北京智源大会 BAAI CONFERENCE 2022.5.31-6.02

CogDL: 一个高效的图神经网络训练框架

清华大学计算机系
博士生 岑宇阔

汇报内容

01 背景介绍

02 CogDL训练框架

03 CogDL核心技术

04 CogDL高效后端

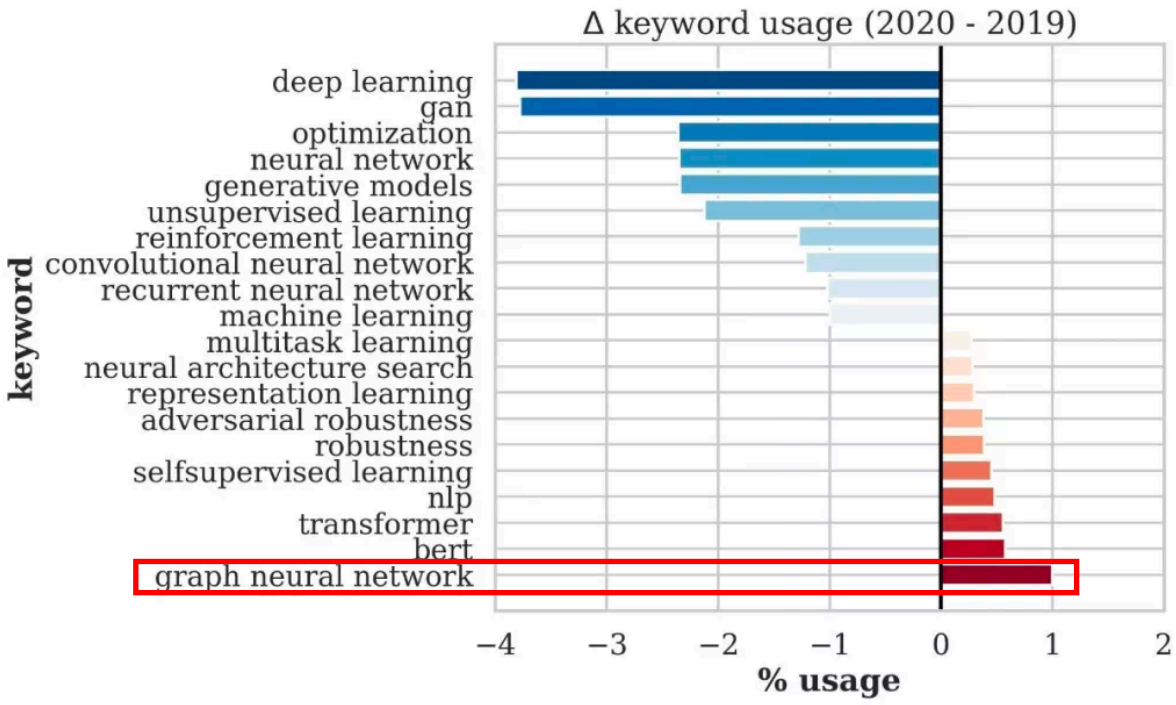
05 CogDL生态建设

