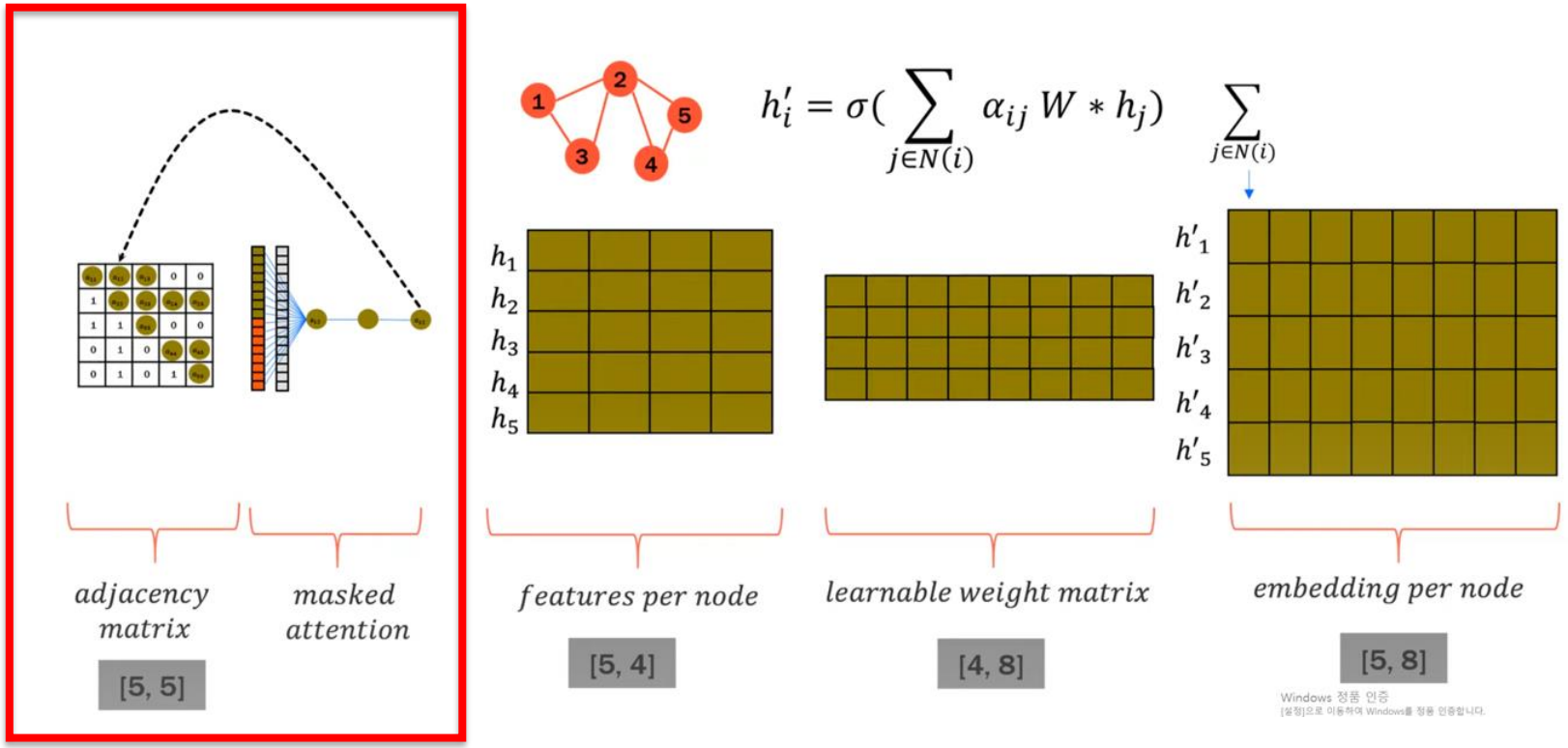
Graph Attention Network

엔자이너 연구실

How to implement GNNs in my study



Graph Attention Network

- ICLR 2018

GRAPH ATTENTION NETWORKS

Petar Veličković*
Department of Computer Science and Technology
University of Cambridge
petar.velickovic@cst.cam.ac.uk

Guillem Cucurull*
Centre de Visió per Computador, UAB
gcucurull@gmail.com

Arantxa Casanova*
Centre de Visió per Computador, UAB
ar.casanova.8@gmail.com

Adriana Romero
Montréal Institute for Learning Algorithms
adriana.romero.soriano@umontreal.ca

Pietro Liò
Department of Computer Science and Technology
University of Cambridge
pietro.lio@cst.cam.ac.uk

Yoshua Bengio
Montréal Institute for Learning Algorithms
yoshua.umontreal@gmail.com

ABSTRACT

We present graph attention networks (GATs), novel neural network architectures that operate on graph-structured data, leveraging masked self-attentional layers to address the shortcomings of prior methods based on graph convolutions or their approximations. By stacking layers in which nodes are able to attend over their neighborhoods' features, we enable (implicitly) specifying different weights to different nodes in a neighborhood, without requiring any kind of costly matrix operation (such as inversion) or depending on knowing the graph structure upfront. In this way, we address several key challenges of spectral-based graph neural networks simultaneously, and make our model readily applicable to inductive as well as transductive problems. Our GAT models have achieved or matched state-of-the-art results across four established transductive and inductive graph benchmarks: the *Cora*, *Citeseer* and *Pubmed* citation network datasets, as well as a *protein-protein interaction* dataset (wherein test graphs remain unseen during training).

