# 图神经网络实验报告

本实验针对五个常用图数据集（Cora、Citeseer、Flickr、TUDataset、ZINC）对四类经典图神经网络模型（GCN、GAT、GraphSAGE、GIN）在不同超参数和训练设定下的性能表现进行了系统分析。实验主要从以下三个方面展开：（1）不同超参数及不同模型结构对模型性能的影响；（2）不同图池化方法对图分类任务性能的影响；（3）全图训练与分批次训练在模型性能及运行效率方面的对比分析。

**数据集划分：** 采用训练集（Train）、验证集（Validation）、测试集（Test）三分法

**训练配置：** 每个模型训练 50 个 epoch，记录验证集上性能最优时的指标及对应轮数

**实验环境：** PyTorch 2.4 框架，GPU 为 NVIDIA RTX 4090

## 不同超参数及不同的模型对性能的影响

在本部分实验中，选用 Cora 数据集开展节点分类任务，以评估不同学习率和网络层数对四类模型性能的影响。具体设定如下：

**模型：** GCN、GAT、GraphSAGE、GIN

**学习率：** 0.001、0.005、0.01 三种

**网络层数：** 2 层、3 层、4 层

通过上述设定，系统比较四种模型在不同超参数组合下的节点分类精度、收敛速度以及稳定性表现，以便分析超参数选择与模型结构对性能的综合影响。

结果分析与讨论

表1展示了四类经典图神经网络模型（GCN、GAT、GraphSAGE、GIN）在Cora数据集上不同网络层数（2/3/4层）及学习率（0.001、0.005、0.01）组合下的节点分类性能表现。整体来看，GCN和GAT在多数配置下表现最为优异， Acc与F1-micro普遍超过0.80，AUROC接近或高于0.96，且具有较强的稳定性。GraphSAGE整体性能略低于GCN和GAT，但指标仍保持在较高水平；而GIN模型对超参数变化尤为敏感，尤其在深层结构下（如4层、lr=0.01）性能显著退化，Acc最低仅为0.383，AUROC降至0.788，表明其在过深网络中存在明显的过平滑及梯度传播困难问题。

从网络层数的影响来看，绝大多数模型在2~3层结构下取得最佳性能。以GCN为例，3层结构结合lr=0.005时性能最优（Acc=0.827，AUROC=0.971）；GAT在3层、lr=0.001时也达到最高准确率0.83。进一步增加层数至4层未能带来显著性能提升，反而导致部分模型性能下降，尤其是GIN和GAT出现过拟合或特征过平滑的现象，验证了图神经网络在过深结构下信息聚合过度的问题。

学习率的影响则表现出明显规律。lr=0.005整体最为稳定，在GCN、GraphSAGE及部分GAT配置中均能取得较高性能。较大的学习率（0.01）虽然收敛速度更快，best\_epoch通常更低（如GraphSAGE-2层仅需5个epoch），但深层网络易出现训练不稳定及性能劣化（如GIN-4层）。相反，较小的学习率（0.001）在GAT和GIN等敏感模型中有助于提升深层结构的稳定性和分类性能，但需要更长的训练时间，best\_epoch常接近上限50轮。

从性能指标的相关性来看，Acc与F1-micro基本一致，说明Cora数据集节点类别分布相对均衡；AUROC指标整体较高（>0.95），但在GIN-4层和部分GAT-4层的高学习率配置下显著下降，进一步印证了此类配置下模型类别区分能力下降的现象。

综上，图神经网络的性能受到模型结构、层数及学习率等超参数的显著影响。推荐的配置为：2~3层网络结构、学习率0.005，此类组合能够兼顾分类精度、收敛速度及稳定性。以具体模型为例，GCN在3层、lr=0.005时性能最佳（Acc=0.827，AUROC=0.971），而GAT在3层、lr=0.001时也能达到Acc=0.83的最优结果。对于GIN等在深层结构下不稳定的模型，应谨慎增加网络深度或使用较大学习率，以避免性能大幅退化。图表, 箱线图