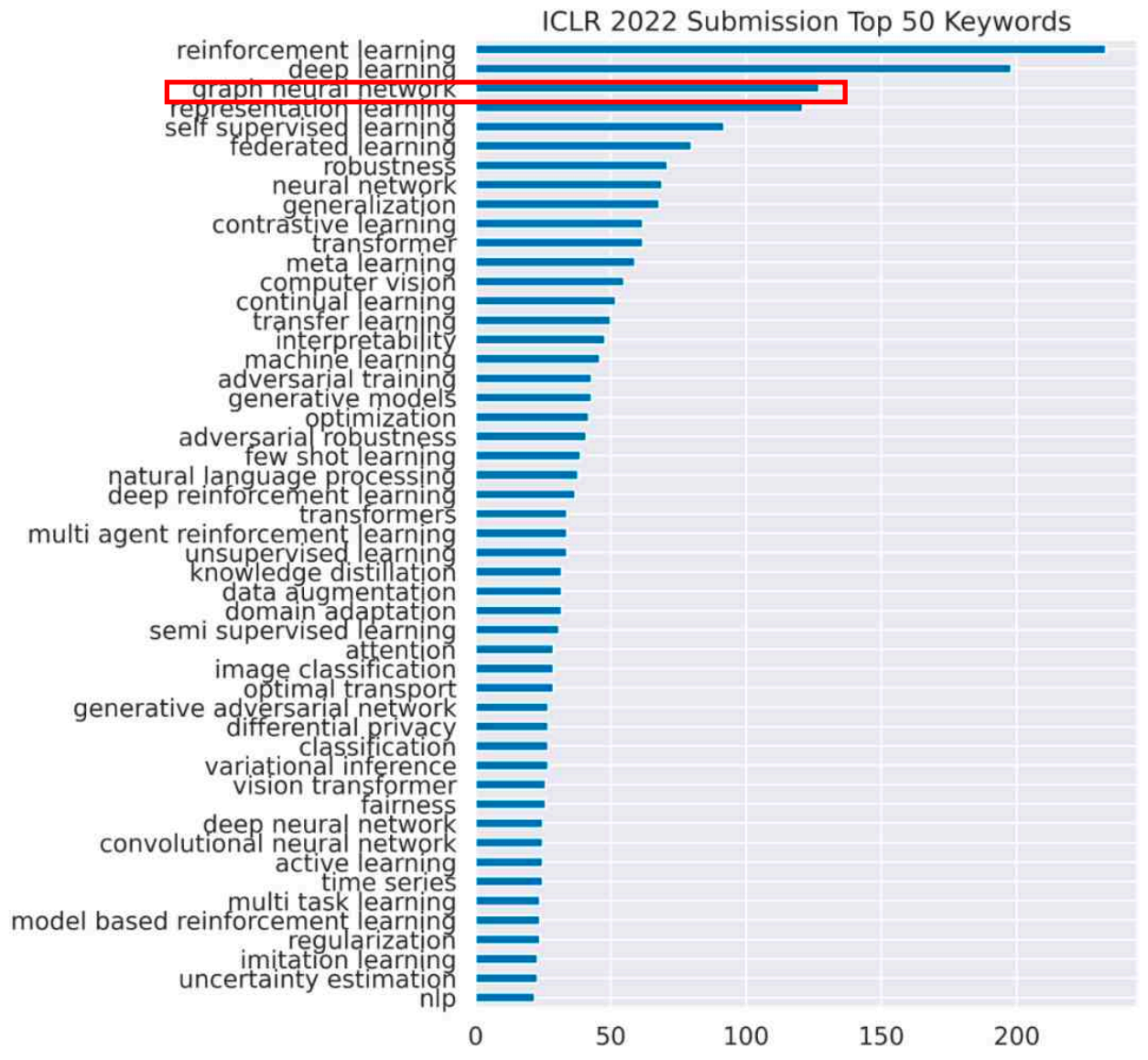
01

背景介绍

图神经网络（Graph Neural Network, GNN）

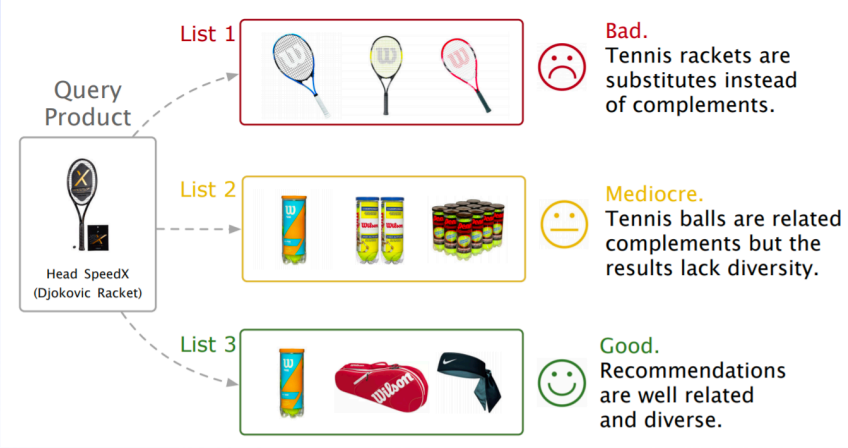
GNN在学术界中成为热门领域

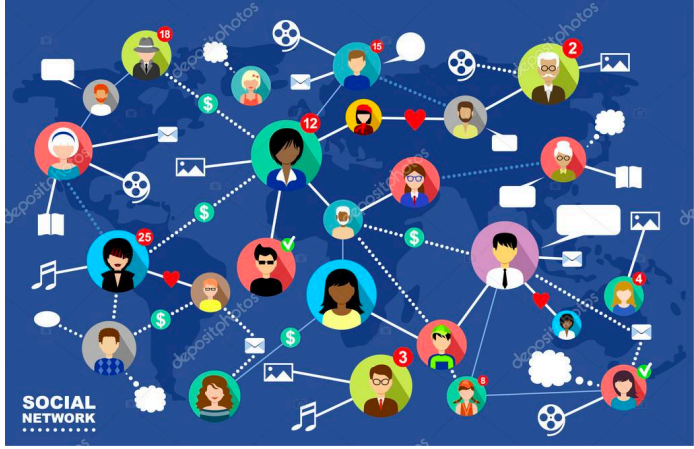


