文章来源

github 代码地址

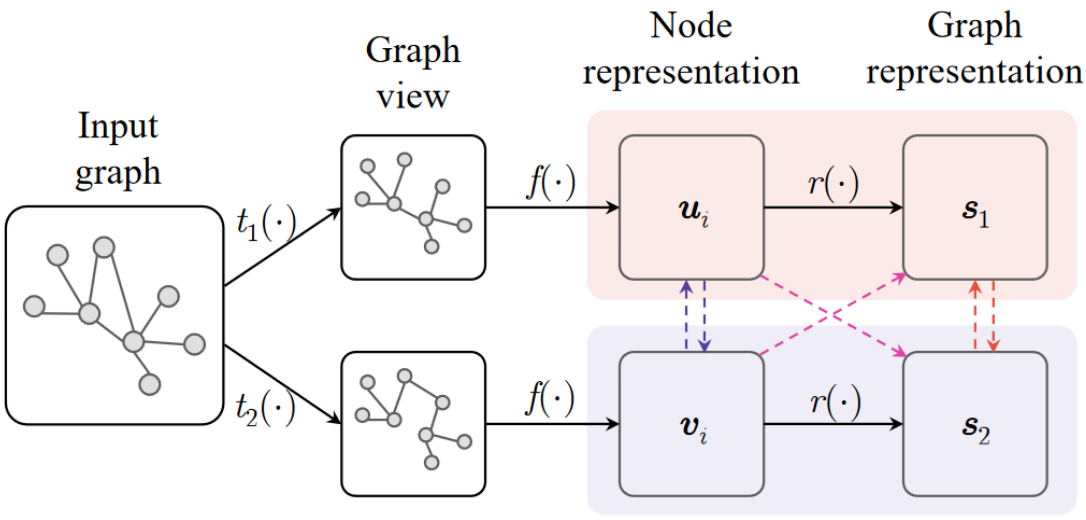
文章地址

Abstract

图对比学习(Graph Contrastive Learning，GCL)建立了一种无需人工标注的图表示学习新范式。尽管 GCL 取得了令人瞩目的进展，但其背后的成功仍然有些神秘。在这项工作中，我们首先确定了一般 GCL 范式中的几个关键设计考虑因素，包括增强功能、对比模式、对比目标和负挖掘技术。然后，为了理解不同 GCL 组件之间的相互作用，我们在不同领域的数据集上对一组基准任务进行了广泛的控制实验。我们的实证研究表明了一组有效的 GCL 的普遍收益，例如产生稀疏图视图的简单拓扑增强带来了有希望的性能提升；对比模式应与末端任务粒度对齐。此外，为了促进未来的研究和简化 GCL 算法的实现，我们开发了一个易于使用的库 Py GCL，该库具有模块化的 CL 组件、标准化的评估和实验管理。我们期望这项工作能够为有效的 GCL 算法提供有用的经验证据，并为未来的研究提供一些见解。

Introduction

通过对比正负样本学习表示。以每个节点作为中心节点为例，正样本一般是其他视图中表示一致的一些节点，而负样本则是在该 Batch 中的给定图或者其他图中选择其他节点作为负样本



一个视图中一致的表示对拉在一起，同时将其他视图拉开。额外的负挖掘技术可以用来提高模型的性能。

GCL 的一般流程。我们从 4 个维度对 GCL 算法进行分解：(a)数据增强函数，(b)对比模式，(c)对比目标，(d)负挖掘策略。

- 在训练的每次迭代中，我们首先执行随机增强从输入图中生成多个图视图。
- 对于每个节点嵌入 vi 作为锚实例，对比模式指定一个正集 P (vi) = { pi } Pi = 1 和一个负集 Q (vi) = { qi } Qi = 1。
- 通过考虑负样本的相对相似度(即硬度)，我们可以采用负挖掘策略来改进负样本集。
- 最后，我们使用一个对比目标 J 对这些指定的正负对进行打分。

本位提出了**三个问题**，并在文中进行了解答：

- 一个有效的 GCL 算法中最重要的组成部分是什么?
- 不同的设计考虑如何影响模型性能?
- 这些设计考虑是否有利于某些特定类型的数据或终端任务?