GNN :: GCN

- The way GCN aggregates is structure-dependent, which can hurt its generalizability.
- However, GAT proposes a different type of aggregation

$$c_{ij} = \sqrt{|\mathcal{N}(i)|}\sqrt{|\mathcal{N}(j)|}$$

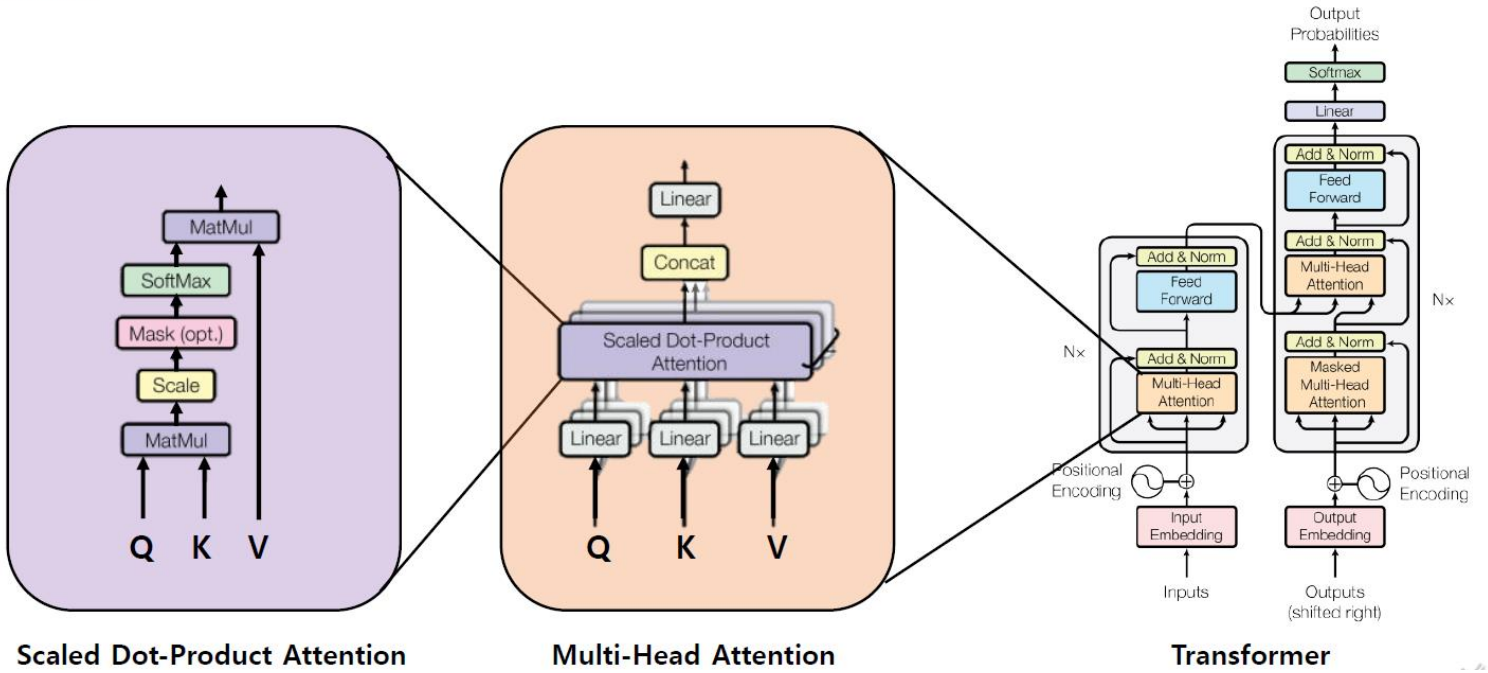
$$h_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i)} \frac{1}{c_{ij}} W^{(l)} h_j^{(l)}\right)$$

Graph Convolutional Network

Graph Attention Network

Transformer

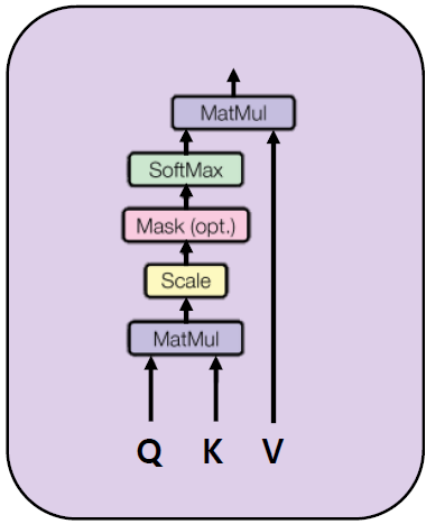
- Transformer

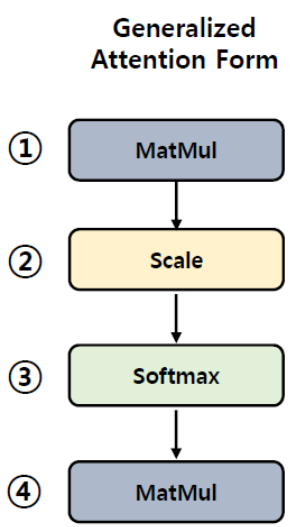


Scaled Dot-Product Attention | Multi-Head Attention | Transformer

Graph Attention Network

- Transformer





$$A(q,K,V) = \sum_{i} softmax(f(K,q))\,V$$

$$f(K,Q) = QK^{T} \quad (K = KW^{K}, Q = QW^{Q}, V = QW^{V})$$

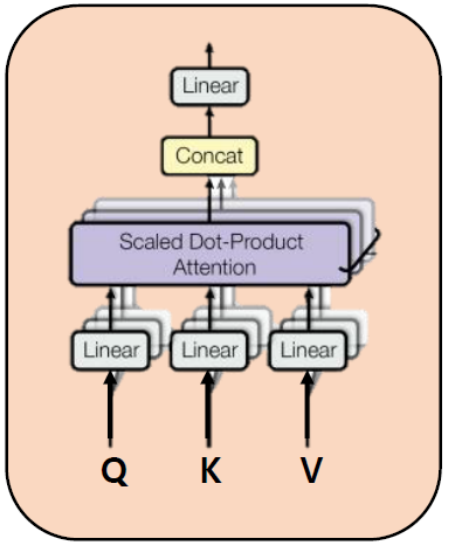
$$\frac{QK^{T}}{\sqrt{d_k}}$$

$$softmax(\frac{QK^{T}}{\sqrt{d_k}})$$

$$softmax(\frac{QK^{T}}{\sqrt{d_k}})V$$

Graph Attention Network

- Transformer



Self-Attention $SA(q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$

① **Linear** $Q` = QW_i^Q \quad K` = KW_i^K \quad V` = VW_i^V \quad (i = 1 \dots h)$