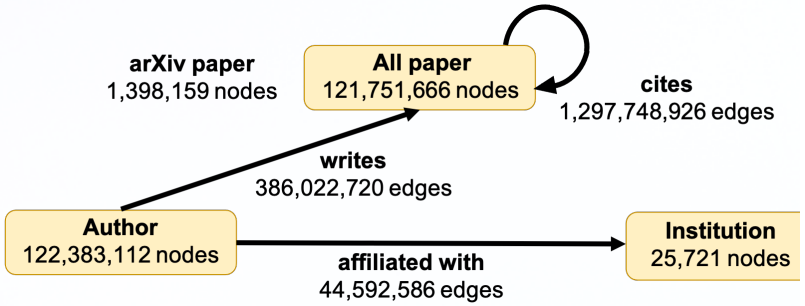


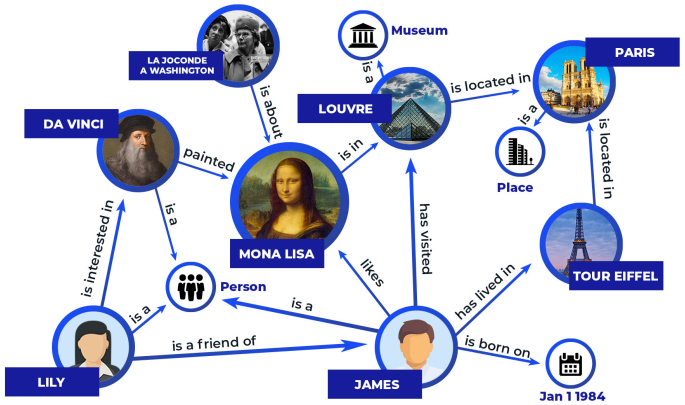
GNN的应用

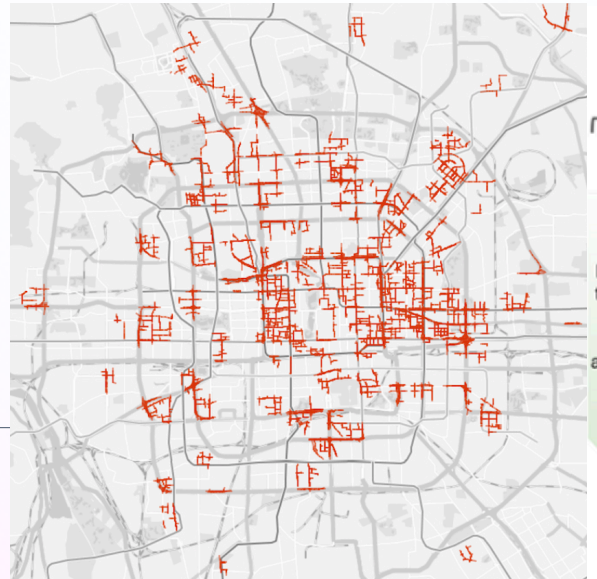
- 社交网络分析
- 学术网络挖掘
- 推荐系统
- 知识图谱
- 交通数据预测
- 药物发现
- 金融反洗钱
- 芯片布局
- …

