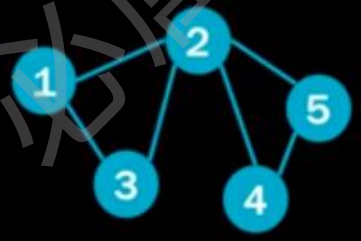
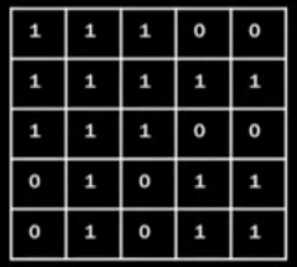
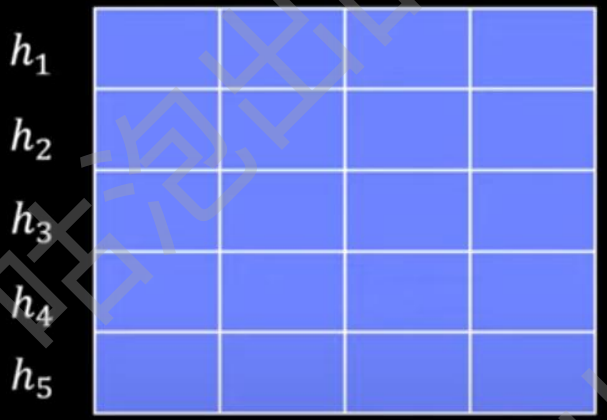
Graph Attention Network

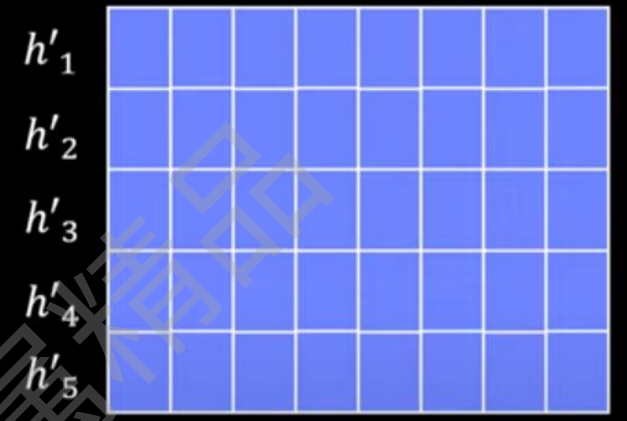
回顾:GNN基本计算方法



$$h_i' = \sigma\Big(\sum_{j \in N(i)} \underbrace{W * h_j}_{h_j^*}\Big)$$







adjacency matrix [5, 5]

features per node [5, 4]

learnable weight matrix [4, 8]

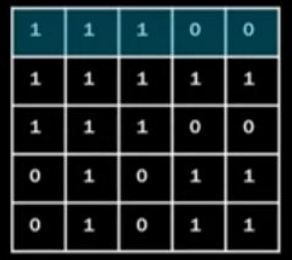
embedding per node [5, 8]

Graph Attention Network

输出得到每一节点特征

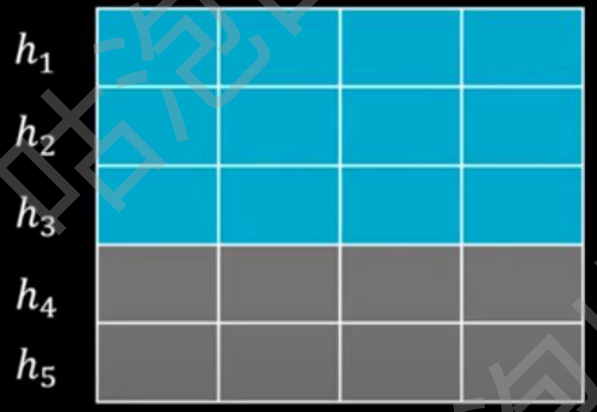
$$h_i' = \sigma\left(\sum_{j \in N(i)} W * h_j\right)$$

h_j^*



adjacency matrix

[5, 5]