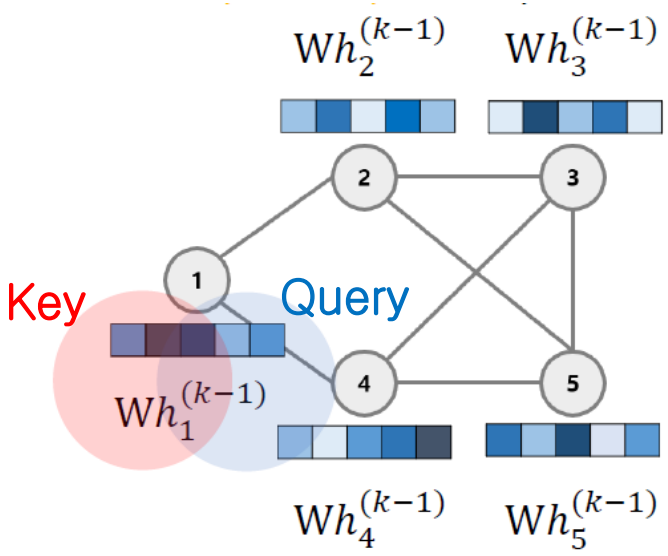
② **Self-Atten** $\text{head}_i = SA(Q`, K`, V`)$

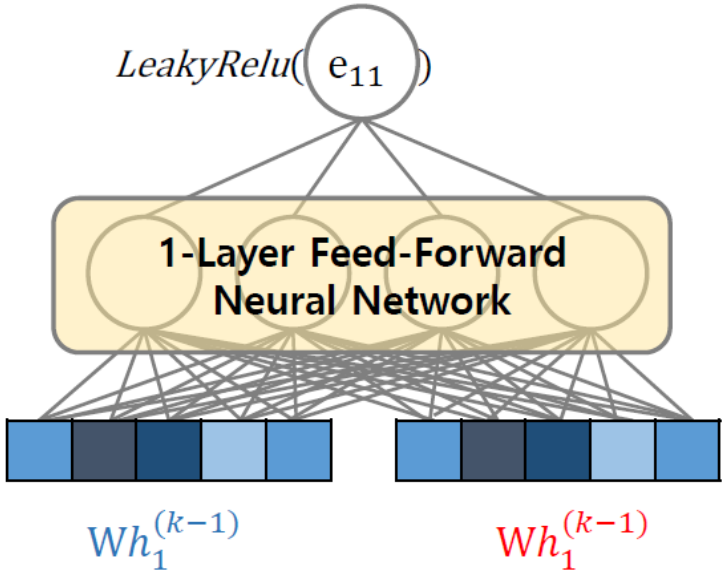
③ **Concat** $[\text{head}_1, \text{head}_2, \dots, \text{head}_h]$

④ **Linear** $[\text{head}_1, \text{head}_2, \dots, \text{head}_h]W^O$
$= \boldsymbol{MultiHead(Q, K, V)}$