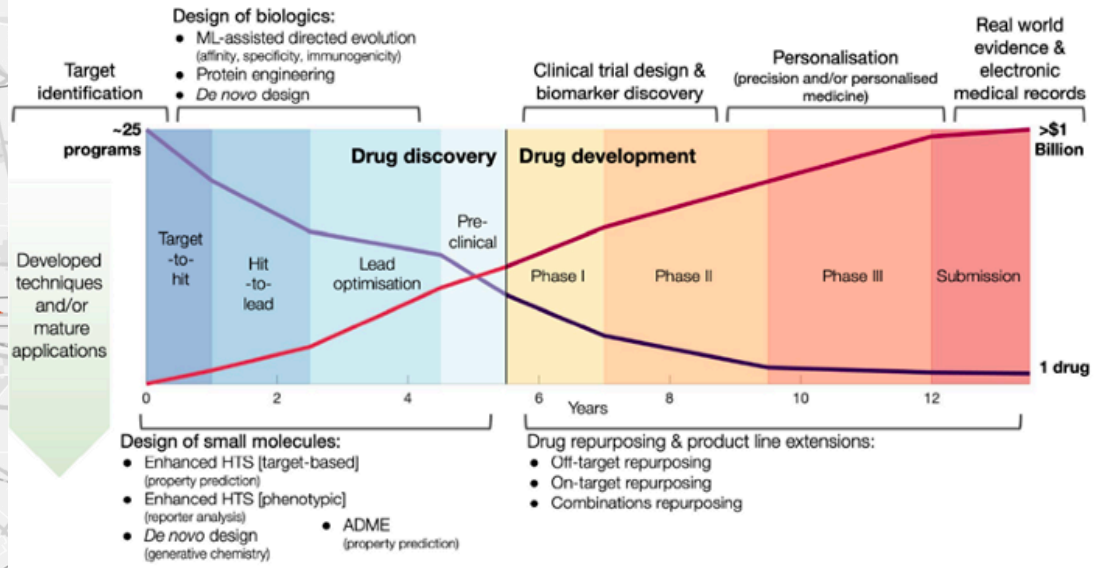












GNN框架发展

北京智源大会 BAAI CONFERENCE 2022.5.31-6.02

基础GNN支持 → 高效GNN训练 → GNN生态

国际

PyG TU Dortmund

DGL AWS

Graph Nets Lib DeepMind

ptgnn 微软

PyG 2.0 TU Dortmund Stanford

Jraph DeepMind

ptgnn v0.10 微软

DGL v0.8 AWS

2018.12 | 2019.10 | 2020.03 | 2020.05 | 2020.10 | 2020.12 | 2021.9 | 2021.10 | 2021.11 | 2021.11 | 2022.03 | 2022.06

