

# 图神经网络：GCN 基础与典型应用

2025 年 10 月 22 日

## 目录

### 1 图卷积网络（GCN）

图卷积网络将卷积思想推广到非欧氏结构的图数据。在无向图  $G = (\mathcal{V}, \mathcal{E})$  中，设邻接矩阵为  $\mathbf{A}$ ，节点特征矩阵为  $\mathbf{X} \in \mathbb{R}^{|\mathcal{V}| \times d}$ 。谱图理论为卷积定义提供了形式化框架。图 ?? 概述了邻居聚合的流程。

#### 1.1 谱域定义

组合拉普拉斯矩阵  $\mathbf{L} = \mathbf{D} - \mathbf{A}$ （ $\mathbf{D}$  为度矩阵）可分解为  $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^\top$ 。谱滤波  $g_\theta(\mathbf{\Lambda})$  作用于信号  $\mathbf{X}$  的结果为

$$g_\theta \star \mathbf{X} = \mathbf{U} g_\theta(\mathbf{\Lambda}) \mathbf{U}^\top \mathbf{X}. \quad (1)$$

利用切比雪夫多项式近似  $g_\theta$ ，得到  $K$  跳邻域支持：

$$g_\theta \star \mathbf{X} \approx \sum_{k=0}^K \theta_k T_k(\tilde{\mathbf{L}}) \mathbf{X}, \quad \tilde{\mathbf{L}} = \frac{2}{\lambda_{\max}} \mathbf{L} - \mathbf{I}. \quad (2)$$

Kipf & Welling 将其简化为一阶形式（ $K = 1$ ），并通过自环重归一化：

$$\mathbf{H}^{(l+1)} = \sigma \left( \tilde{\mathbf{D}}^{-1/2} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-1/2} \mathbf{H}^{(l)} \mathbf{W}^{(l)} \right), \quad (3)$$

其中  $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$ ， $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_{ij}$ 。对称归一化保证了数值稳定性，并包含自身节点信息。

#### 1.2 信息传递视角

从消息传递角度看，GCN 层通过邻居特征平均更新节点：

$$\mathbf{h}_v^{(l+1)} = \sigma \left( \sum_{u \in \mathcal{N}(v) \cup \{v\}} \frac{1}{\sqrt{\tilde{d}_v \tilde{d}_u}} \mathbf{h}_u^{(l)} \mathbf{W}^{(l)} \right), \quad (4)$$

其中  $\tilde{d}_v$  为加入自环后的度。图 ?? 展示了两层 GCN 的信息传播范围。

### 1.3 过平滑与缓解

随着层数增加，节点表示趋于一致，即“过平滑”。常见缓解方法：

- 引入残差/跳连（Res-GCN、JK-Net）保留低阶特征。
- 使用正则化归一（PairNorm、BatchNorm）稳定嵌入方差。
- 个性化传播（APNP）结合随机游走与抛掷概率：

$$\mathbf{Z} = (1 - \alpha)(\mathbf{I} - \alpha\tilde{\mathbf{P}})^{-1}\mathbf{H}^{(K)}, \quad \tilde{\mathbf{P}} = \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}. \quad (5)$$

### 1.4 训练流程

Listing 1: 基于 PyTorch Geometric 的半监督二维 GCN 节点分类示例。

```

1 import torch
2 import torch.nn.functional as F
3 from torch_geometric.nn import GCNConv
4
5 class GCN(torch.nn.Module):
6     def __init__(self, in_dim, hidden_dim, out_dim, dropout=0.5):
7         super().__init__()
8         self.conv1 = GCNConv(in_dim, hidden_dim, normalize=True)
9         self.conv2 = GCNConv(hidden_dim, out_dim, normalize=True)
10        self.dropout = dropout
11
12    def forward(self, x, edge_index):
13        x = self.conv1(x, edge_index)
14        x = F.relu(x)
15        x = F.dropout(x, p=self.dropout, training=self.training)
16        x = self.conv2(x, edge_index)
17        return F.log_softmax(x, dim=1)
18
19 model = GCN(dataset.num_node_features, 64, dataset.num_classes)
20 optimizer = torch.optim.Adam(model.parameters(), lr=0.01, weight_decay
    =5e-4)
21
22 for epoch in range(200):
23     model.train()
24     optimizer.zero_grad()
25     out = model(data.x, data.edge_index)
26     loss = F.nll_loss(out[data.train_mask], data.y[data.train_mask])
27     loss.backward()
28     optimizer.step()