Graph Attention Network

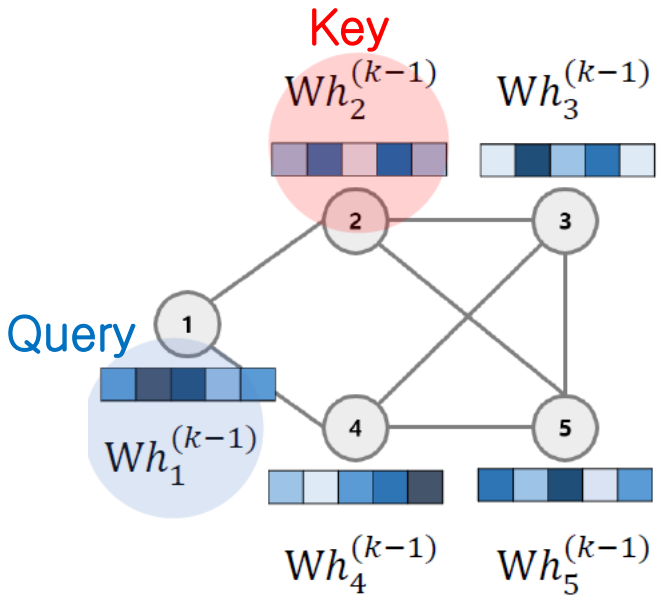
- Graph Attention Network

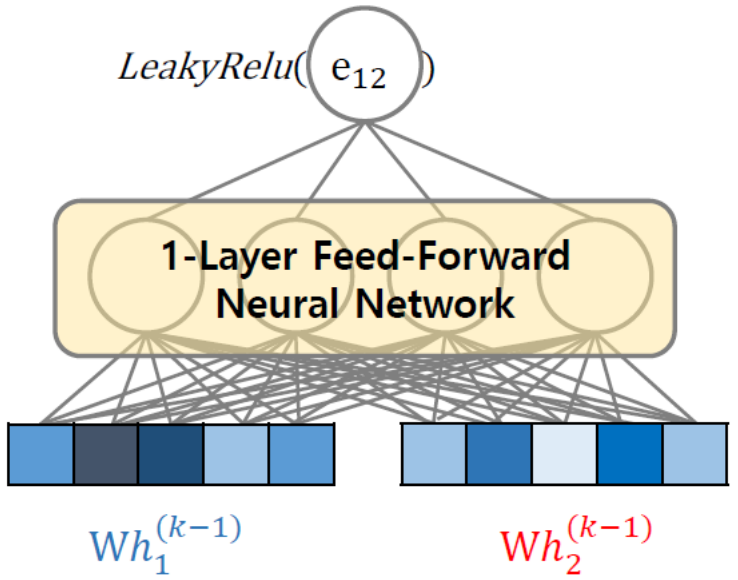




Graph Attention Network

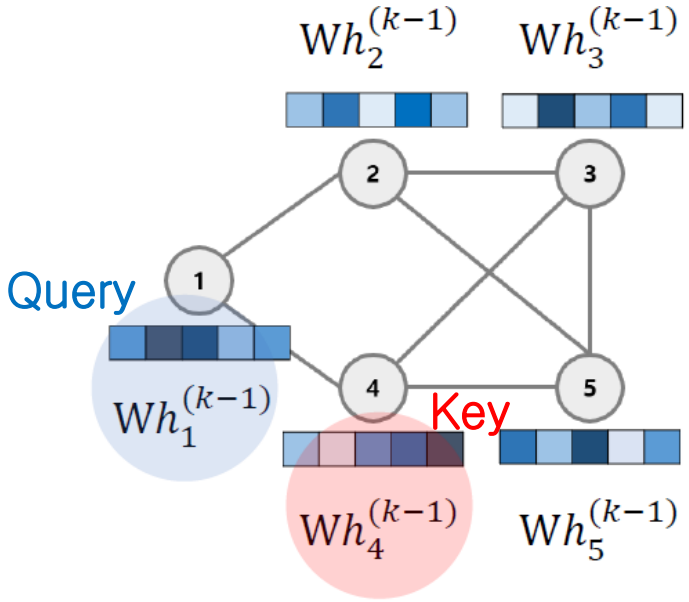
- Graph Attention Network

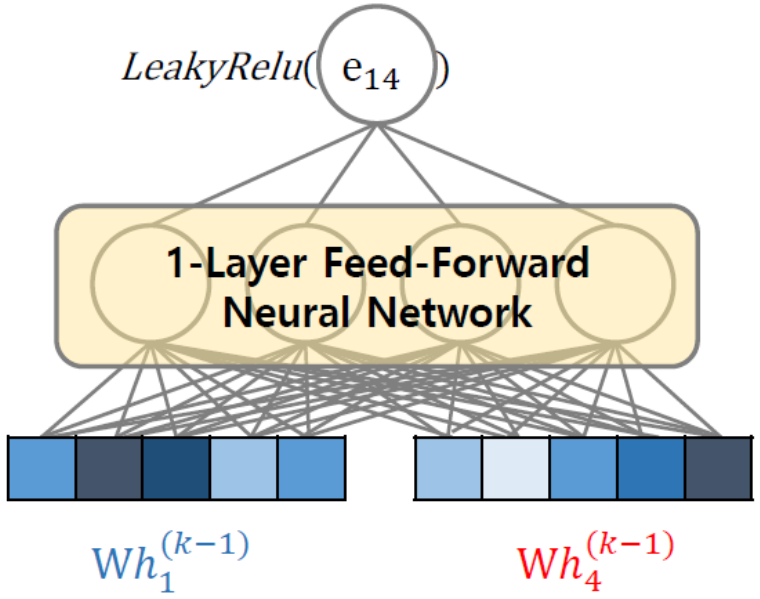




Graph Attention Network

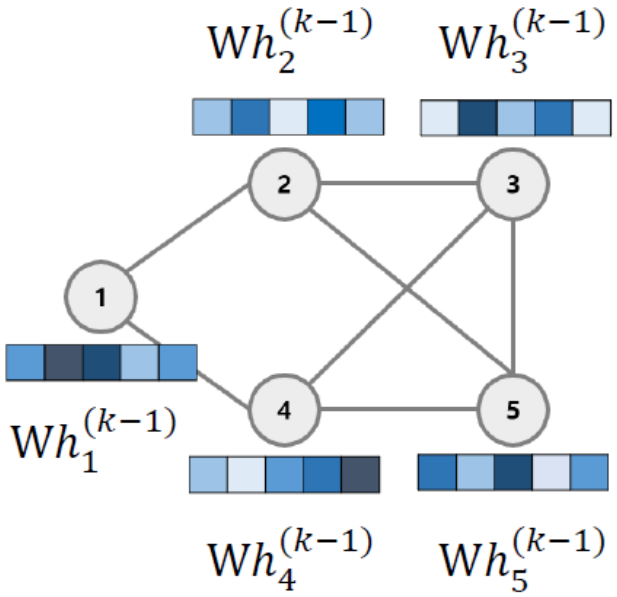
- Graph Attention Network





Graph Attention Network

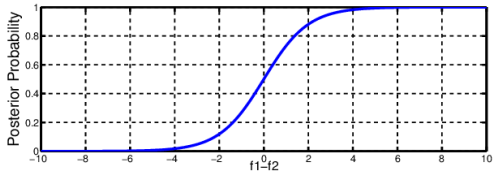
- Graph Attention Network



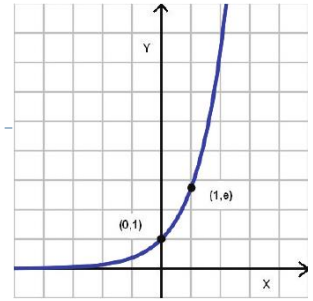
e_{11}	e_{12}		e_{14}	
e_{21}	e_{22}	e_{23}		e_{25}
	e_{32}	e_{33}	e_{34}	e_{35}
e_{41}		e_{43}	e_{44}	e_{45}
	e_{25}	e_{53}	e_{54}	e_{55}