GNN生态

AliGraph 阿里

PGL 百度

CogDL 清华KEG

PGL 2.0 百度

CogDL v0.5 清华KEG

CogDL v0.5.3 清华KEG

国内

downloads 23k

Fork 262 | Starred 1.1k

https://github.com/THUDM/cogdl

02

CogDL训练框架

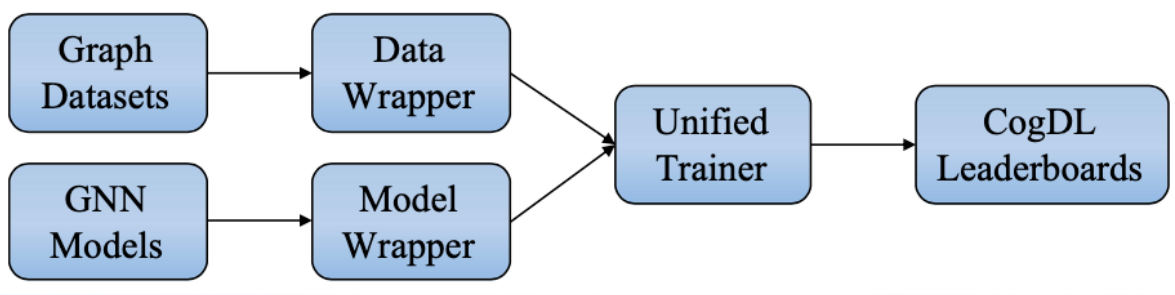
CogDL统一的训练流程

统一的训练器

CogDL开发了统一的GNN训练器（Unified Trainer）来减少冗余代码并提供了训练优化、实验管理等选项。

解耦式的模块设计

分为四个部分：数据集（Dataset）、数据准备（Data Wrapper）、模型（Models）、模型训练方式（Model Wrapper）。



	Model	Model Wrapper	Data Wrapper
GCN [3]	GCN	supervised	full-graph
GAT [23]	GAT	supervised	full-graph
GraphSAGE [4]	SAGE	sage_mw[1]	neighbor sampling
Cluster-GCN [24]	GCN	supervised	graph clustering
DGI [7]	GCN	dgi_mw[2]	full-graph
GRAND [6]	RandProp[3]	consis_reg_mw[3]	full-graph

[1] Yukuo Cen, Zhenyu Hou, ..., Peng Zhang, Guohao Dai, Yu Wang, Chang Zhou, Hongxia Yang, Jie Tang. CogDL: A Toolkit for Deep Learning on Graphs. In arXiv:2103.00959.

用法示例与对比

CogDL

```
model = GCN(...)
data = CoraDataset()[0]
mw = ModelWrapper(model, ...)
dw = DataWrapper(data, ...)
trainer = Trainer(epochs=100)
result = trainer.run(mw, dw)
```

PyG

```
model = GCN(...)
data = Planetoid(name="Cora")[0]
optimizer = torch.optim.Adam(...)
for epoch in range(100):
    pred = model(data.x, data.edge_index)
    labels = data.y
    mask = data.train_mask
    loss = F.nll_loss(pred[mask],
                      labels[mask])
    optimizer.zero_grad()
    loss.backward()
    optimizer.step()
    val_acc = evaluate(model, data)
    if val_acc > best_val_acc:
        best_val_acc = val_acc
        best_model = deepcopy(model)
result = test(best_model, data)
```

DGL

```
model = GCN(...)
data = CoraGraphDataset()[0]
optimizer = torch.optim.Adam(...)
for epoch in range(100):
    pred = model(data)
    labels = data.ndata['label']
    mask = data.ndata['train_mask']
    loss = F.nll_loss(pred[mask],
                      labels[mask])
    optimizer.zero_grad()
    loss.backward()
    optimizer.step()
    val_acc = evaluate(model, data)
    if val_acc > best_val_acc:
        best_val_acc = val_acc
        best_model = deepcopy(model)
result = test(best_model, data)
```

基于统一训练器的优化

北京智源大会 BAAI CONFERENCE 2022.5.31-6.02

混合精度训练

CogDL借助PyTorch的amp进行混合精度训练来减少GPU显存占用并加速模型训练。仅需通过设置fp16=True即可启用。相关性能见右表所示。

2层GCN模型在Reddit数据集上的性能（单精度/混合精度）

	GPU Memory	Accuracy	Training Speed	
			2080 Ti	3090
Single precision training	5,567 MB	95.44 ± 0.07	2.20 it/s	3.93 it/s
Mixed precison training	4,046 MB	95.35 ± 0.08	3.17 it/s	7.97 it/s

激活压缩训练

CogDL集成了最前沿的激活压缩训练技术来大幅降低训练过程中的显存占用。从右表所示，相关技术在准确率大致相同的情况下能带来6.4x~16x的显存降低。

2层GCN模型在3个数据集上的准确率和激活的显存占用（MB）

	origin (32-bit)	+actnn (4-bit)	+actnn (3-bit)	+actnn (2-bit)
Flickr	51.17 ± 0.19 (288)	51.08 ± 0.18 (37)	51.14 ± 0.18 (26)	51.20 ± 0.18 (18)
Reddit	95.33 ± 0.07 (1532)	95.32 ± 0.07 (194)	95.31 ± 0.07 (158)	95.34 ± 0.06 (112)
Yelp	39.86 ± 0.94 (4963)	40.06 ± 0.74 (773)	40.21 ± 0.82 (665)	39.89 ± 1.45 (551)

[1] https://pytorch.org/docs/stable/amp.html.
[2] ActNN: Reducing Training Memory Footprint via 2-Bit Activation Compressed Training. Chen, Jianfei and Zheng, Lianmin and Yao, Zhewei and Wang, Dequan and Stoica, Ion and Mahoney, Michael W and Gonzalez, Joseph E. In ICML'21.

