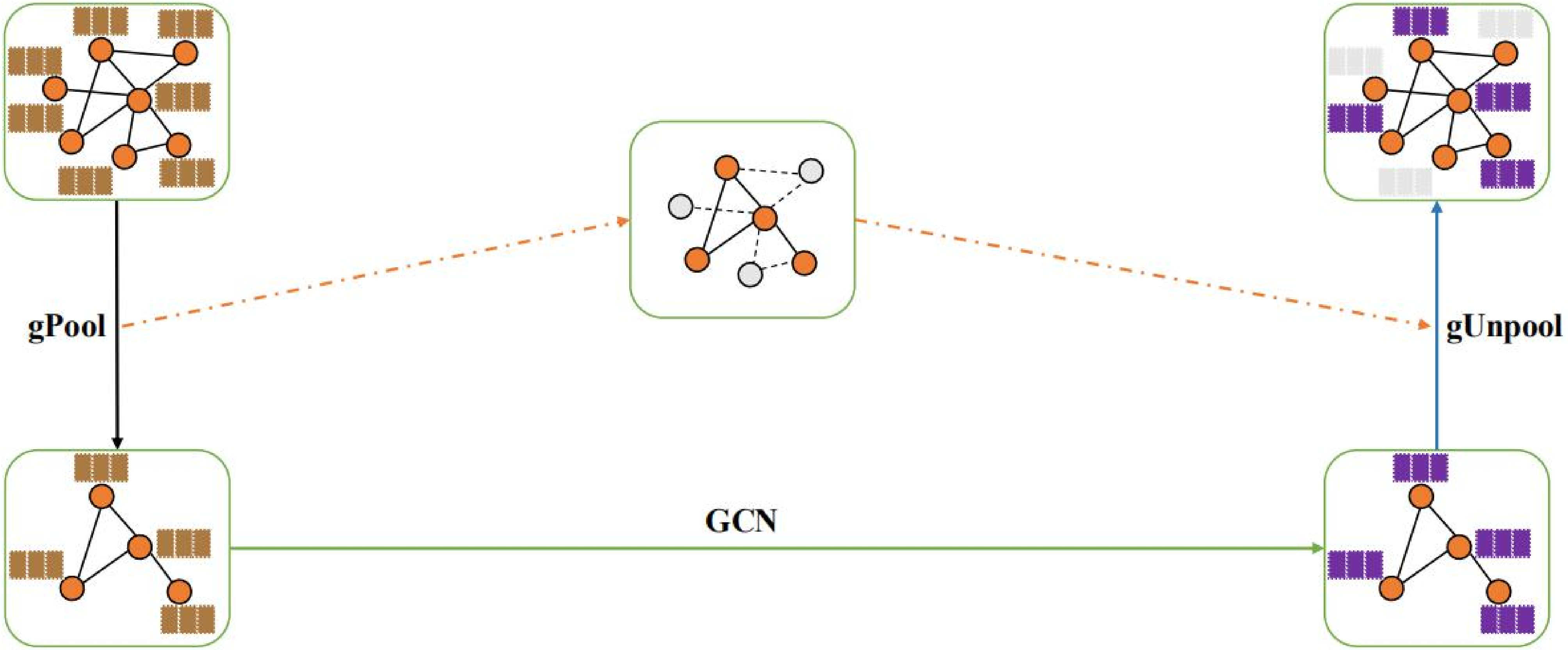
Graph Unpooling Layer

Formally, we propose the layer-wise propagation rule of graph unpooling layer as

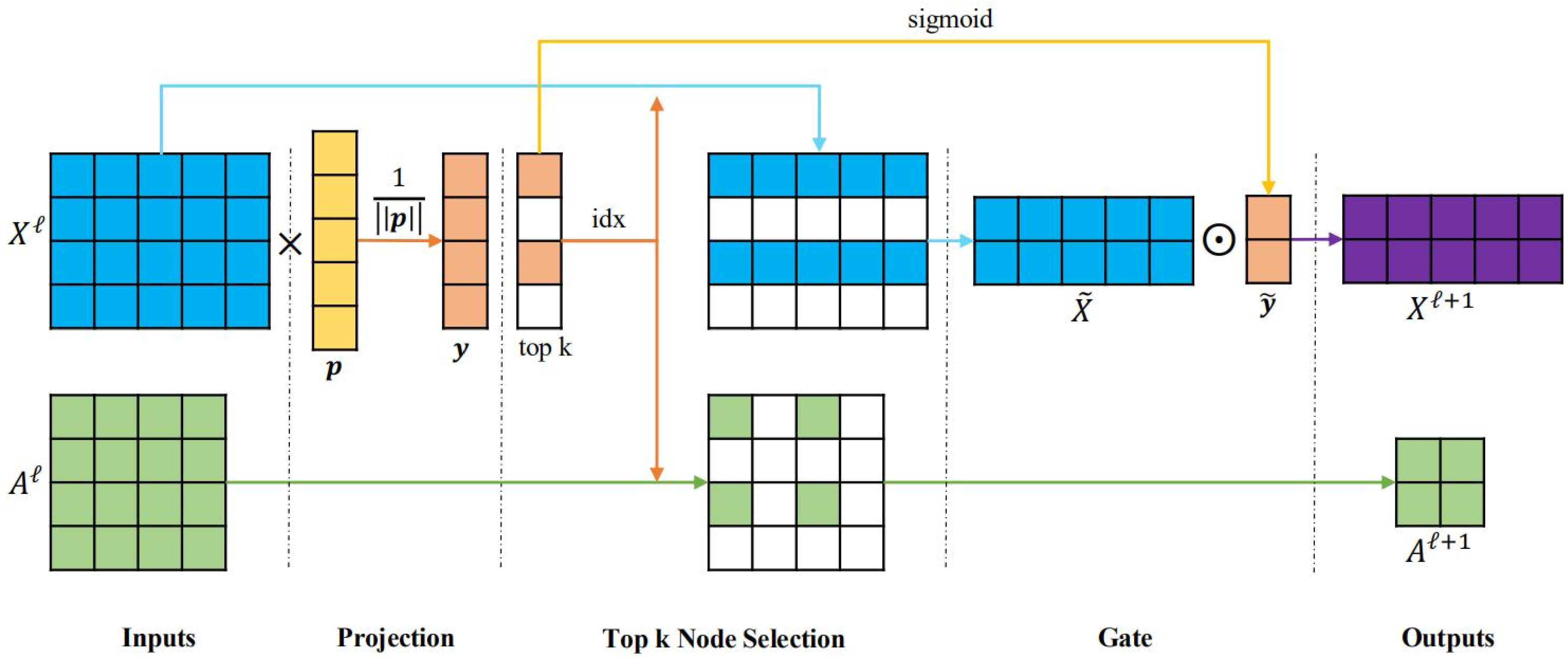
$$X^{\ell+1} = \text{distribute}(0_{N\times C}, X^{\ell}, \text{idx}), \qquad (3)$$

0N×C are the initially empty feature matrix for the new graph. In X`+1, row vectors with indices in idx are updated by row vectors in X`, while other row vectors remain zero.Up-sampling operations are important for encoder-decoder networks such as U-Net.