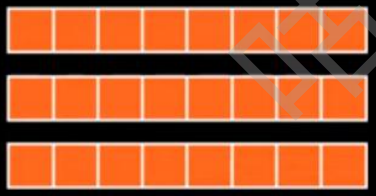
features per node

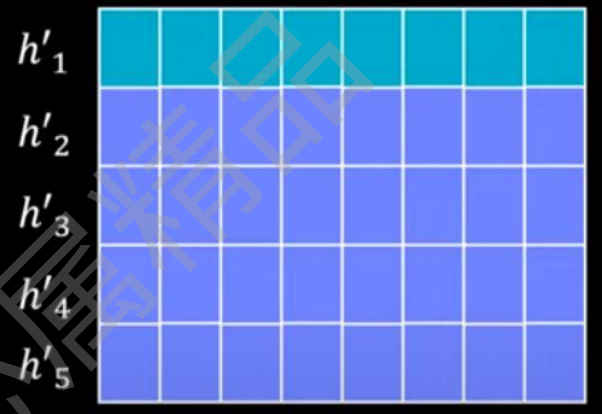
[5, 4]

learnable weight matrix

[4, 8]

$\sum_{j \in N(i)}$



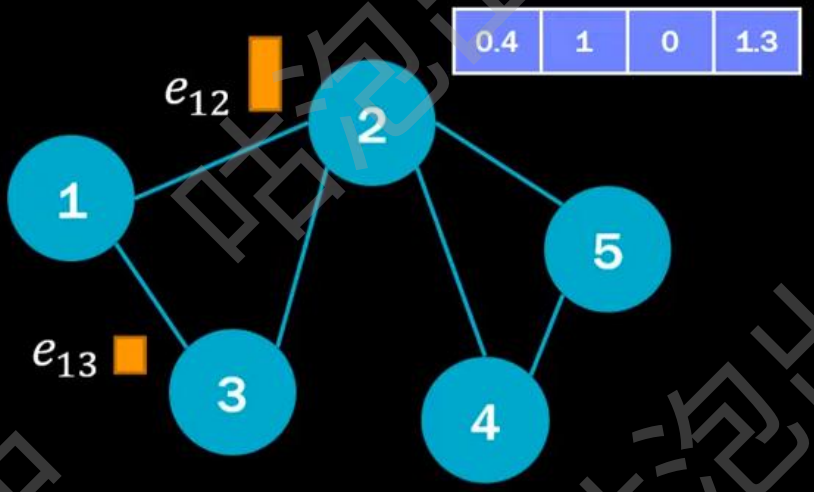


embedding per node

[5, 8]