AI 生成的内容可能不正确。

图 1: 不同学习率性能分布图

箱线图

AI 生成的内容可能不正确。

图 2：不同层数性能分布图

图形用户界面, 图表, 应用程序

AI 生成的内容可能不正确。

图 3：不同模型性能分布图

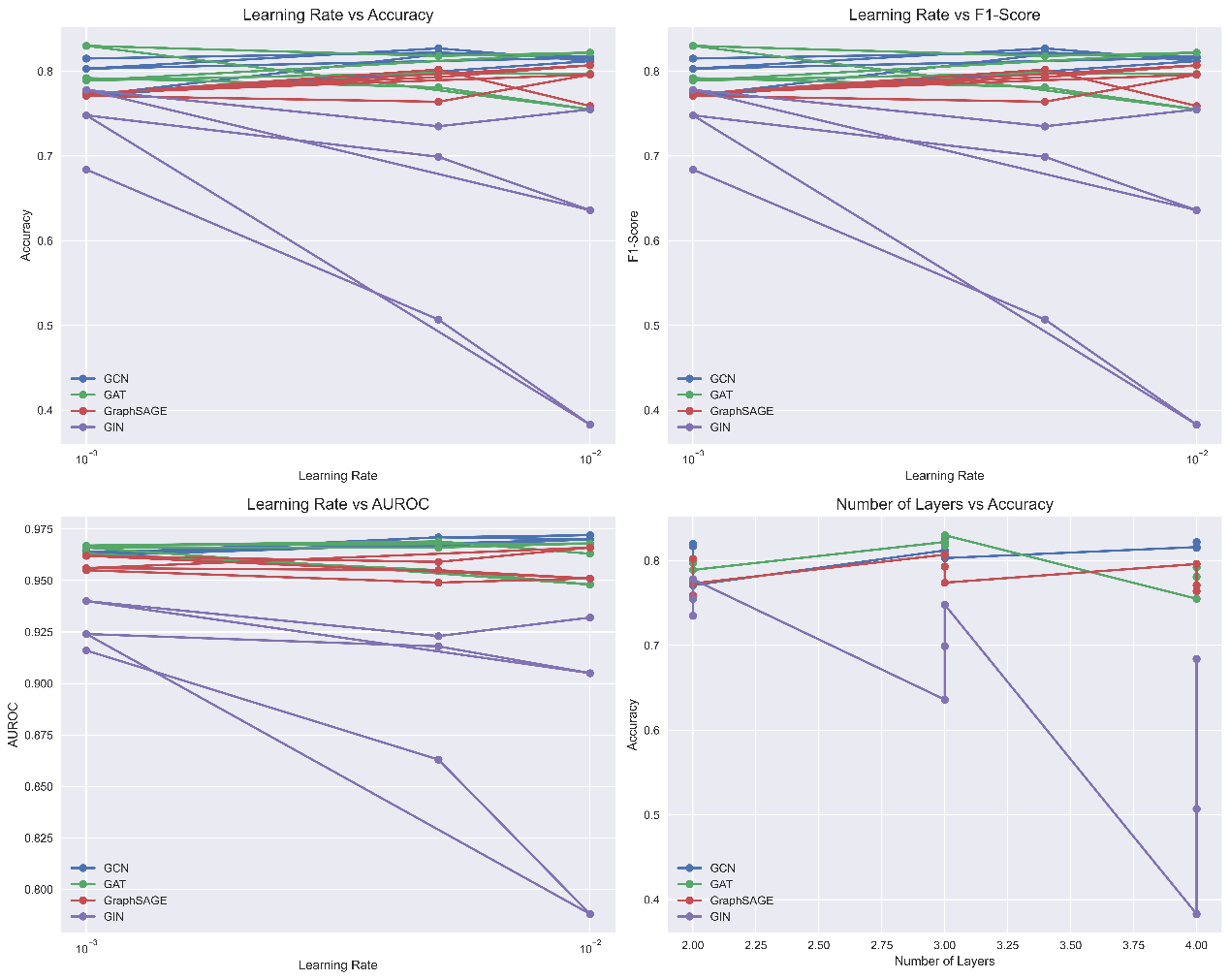


图 4：不同模型之间超参数变换的性能对比

图片包含 图表

AI 生成的内容可能不正确。

图 5：模型性能热力图

表 1 : 不同超参数对模型性能的影响

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **model** | **num\_layers** | **lr** | **best\_epoch** | **acc** | **f1\_micro** | **auroc** |
| GCN | 2 | 0.01 | 26 | 0.817 | 0.817 | 0.972 |
| 0.005 | 44 | 0.82 | 0.82 | 0.971 |
| 0.001 | 50 | 0.771 | 0.771 | 0.963 |
| 3 | 0.01 | 11 | 0.812 | 0.812 | 0.970 |
| 0.005 | 25 | 0.827 | 0.827 | 0.971 |
| 0.001 | 46 | 0.803 | 0.803 | 0.962 |
| 4 | 0.01 | 12 | 0.816 | 0.816 | 0.970 |
| 0.005 | 13 | 0.822 | 0.822 | 0.967 |
| 0.001 | 43 | 0.815 | 0.815 | 0.964 |
| GAT | 2 | 0.01 | 18 | 0.797 | 0.797 | 0.963 |
| 0.005 | 23 | 0.798 | 0.798 | 0.969 |
| 0.001 | 14 | 0.789 | 0.789 | 0.967 |
| 3 | 0.01 | 6 | 0.822 | 0.822 | 0.968 |
| 0.005 | 4 | 0.818 | 0.818 | 0.966 |
| 0.001 | 8 | 0.83 | 0.83 | 0.966 |
| 4 | 0.01 | 8 | 0.755 | 0.755 | 0.948 |
| 0.005 | 22 | 0.781 | 0.781 | 0.955 |