Graph Attention Network

- Graph Attention Network

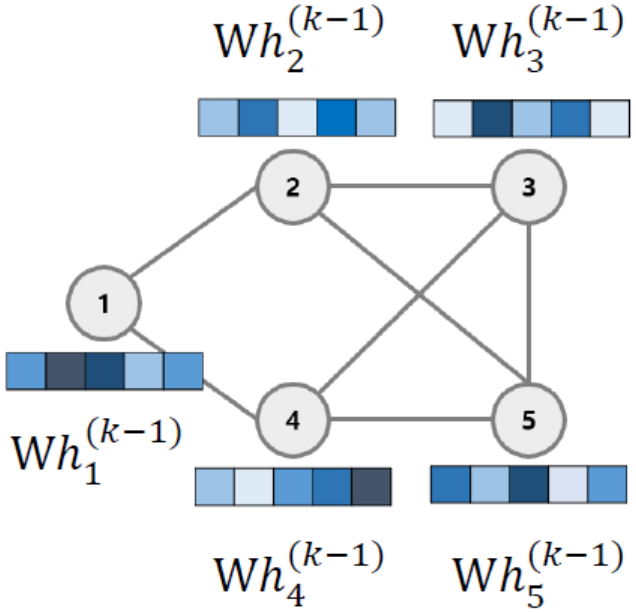


Softmax



e exponential

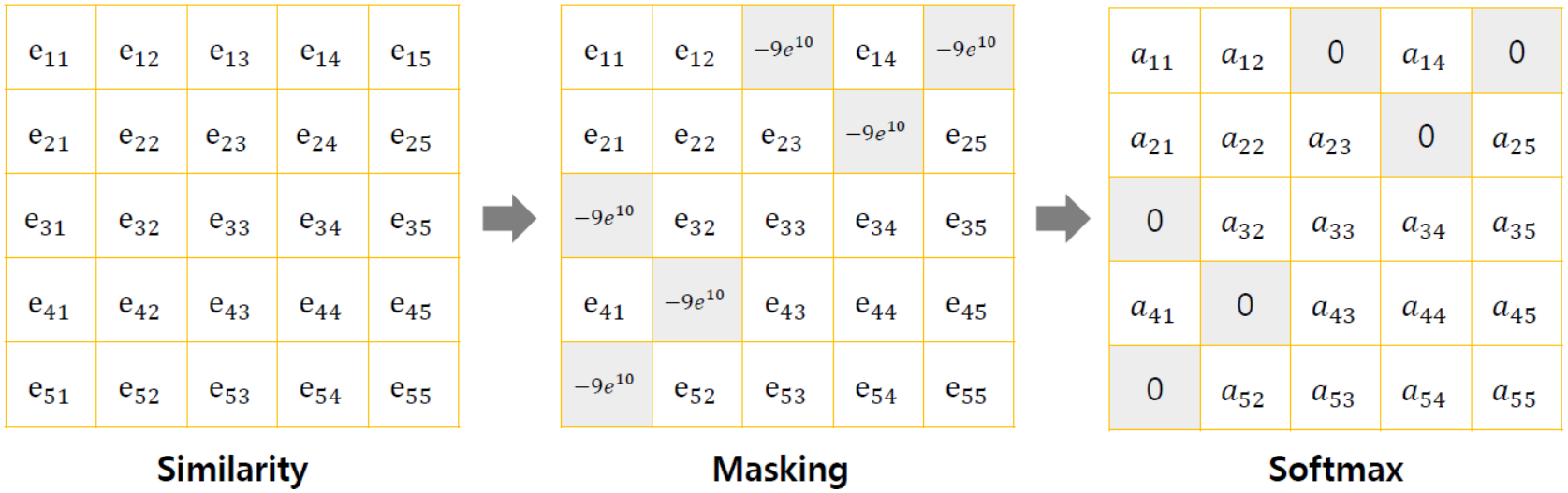
$$\sigma(\vec{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$



e_{11}	e_{12}	Softmax	e_{14}	
e_{21}	e_{22}	Softmax		e_{25}
	e_{32}	Softmax	e_{34}	e_{35}
e_{41}		Softmax	e_{44}	e_{45}
	e_{25}	Softmax	e_{54}	e_{55}