```

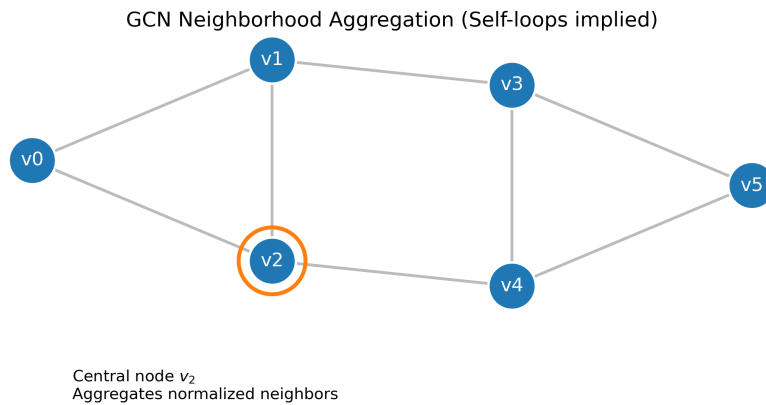


图 1: 图卷积使用归一化邻域聚合，包含自环信息以避免特征缺失。

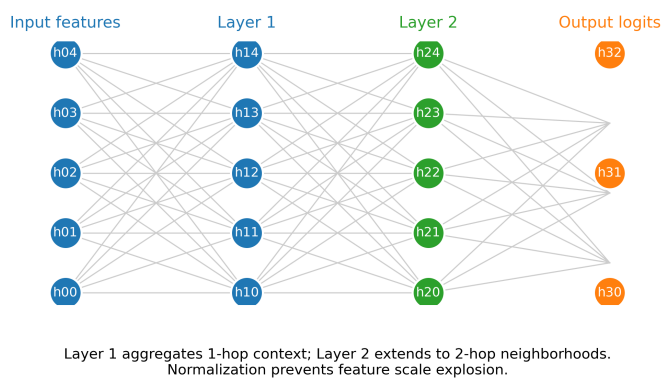


图 2: 两层 GCN 的信息范围示意：第一层覆盖一跳邻居，第二层扩展至两跳。

## 2 应用：社交网络、分子预测、推荐系统

GCN 及广义图神经网络在多种关系数据场景中表现突出。图 ?? 描绘了典型应用流程。

### 2.1 社交网络分析

社交图中，GCN 能够结合相似性与结构信息学习用户嵌入，用于社区发现、影响力预测、内容推荐。对于包含多类型节点/边的异构图，可使用关系图卷积网络（R-GCN）：

$$\mathbf{h}_v^{(l+1)} = \sigma \left( \sum_{r \in \mathcal{R}} \sum_{u \in \mathcal{N}_r(v)} \frac{1}{c_{v,r}} \mathbf{W}_r^{(l)} \mathbf{h}_u^{(l)} + \mathbf{W}_0^{(l)} \mathbf{h}_v^{(l)} \right). \quad (6)$$

在大规模平台上，需要 GraphSAGE、Cluster-GCN 等采样技术以控制算力开销。

## 2.2 分子性质预测

分子可视为节点为原子、边为化学键的图。消息传递神经网络（MPNN）在 GCN 基础上引入边特征：

$$\mathbf{m}_v^{(l+1)} = \sum_{u \in \mathcal{N}(v)} \phi^{(l)}(\mathbf{h}_v^{(l)}, \mathbf{h}_u^{(l)}, \mathbf{e}_{uv}), \quad (7)$$

$$\mathbf{h}_v^{(l+1)} = \psi^{(l)}(\mathbf{h}_v^{(l)}, \mathbf{m}_v^{(l+1)}), \quad (8)$$

其中  $\mathbf{e}_{uv}$  表示键类型/长度。全局池化（求和、平均、set2set）将节点嵌入汇总为分子指纹。结合等变网络（E(n)-GNN）与量子化特征，可在 QM9、Materials Project 等数据集取得领先结果。

## 2.3 推荐系统

用户-物品交互形成二部图。LightGCN 移除非线性与特征变换，简化传播：

$$\mathbf{E}^{(k+1)} = \tilde{\mathbf{D}}^{-1/2} \tilde{\mathbf{A}} \tilde{\mathbf{D}}^{-1/2} \mathbf{E}^{(k)}, \quad \mathbf{E} = \frac{1}{K+1} \sum_{k=0}^K \mathbf{E}^{(k)}. \quad (9)$$

最终嵌入用于打分  $s_{ui} = \mathbf{e}_u^\top \mathbf{e}_i$ 。真实业务场景还需考虑时间因素、内容特征及因果去偏。

## 2.4 部署与工程实践

- **可扩展性：** PinSAGE、GraphSAINT 利用随机游走采样；图划分、分布式训练支撑十亿级边。
- **可解释性：** GNNExplainer、GraphMask 等方法突出关键子图，提高模型透明度。
- **鲁棒性：** 针对对抗攻击的防御方法包括对抗训练、随机平滑认证等。

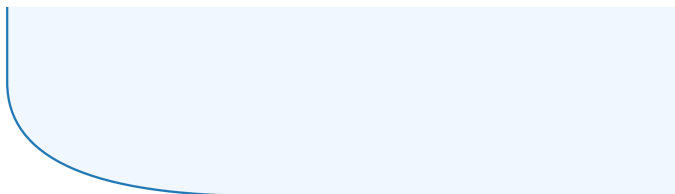


图 3: GNN 在社交网络、分子化学、推荐系统中的应用概览。

## 延伸阅读

- Thomas N. Kipf & Max Welling: 《Semi-Supervised Classification with Graph Convolutional Networks》, ICLR 2017。
- Petar Veličković 等: 《Graph Attention Networks》, ICLR 2018。
- Will Hamilton 等: 《Inductive Representation Learning on Large Graphs》, NIPS 2017。
- Keyulu Xu 等: 《How Powerful are Graph Neural Networks?》, ICLR 2019。
- He 等: 《LightGCN: Simplifying and Powering Graph Convolution Network for Recommendation》, SIGIR 2020。