基于统一训练器的功能

北京智源大会 BAAI CONFERENCE 2022.5.31-6.02

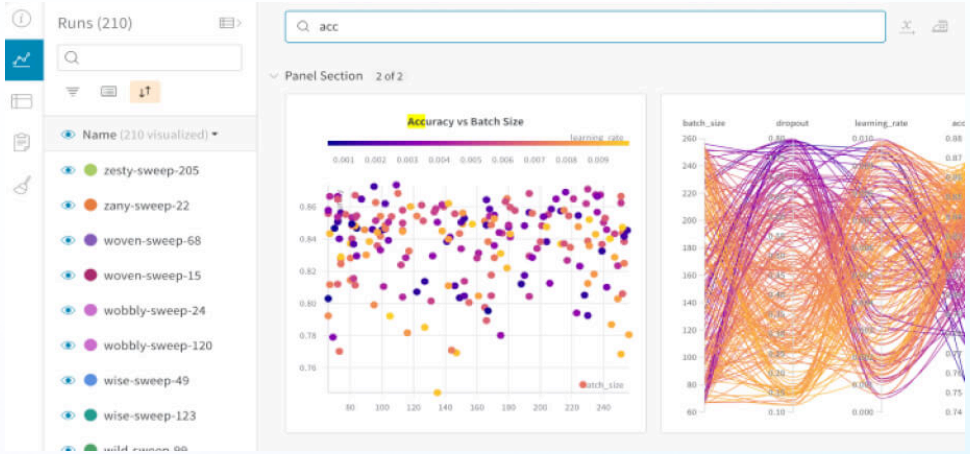
超参搜索

CogDL集成了高效的AutoML库（Optuna）来为用户提供最简洁的使用方式。如右侧所示，仅需在API中传入超参的搜索范围即可。

```
def func(trial):
  return {
    "lr": trial.suggest_categorical("lr",[0.01, 0.1]),
    "hidden_size": trial.suggest_categorical("hidden_size", [32, 64, 128]),
    "dropout": trial.suggest_uniform("dropout", 0.1, 0.8),
  }
experiment(dataset="cora", model="gcn", search_space=func) # hyper-parameter search
```

实验管理

实验管理逐渐成为了训练深度学习模型的必需品，能够更好地帮助用户调试。除了常用的 TensorBoard 之外，CogDL 还集成了WandB这个更加好用的实验管理工具。仅需在API中设置相应的logger即可开启。



[1] Optuna: A Next-generation Hyperparameter Optimization Framework. Akiba, Takuya and Sano, Shotaro and Yanase, Toshihiko and Ohta, Takeru and Koyama, Masanori. In KDD'19
[2] Experiment Tracking with Weights and Biases. Biewald, Lukas. https://www.wandb.com/.