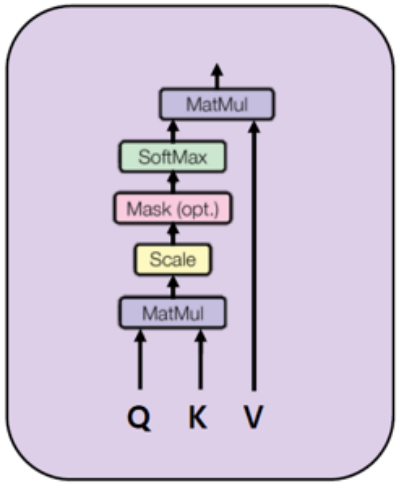
Graph Attention Network

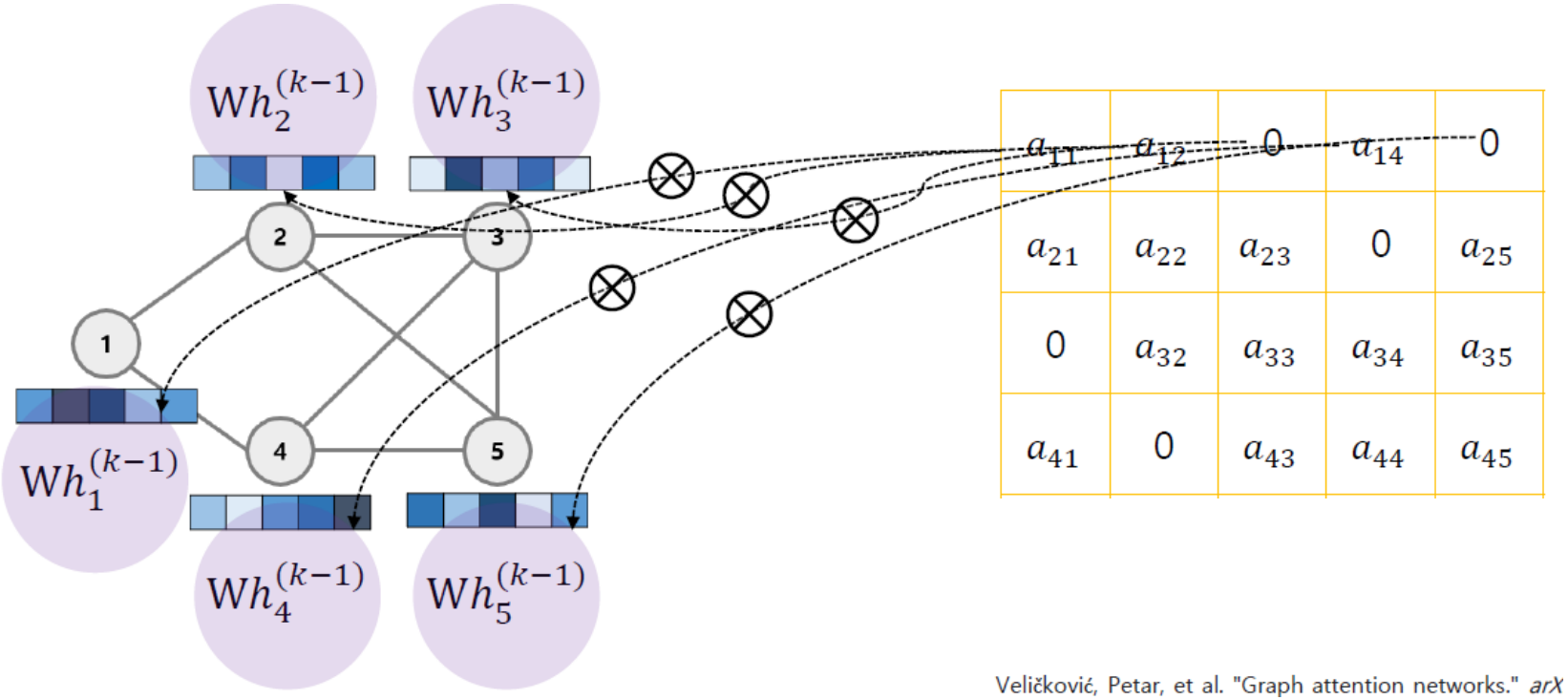
- Graph Attention Network



Graph Attention Network

- Graph Attention Network

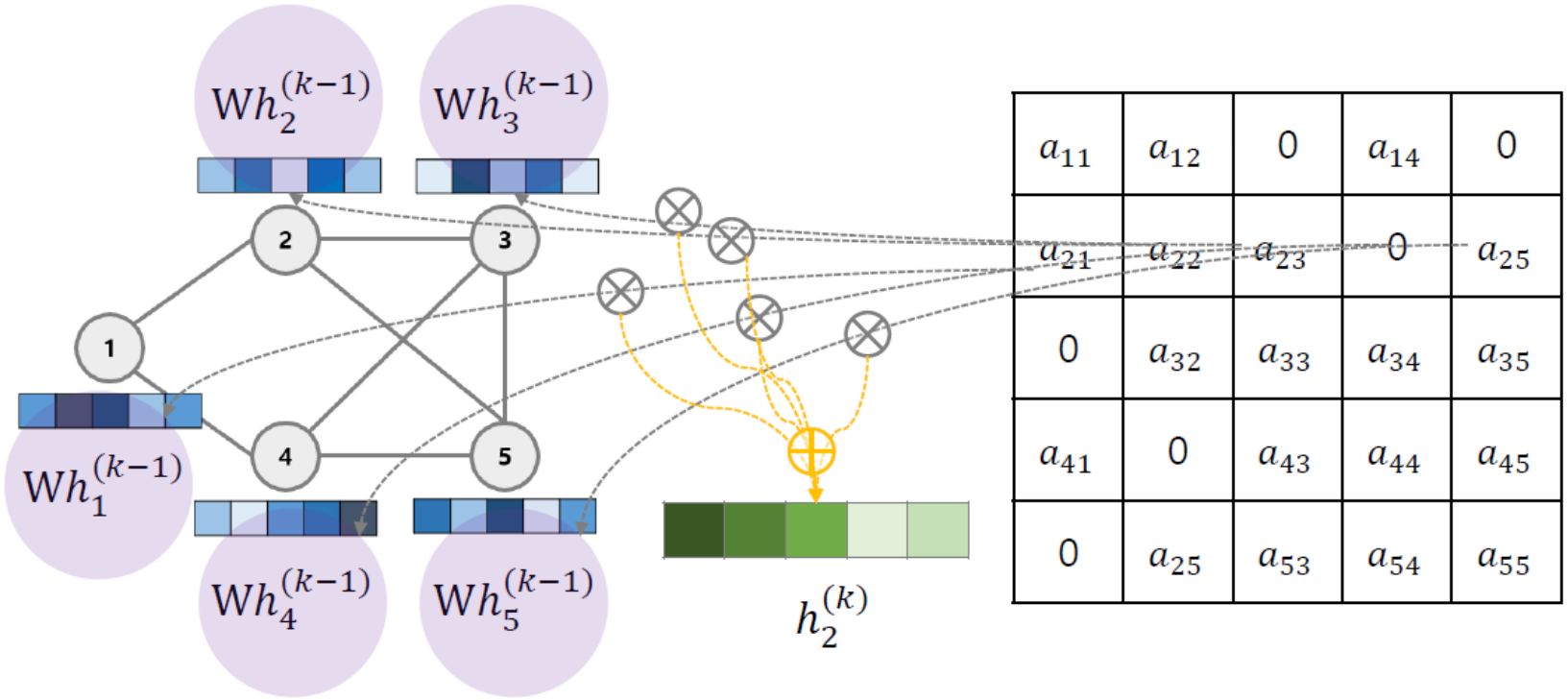




Veličković, Petar, et al. "Graph attention networks." *arX*

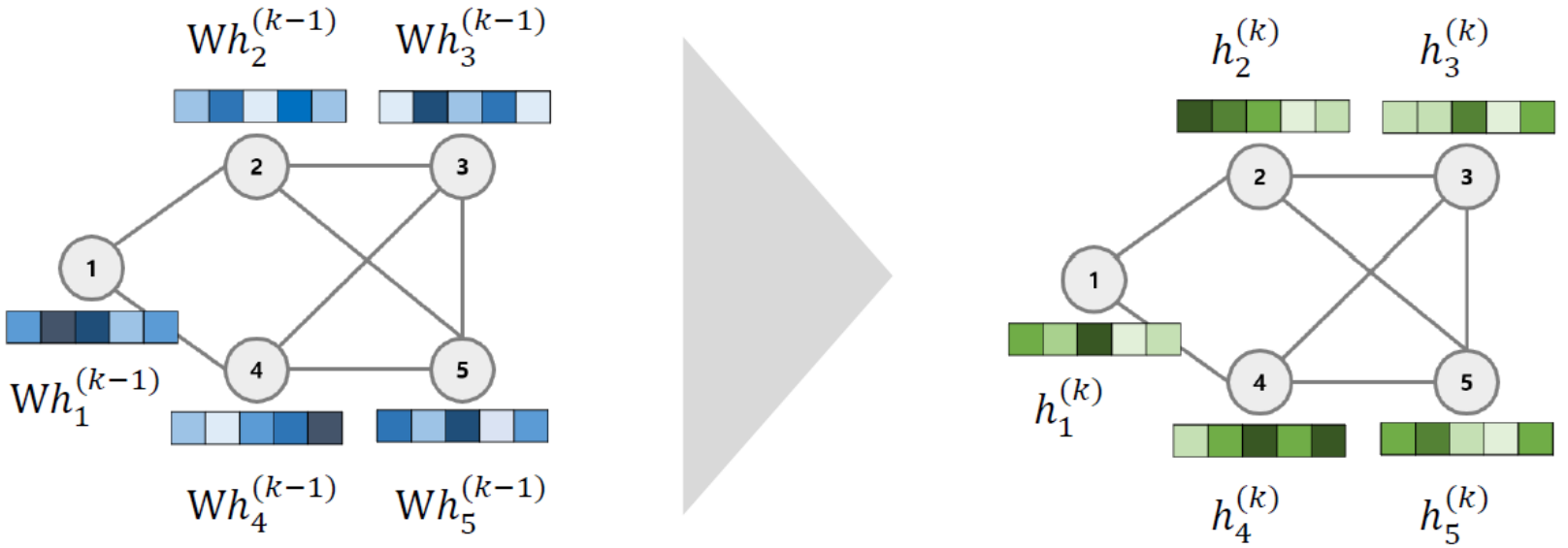
Graph Attention Network

- Graph Attention Network



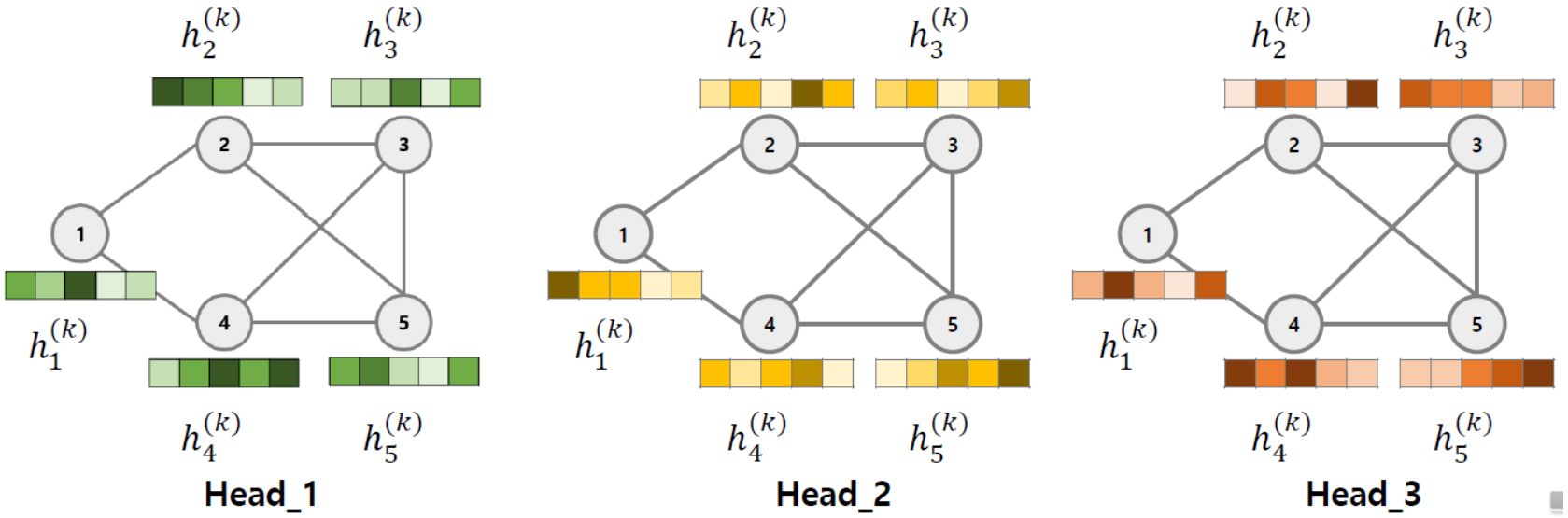
Graph Attention Network

- Graph Attention Network



Graph Attention Network

- Graph Attention Network



Graph Attention Network

- Graph Attention Network

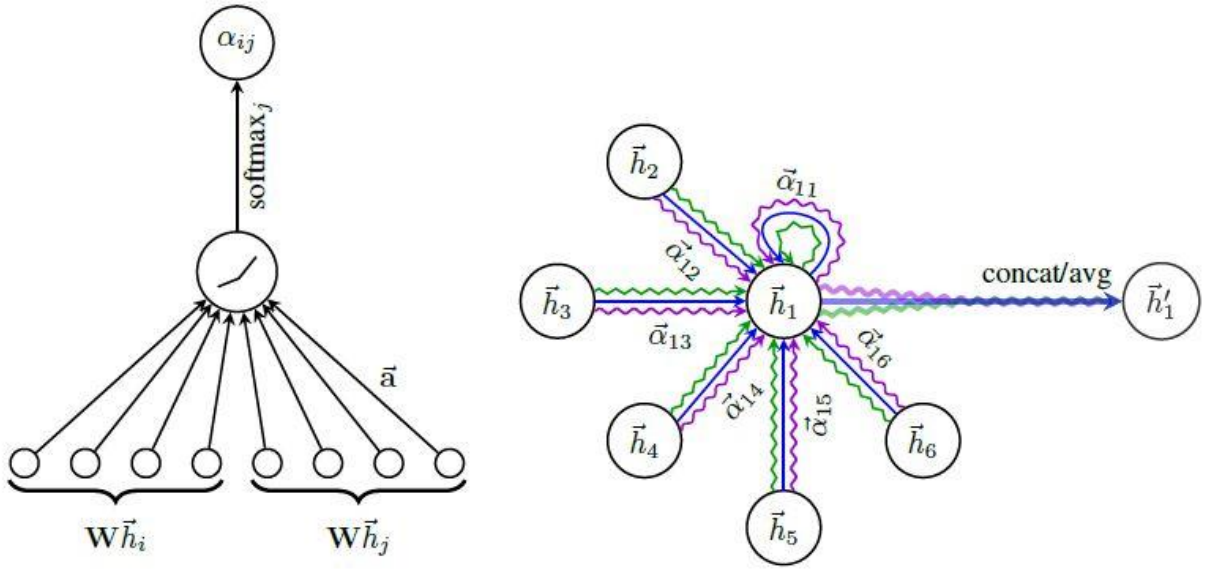