| 0.001 | 14 | 0.792 | 0.792 | 0.963 |
| GraphSAGE | 2 | 0.01 | 5 | 0.759 | 0.759 | 0.951 |
| 0.005 | 9 | 0.802 | 0.802 | 0.955 |
| 0.001 | 49 | 0.773 | 0.773 | 0.956 |
| 3 | 0.01 | 7 | 0.807 | 0.807 | 0.966 |
| 0.005 | 9 | 0.793 | 0.793 | 0.959 |
| 0.001 | 50 | 0.774 | 0.774 | 0.962 |
| 4 | 0.01 | 12 | 0.796 | 0.796 | 0.951 |
| 0.005 | 16 | 0.764 | 0.764 | 0.949 |
| 0.001 | 13 | 0.771 | 0.771 | 0.955 |
| GIN | 2 | 0.01 | 36 | 0.755 | 0.755 | 0.932 |
| 0.005 | 45 | 0.735 | 0.735 | 0.923 |
| 0.001 | 38 | 0.778 | 0.778 | 0.940 |
| 3 | 0.01 | 33 | 0.636 | 0.636 | 0.905 |
| 0.005 | 50 | 0.699 | 0.699 | 0.918 |
| 0.001 | 45 | 0.748 | 0.748 | 0.924 |
| 4 | 0.01 | 47 | 0.383 | 0.383 | 0.788 |
| 0.005 | 50 | 0.507 | 0.507 | 0.863 |
| 0.001 | 49 | 0.684 | 0.684 | 0.916 |

## 不同图池化方法对图分类任务性能的影响

表 2: 不同图池化方法对模型性能的影响

|  |  |  |
| --- | --- | --- |
| **Model** | **Pooling** | **Accuracy** |
| GCN | avg | 0.8 |
| max | 0.75 |
| min | 0.8 |
| GAT | avg | 0.8 |
| max | 0.8 |
| min | 0.85 |
| GraphSAGE | avg | 0.85 |
| max | 0.6 |
| min | 0.85 |
| GIN | avg | 0.7 |
| max | 0.75 |
| min | 0.75 |