Graph Attention Network

图中的Attention



$$e_{ij} = a(Wh_i, Wh_j)$$

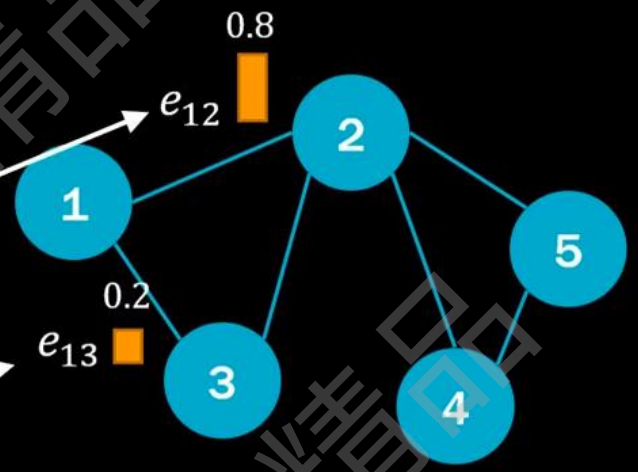
$attention\ coefficient$

Graph Attention Network

Attention的计算

$$e_{ij} = a(Wh_i, Wh_j)$$

attention coefficient

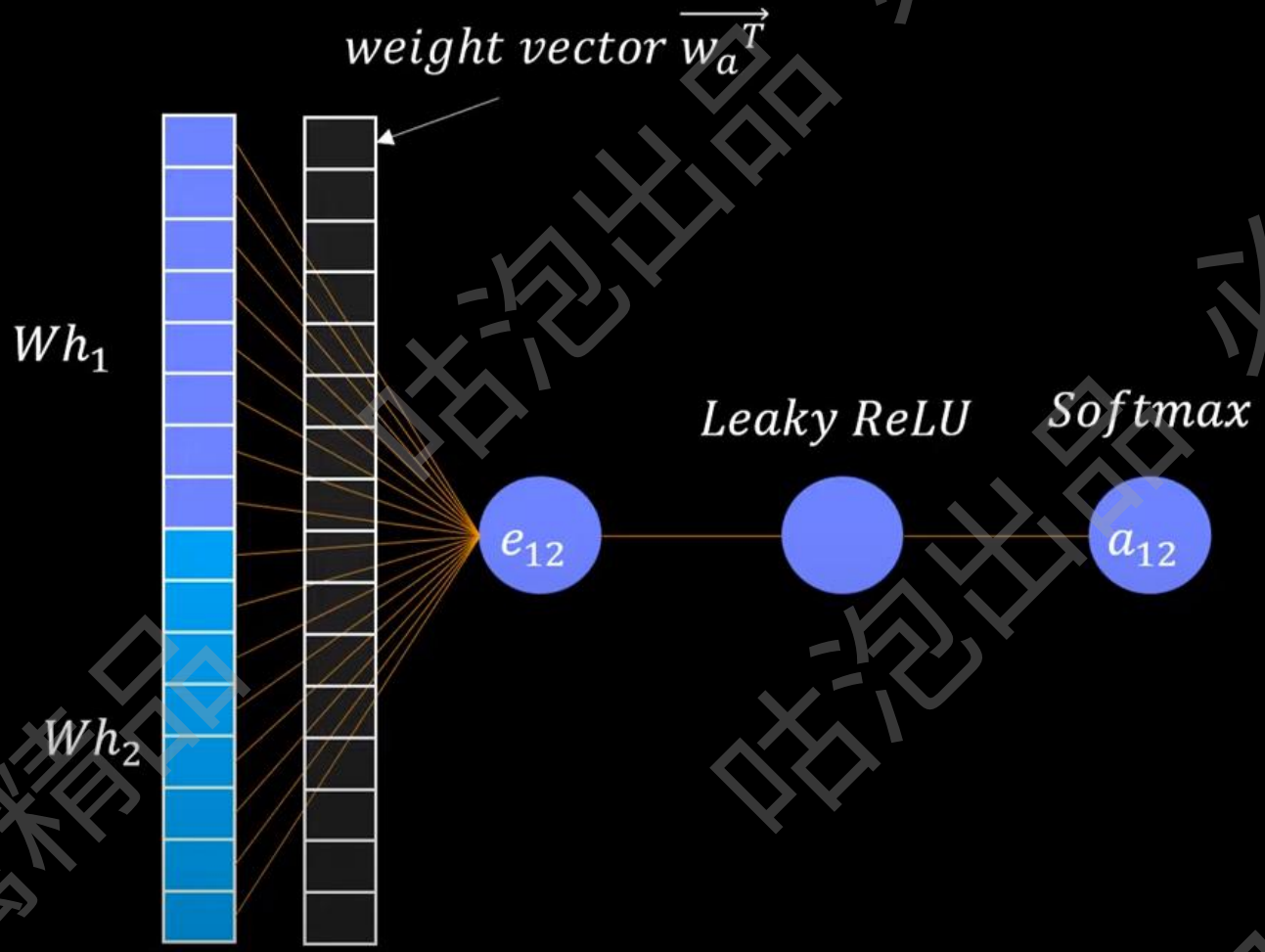


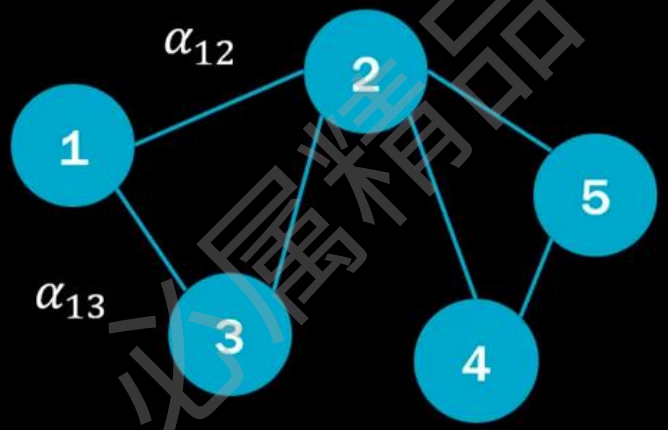
$$\alpha_{ij} = softmax_j(e_{ij}) = \frac{\exp(e_{ij})}{\sum_{k \epsilon N(i)} \exp(e_{ik})} = \frac{\exp(a(Wh_i, Wh_j))}{\sum_{k \epsilon N(i)} \exp(a(Wh_i, Wh_j))}$$

normalized attention coefficient

Graph Attention Network