为设计高效的 GCL 算法，实验得出几个指导原则：

- 生成稀疏视图的拓扑增强对 GCL 的提升最大。从**拓扑结构和特征级别**都做增强会进一

步提升效果。

- 对比模式的尺度应该与下游任务的粒度一致，即下游任务是节点级任务，对比应该是节点级别的。
- **InfoNCE 目标函数最稳定且效果提升最好，但是要求大量的负样本**
- **一些免负采样的目标函数可以保证效果的同时降低计算复杂度**
- 基于 **embedding 相似度的负采样策略对 GCL 效果提升甚微**

Data Augmentations

数据增强的目的是为给定图生成一致的，恒等的正样本。目前大多数 GCL 是使用两级的增强技术即结构转化和特征转化。

Topology augmentations

边：1）边移除(ER)， 2）边添加(EA)， 3）边翻转(EF)，

点：1）点丢弃(ND)，2）随机游走的子图(RWS)， 3）使用个性化 pagerank 的扩散(PPR)，4）使用马尔可夫扩散核的扩散(MDK)

Feature augmentations

1）特征遮掩(FM)， 2）特征丢弃(FD)

Contrasting modes

对于每一个点，对比模式需要确定图上不同粒度的正负样本集、在主流的工作中广泛应用的三种对比模式即 1）local-local，2）global-global，3）global-local。局部-局部和全局-局部 CL 适用于节点数据集，其中三种模式均可用于图数据集

Contrastive objectives

目标函数是用于衡量正样本之间的相似性和负样本之间的差异性的。按照是否需要负采样分为

1）依赖负样本

a. InfoNCE，b. Jensen-Shannon Divergence，c. Triplet Margin loss

2）不依赖负样本

a. Boostrapping Latent loss，b. Barlow Twins loss，c. VICReg loss

Negative mining strategies

- Hard Negative Mixing
- Debiased Contrastive Learning
- Hardness-Biased Negative Mining
- Conditional Negative Mining

Method	Primary task	Topology augmentation	Feature augmentation	Contrasting mode	Dual branches?	Contrastive objective
DGI [13]	Node classification	—	—	Global-local	✗	JSD
GMI [15]	Node classification	—	—	Global-local	✗	SP-JSD
InfoGraph [19]	Graph classification	—	—	Global-local	✗	SP-JSD
MVGRL [14]	Node & graph classification	PPR	—	Global-local	✓	JSD
GCC [24]	Transfer learning	RWS	—	Local-local	✗	InfoNCE
GraphCL [16]	Graph classification	RWS/ND/EA/ED	FD	Global-global	✓	InfoNCE
GRACE [17]	Node classification	ER	MF	Local-local	✓	InfoNCE
GCA [18]	Node classification	ER	MF	Local-local	✓	InfoNCE
BGRL [25]	Node classification	ER	MF	Local-local	✓	BL
GBT [26]	Node classification	ER	MF	Local-local	✓	BT

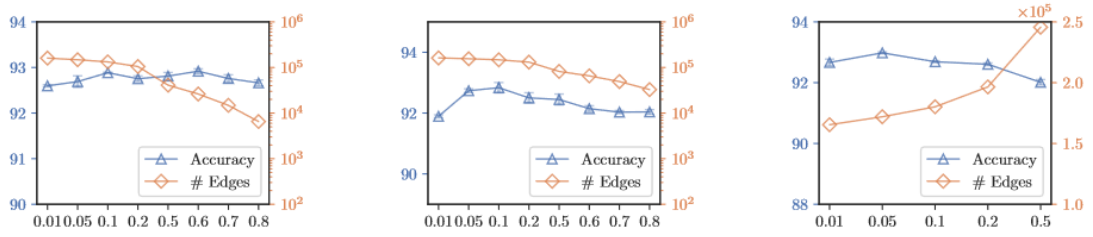
有几次评估 GCL 模型的表达能力

Data Augmentations 对结果的作用

Augmentations: *eval* Contrasting mode: InfoNCE Objectives: L–L Negative mining strategy: None