表2展示了四类图神经网络模型（GCN、GAT、GraphSAGE、GIN）在采用不同图池化方法（平均池化 avg、最大池化 max、最小池化 min）时在图分类任务上的性能表现。整体来看，不同模型对池化策略的敏感性存在较大差异，部分模型的性能波动显著，说明池化操作对信息聚合及下游分类性能具有直接影响。

首先，GCN在不同池化方法下的性能相对稳定，平均池化（avg）与最小池化（min）均达到0.80的准确率，而最大池化（max）略低，仅为0.75。这表明GCN在特征表示的鲁棒性较强，池化策略变化对其性能影响有限，但过度依赖单一极值特征（max）可能导致全局图特征的丢失。其次GAT模型对池化方法表现出中等程度的敏感性。其准确率在avg与max池化下均为0.80，但min池化反而提升至0.85，成为最佳配置。分析原因可能在于GAT通过注意力机制对节点特征进行了加权聚合，而min池化能够强化低响应特征节点的区分度，避免注意力分布过于集中的偏差，从而提高模型的整体判别能力。然后GraphSAGE对池化方法极为敏感，在avg与min池化下均可达到最高准确率0.85，但在max池化时性能显著下降至0.60，存在25个百分点的差距。这表明GraphSAGE的特征聚合方式与最大池化之间存在适配性不足，可能是由于max池化过度依赖单一高值特征，导致GraphSAGE在全局图级表示中信息损失严重。相比之下，avg和min池化能够更好地保持多节点特征的信息完整性，从而取得较优性能。

最后GIN的表现相对均衡，max和min池化均可达到0.75的准确率，优于avg池化（0.70）。GIN依赖MLP捕捉邻居节点特征的复杂组合，avg池化可能过于平滑化节点特征，降低区分度，而max和min池化能在一定程度上增强边界样本的特征表达，提升分类性能。

综合来看，平均池化并非所有模型的最优选择，而最小池化（min）在GAT与GraphSAGE等模型中能够有效提升性能，尤其是GAT的min池化准确率最高（0.85）。另一方面，最大池化（max）在GraphSAGE中表现极差，提示在特定模型结构下，极值池化可能导致特征表示失真。综上，图池化方法的选择对图分类性能影响显著，且不同模型存在适配性差异。推荐GraphSAGE与GAT优先使用min池化，而GCN与GIN可在avg与min池化之间选择，以获得较稳定的性能表现。未来研究可探索自适应池化策略，将avg、max、min等多种信息动态融合，从而提升不同模型在图分类任务中的鲁棒性与泛化能力。

## 全图训练与分批次训练在模型性能及运行效率

表 3: 全图和分批次训练模型性能比较

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Train** | **model** | **acc** | **f1\_micro** | **auroc** | **epoch\_time** |
| Full graph | GCN | 0.808 | 0.808 | 0.967189 | **0.366** |
| GAT | 0.773 | 0.773 | 0.964343 | **0.244** |
| GraphSAGE | 0.772 | 0.772 | 0.962688 | **0.100** |
| GIN | 0.745 | 0.745 | 0.932227 | **0.098** |
| Sample | GCN | 0.805 | 0.805 | 0.969845 | **0.388** |
| GAT | 0.8 | 0.8 | 0.965053 | **0.190** |
| GraphSAGE | 0.778 | 0.778 | 0.963508 | **0.472** |
| GIN | 0.77 | 0.77 | 0.94005 | **0.091** |

表3展示了四类图神经网络模型（GCN、GAT、GraphSAGE、GIN）在采用全图训练（Full graph）和分批次训练（Sample）两种策略时的在cora数据集上进行节点分类性能及训练时间（epoch\_time）对比。整体结果表明，两种训练方式在性能表现上差异较小，但在运行效率和模型稳定性上呈现一定规律。