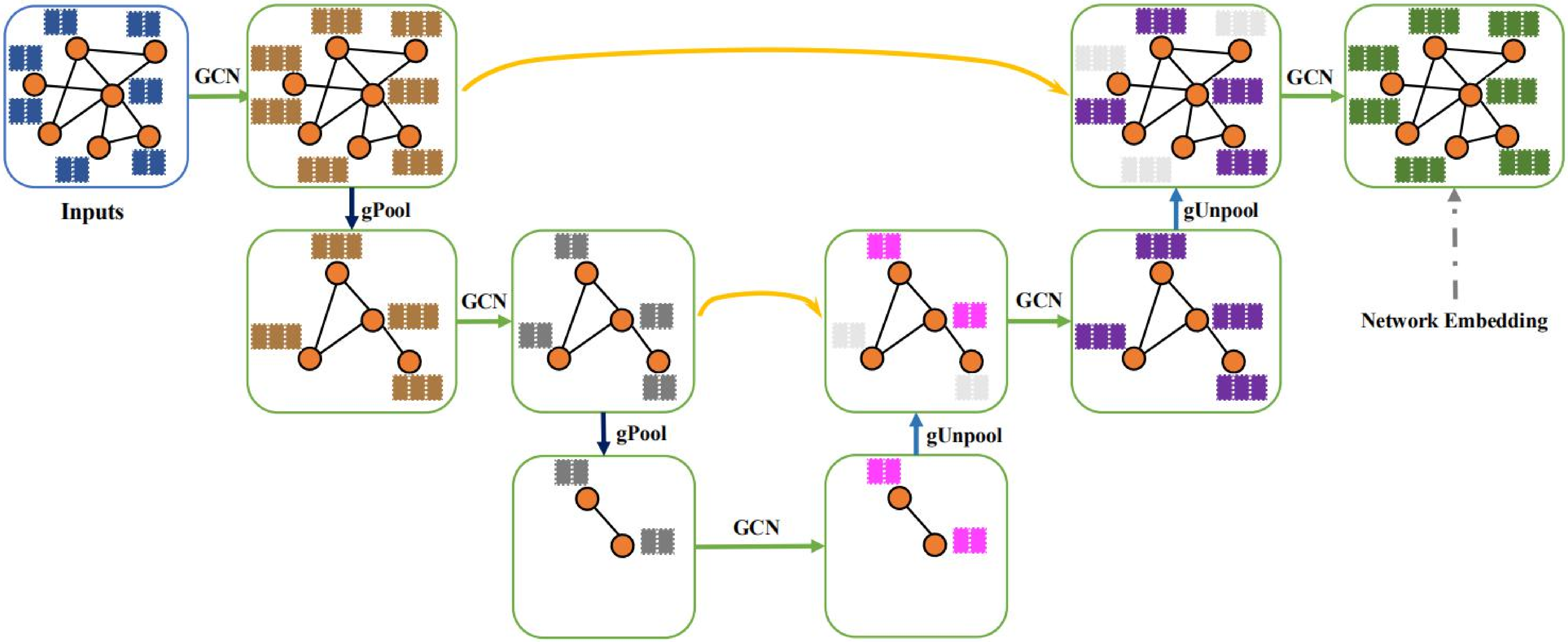
Graph Unpooling Layer



Graph U-Nets Architecture



Graph U-Nets Architecture



Trick1: Graph Connectivity Augmentation via Graph Power

we propose to use the k th graph power Gk to increase the graph connectivity. This operation builds links between nodes whose distances are at most k hops (Chepuri & Leus, 2016).

$$A^2 = A^\ell A^\ell, \qquad A^{\ell+1} = A^2(\mathrm{idx}, \mathrm{idx}),$$

where A2 ∈ RN×N is the 2nd graph power.

Trick2: Improved GCN Layer

$$\hat{A} = \breve{A} + I$$

$$\hat{A} = \breve{A} + 2I$$

imposing larger weights on self loops in the graph

Experimental

Table 1. Summary of datasets used in our node classification experiments (Yang et al., 2016; Zitnik & Leskovec, 2017). The Cora, Citeseer, and Pubmed datasets are used for transductive learning experiments.

Dataset	**Nodes**	**Features**	**Classes**	**Training**	**Validation**	**Testing**	**Degree**
Cora	2708	1433	7	140	500	1000	4
Citeseer	3327	3703	6	120	500	1000	5
Pubmed	19717	500	3	60	500	1000	6

Table 2. Summary of datasets used in our inductive learning experiments. The D&D (Dobson & Doig, 2003), PROTEINS (Borgwardt et al., 2005), and COLLAB (Yanardag & Vishwanathan, 2015) datasets are used for inductive learning experiments.

Dataset	**Graphs**	**Nodes (max)**	**Nodes (avg)**	**Classes**
D&D	1178	5748	284.32	2
PROTEINS	1113	620	39.06	2
COLLAB	5000	492	74.49	3

Experimental

Table 3. Results of transductive learning experiments in terms of node classification accuracies on Cora, Citeseer, and Pubmed datasets. g-U-Nets denotes our proposed graph U-Nets model.

Models	**Cora**	**Citeseer**	**Pubmed**
DeepWalk (Perozzi et al., 2014)	67.2%	43.2%	65.3%
Planetoid (Yang et al., 2016)	75.7%	64.7%	77.2%
Chebyshev (Defferrard et al., 2016)	81.2%	69.8%	74.4%
GCN (Kipf & Welling, 2017)	81.5%	70.3%	79.0%
GAT (Veličković et al., 2017)	$83.0 \pm 0.7\%$	$72.5 \pm 0.7\%$	$79.0 \pm 0.3\%$
g-U-Nets (Ours)	$\mathbf{84.4 \pm 0.6\%}$	$\mathbf{73.2 \pm 0.5\%}$	$\mathbf{79.6 \pm 0.2\%}$

Table 4. Results of inductive learning experiments in terms of graph classification accuracies on D&D, PROTEINS, and COLLAB datasets. g-U-Nets denotes our proposed graph U-Nets model.

Models	**D&D**	**PROTEINS**	**COLLAB**
PSCN (Niepert et al., 2016)	76.27%	75.00%	72.60%
DGCNN (Zhang et al., 2018)	79.37%	76.26%	73.76%
DiffPool-DET (Ying et al., 2018)	75.47%	75.62%	**82.13%**
DiffPool-NOLP (Ying et al., 2018)	79.98%	76.22%	75.58%
DiffPool (Ying et al., 2018)	80.64%	76.25%	75.48%
g-U-Nets (Ours)	**82.43%**	**77.68%**	77.56%

Thanks!