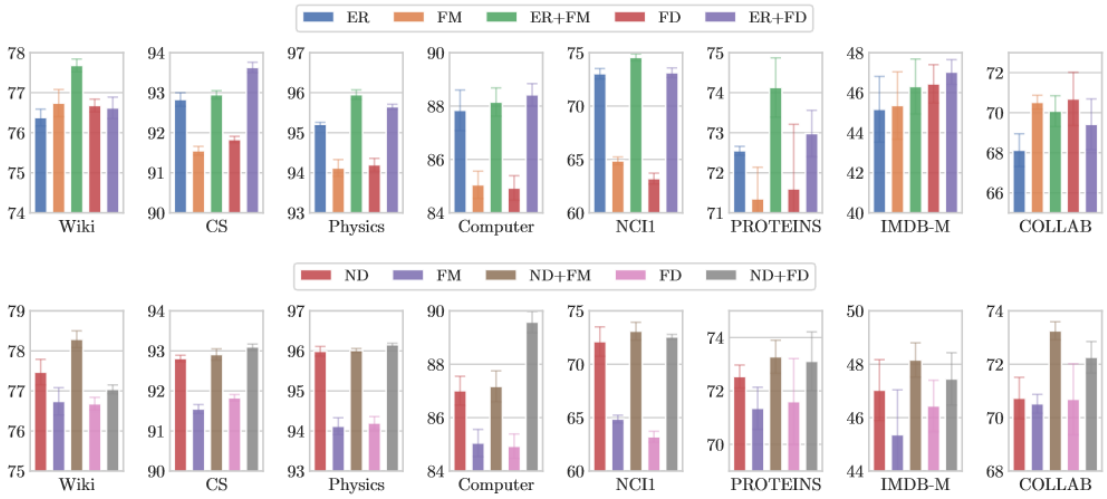
Observation1：拓扑增强对模型性能影响很大。产生更稀疏图的增强函数通常会带来更好的性能。

随着节点或边的减少，准确率会有一定的提升但随着大量的节点或边丢失，效果会极具下降。当边添加的越多时，准确率下降，这是因为大多数现实的图一般是稀疏的，太多的边添加进来会引入噪声，降低学习到的表示的质量。



(a) Node dropping probability (b) Edge removing probability (c) Edge adding probability

Observation2：特征增强为 GCL 带来了额外的好处。结构和属性层面的组合增强对 GCL 的益处最大。



FM 代表 Feature Masking，它随机掩盖节点功能的子集，而 FD 代表 Feature Dropout，它随机删除节点功能的子集。FM 和 FD 都类似于在输入层上应用滤波技术。该论文表明，除了结构增强外，使用特征增强还有益于 GCL，这表明拓扑和结构对于学习图表示都很重要。

从实验结果来观察：ER+FM，ND+FM 效果都会比较好

Observation3：确定性增强方案应伴随着随机增强。

单一使用确定性的增强并不一定会达到最好的结果。对比学习的目标函数就是区分从真实数据分布中采样的样本和从噪声分布中采样的样本。因此，随机增强可以更好的近似噪声分布。

Contrasting Modes and Contrastive Objectives 对结果的作用

Augmentations: ND + FM	Contrasting mode: *eval*	Objectives: *eval*	Negative mining strategy: None

Observation4：同尺度对比一般表现更好。不同粒度的下游任务倾向于不同的对比模式。

local-local 在节点分类任务中效果最好，**global-global 在图级任务**中效果最好。一种可能的解释是在 global-local 模式中，图中所有的节点表示可能恰好是每个图嵌入的正样本即在这种模式下，会把所有 node-graph 对拉入一个嵌入空间中，造成次优结果。