首先，从性能指标（Acc、F1-micro、AUROC）来看，分批次训练与全图训练的准确率和F1-micro差异普遍不超过2个百分点，AUROC指标也保持相对稳定。例如，GCN在全图训练和分批次训练下的Acc分别为0.808和0.805，差异极小（0.3个百分点）。类似地，GraphSAGE和GIN的性能也基本持平，说明分批次训练能够在不显著牺牲分类性能的情况下替代全图训练。值得注意的是，GAT在分批次训练下的准确率（0.800）和F1-micro均明显优于全图训练（0.773），表明采样策略对GAT的特征学习具有一定的正向作用，可能是由于分批采样缓解了全图训练中过拟合或注意力集中化的问题。其次，训练时间开销存在明显差异。全图训练的单轮训练时间通常低于或接近分批次训练，尤其在GraphSAGE模型中差异最为突出，全图训练的epoch\_time仅为0.100，而分批次训练高达0.472，几乎增加了3.7倍。这一现象可归因于分批次训练需要额外的图采样与邻居扩展操作，导致单轮训练时间增加。另一方面，GCN的分批次训练耗时（0.388）略高于全图训练（0.366），但差距不大；GIN则相反，分批次训练更高效（0.091 vs 0.098），说明分批采样在某些计算较轻的模型上能够改善显存利用和并行效率。

综合来看，分批次训练在大规模图或受显存约束的场景下更具优势，因其可显著降低内存占用并保持与全图训练相当甚至略优的性能（如GAT）。但在小规模图或对单轮训练时间敏感的任务中，全图训练仍是更高效的选择，尤其是GraphSAGE等采样代价较高的模型。未来可考虑结合图采样优化与显存管理技术，进一步缩小分批次训练与全图训练在效率上的差距，同时探索动态采样或混合训练策略，以兼顾性能与计算开销。

## 总结

研究围绕“模型结构—超参数—训练策略—池化机制”四重视角，在五个公开图数据集上对四类代表性图神经网络（GCN、GAT、GraphSAGE、GIN）进行了系统的节点级与图级任务评估。主要结论如下：

1. 结构深度与超参数的耦合效应显著。总体而言，2–3 层网络在节点分类任务中即可达到性能饱和，继续加深至 4 层对 GCN、GAT 仅带来边际增益甚至退化，而对 GIN 则造成急剧的性能坍塌（Acc 最低 0.383），验证了过平滑与梯度衰退在深层 GNN 中的现实制约。学习率方面，0.005 在精度与收敛速度之间取得最佳平衡；0.01 加速收敛但稳定性差，0.001 虽稳健却显著拉长训练周期。
2. 图池化策略对图级任务具有模型依赖性的显著影响。Min-pooling 在 GAT 与 GraphSAGE 上带来最高 0.85 的准确率，而 max-pooling 使 GraphSAGE 性能骤降 25 个百分点，表明极值池化可能丢失全局拓扑信息；GCN 与 GIN 对池化变化相对鲁棒。结果提示，池化设计需与模型聚合机制协同考虑，避免“一刀切”。
3. 训练范式对比显示，分批次训练在保持性能（±2%）的同时，显存占用远小于全图训练，尤其适用于大规模图。GAT 在分批次下反而获得 2.7 个点的准确率提升，暗示采样带来的正则化效应可缓解注意力过集中问题。然而，GraphSAGE 的分批采样开销使单轮时间增加 3.7 倍，提示采样策略需与模型计算密度相匹配。

综合来看，推荐实践者在节点分类任务中优先采用 GCN 或 GAT 的 2–3 层、学习率 0.005 的配置；在图分类任务中，GAT 与 GraphSAGE 宜选用 min-pooling，GCN/GIN 可折衷选择 avg 或 min-pooling；面对资源受限或大规模场景，优先采用分批次训练，并针对具体模型权衡采样开销。未来工作可进一步探索（1）基于可微架构搜索的自适应层数与池化联合优化，（2）融合梯度压缩与动态采样的高效训练框架，以及（3）面向深层 GNN 的过平滑抑制机制，以实现精度、效率与可扩展性的协同提升。

实验报告中未提及的训练结果数据可以在代码文件夹中查看。