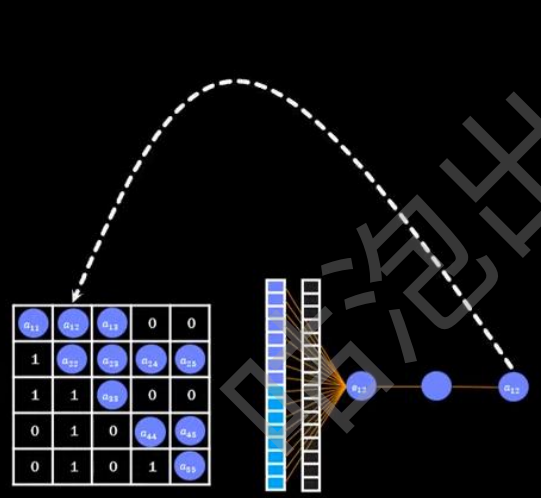
计算方法其实很简单

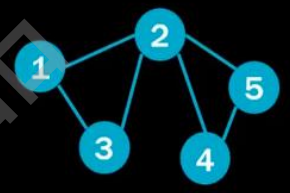




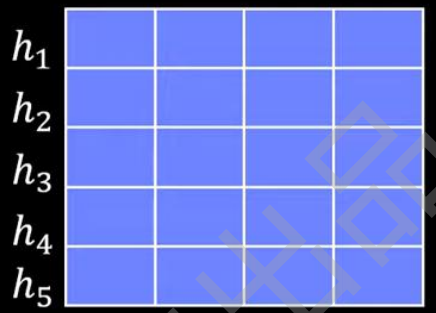
Graph Attention Network

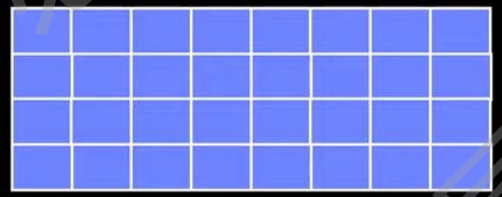
其实就是对邻接矩阵进行了加权



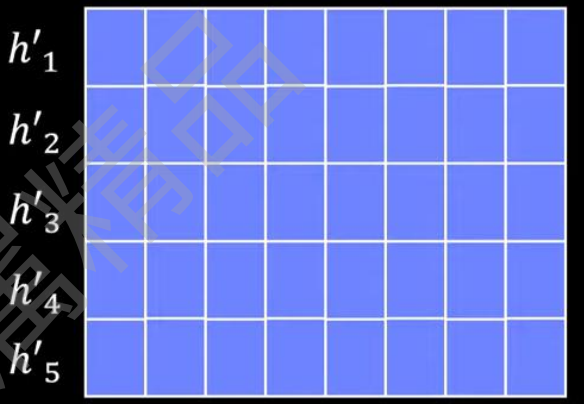


$$h_i' = \sigma\left(\sum_{j \in N(i)} \alpha_{ij} W * h_j\right)$$





$\sum_{j \in N(i)}$



adjacency matrix

[5, 5]

masked attention

features per node

[5, 4]

learnable weight matrix

[4, 8]

embedding per node

[5, 8]

Graph Attention Network

计算流程