Observation5：在基于负样本的目标中，使用 InfoNCE 目标可以在所有设置中表现最好

Observation6：辅助程序 Latent 和 Barlow Twins 损失在没有明确负样本的情况下获得了与基于负样本的损失相当的性能，同时减少了计算负担

Negative Mining Strategies 对结果的作用

Augmentations: ND + FM	Contrasting mode: InfoNCE	Objectives: L–L	Negative mining strategy: *eval*

observation7：现有的基于计算嵌入相似度的负挖掘技术给 GCL 带来的收益有限

目前的负采样技术一般是通过计算样本嵌入的内积来计算样本之间相似性的。在无监督情况下，目标函数会把不同的表示分开而不考虑真实的语义关系。更糟糕的是，大多是 GCN 会倾向于为邻居节点生成相似的表示并不会考虑语义信息，这样会加重负采样选择的偏差。一些可能是正样本的样本被选为负样本会阻碍训练效果。

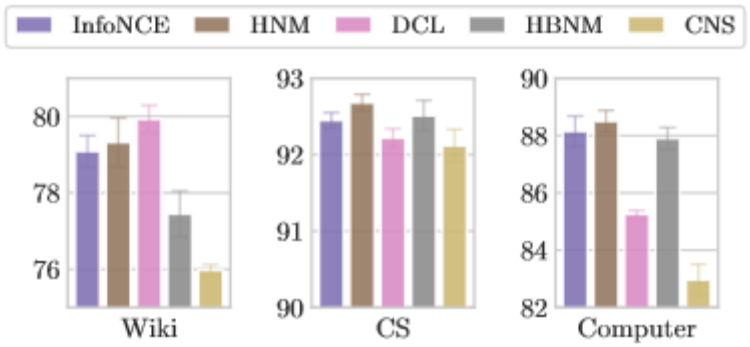


Figure 6: Performance of negative mining strategies.

Conclusion

在本文中，我们首先给出了 GCL 的一个分类，从四个方面对现有的工作进行了分类：**数据增强、对比模式、对比目标和负挖掘策略。**然后，我们通过在一组全面的基准测试任务和数据集上进行广泛的实证研究，分析了每个 GCL 组件的设计选择。我们的严谨的实证研究揭示了 GCL 成分的几个有趣的发现，这些发现可能有助于未来算法的发展。我们还提供了一个开源的基于 PyTorch 的库 PyGCL，以方便 GCL 模型的实现。虽然 GCL 已经在各种下游任务中展示了强大的实证性能，但它仍处于起步阶段，许多挑战仍然广泛存在。我们希

望我们的工作能够为这个充满活力的领域的未来研究提供一些实际的指导。

1）影响因素的局限。本文只从四个角度分析了影响 GCL 效果的因素，但是对于模型相关的因素如是否在 InfoNCE 目标函数中加入映射头以及在 GCL 中应该用什么图编码器有还没有分析到。

2）下游任务的局限。本文主要从节点和图级别的分类任务，图级别的回归任务上做了处理。大量下游任务如连接预测，社区检测都尚未考虑

3）缺乏理论分析。本文只是从实验角度分析了结果，缺乏理论解释。

未来方向

1）**自动增强。**我们知道拓扑结构增强对于 GCL 是至关重要的，但是现存工作还是要手动设计增强策略，这样可能会导致次优化。目前，图结构学习上的工作可以学习最优的图结构，可以用于自动学习合适的增强函数

2）**理解前置任务和下游任务之间的表现差异。**我们的工作分析了最终任务和对比目标函数的关系，但是前置任务和下游任务之间的效果差异并未分析

3）**基于结构感知的负采样。**在视觉领域，相似的视觉特征一般语义类别也相似，但是在图结构中很难通过嵌入相似性进行衡量。以前的图嵌入工作在结构角度设计了一些方法，但是如何整合丰富的结构信息为 GCL 建模更好的负分布尚未探索。