Figure 1: **Left:** The attention mechanism $a(\mathbf{W}\vec{h}_i, \mathbf{W}\vec{h}_j)$ employed by our model, parametrized by a weight vector $\vec{a} \in \mathbb{R}^{2F'}$, applying a LeakyReLU activation. **Right:** An illustration of multi-head attention (with $K = 3$ heads) by node 1 on its neighborhood. Different arrow styles and colors denote independent attention computations. The aggregated features from each head are concatenated or averaged to obtain $\vec{h}'_1$.

Graph Attention Network

- Graph Attention Network

$$z_i^{(l)} = W^{(l)} h_i^{(l)}, \tag{1}$$

$$e_{ij}^{(l)} = \text{LeakyReLU}(\vec{a}^{(l)^T}(z_i^{(l)} || z_j^{(l)})), \tag{2}$$

$$\alpha_{ij}^{(l)} = \frac{\exp(e_{ij}^{(l)})}{\sum_{k \in \mathcal{N}(i)} \exp(e_{ik}^{(l)})}, \tag{3}$$

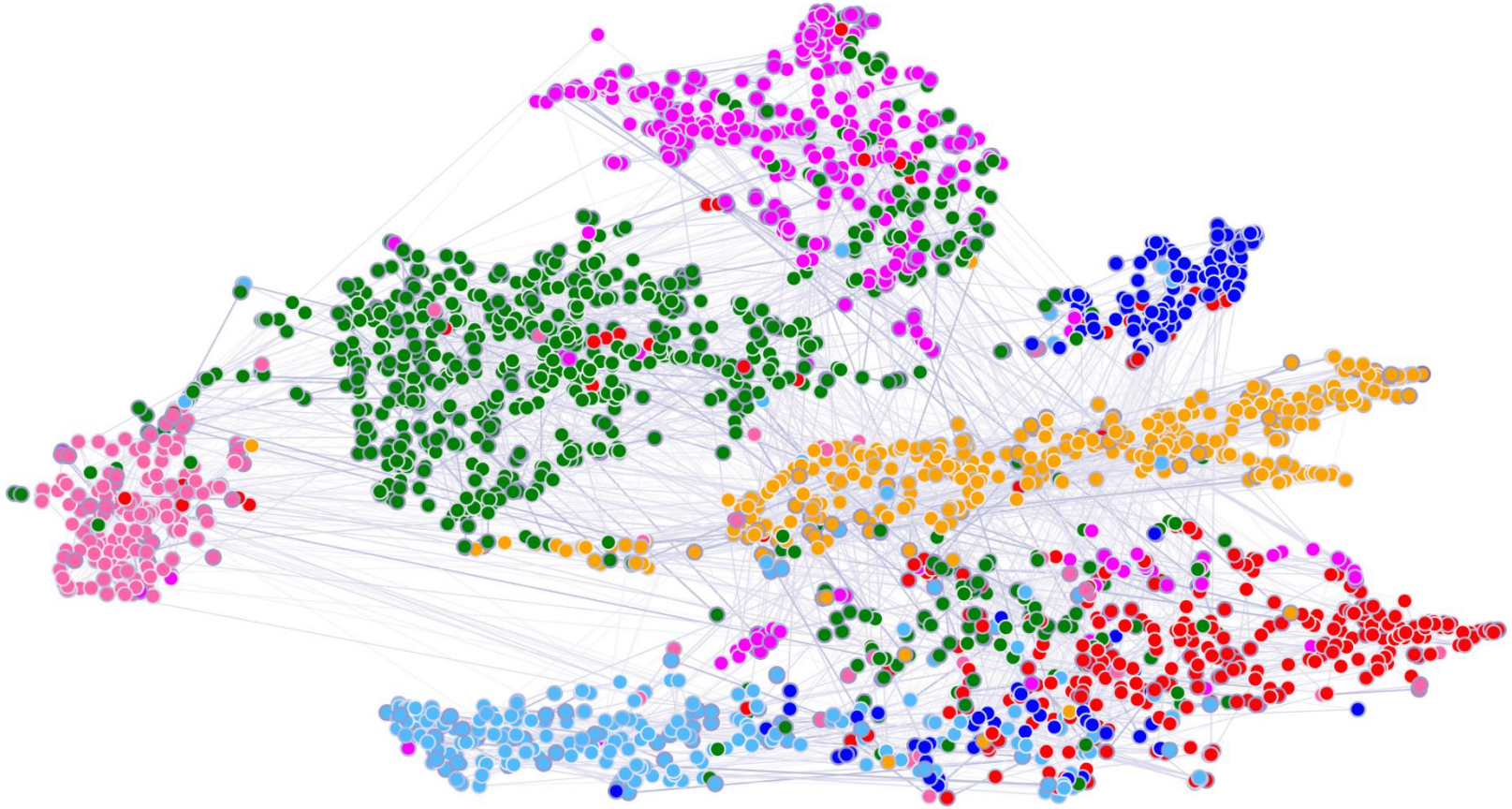
$$h_i^{(l+1)} = \sigma\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{(l)} z_j^{(l)}\right), \tag{4}$$

$$\text{concatenation} : h_i^{(l+1)} = ||_{k=1}^{K} \sigma\left(\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{k} W^k h_j^{(l)}\right)$$

$$\text{average} : h_i^{(l+1)} = \sigma\left(\frac{1}{K}\sum_{k=1}^{K}\sum_{j \in \mathcal{N}(i)} \alpha_{ij}^{k} W^k h_j^{(l)}\right)$$

Graph Attention Network

- Node Classification



Graph Attention Network

- GAT -> Cluster-GCN -> GCN2