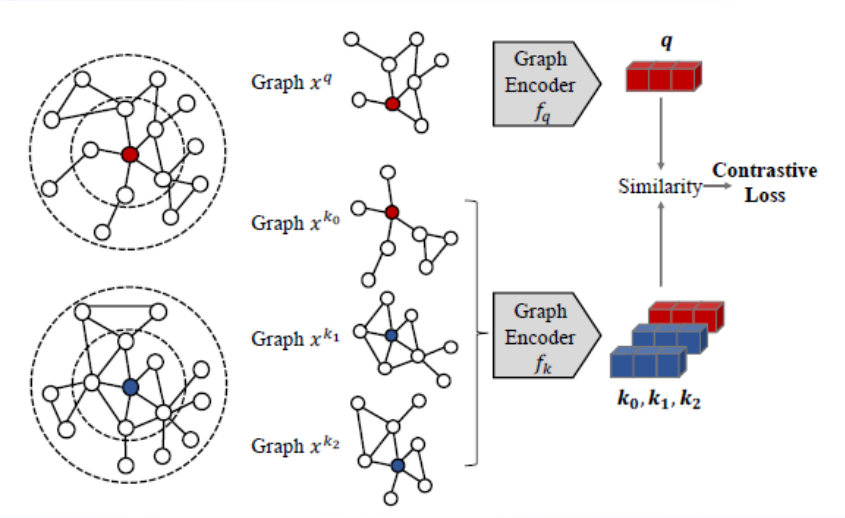
03

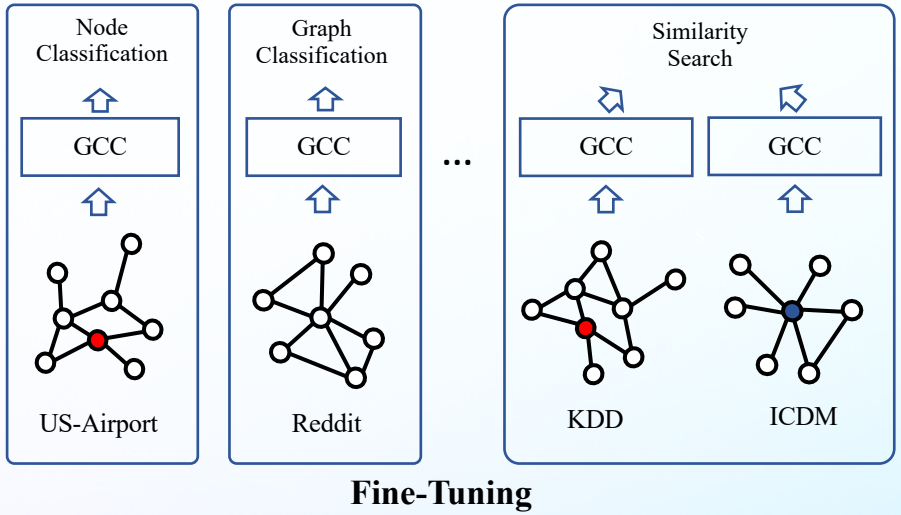
CogDL核心技术

图自监督学习与预训练技术

图对比编码GCC (KDD'20)

GCC采用了对比学习的框架，基本思想即让正例的表示在空间更相近，并让负例距离更远。正例的样本是从同一个节点采样得到的子图，而大量的负例是从其他节点采样的子图。获得了正例和负例后，我们需要构建图编码器实现图表征学习。GCC可以应用于许多下游的图学习任务，包括节点分类、图分类和相似搜索。





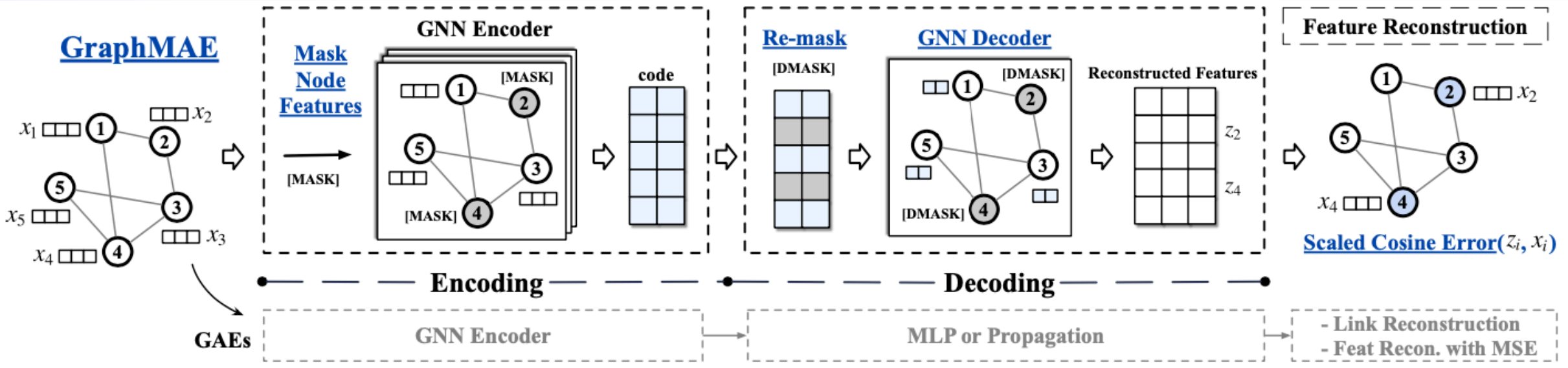
[1] Jiezhong Qiu et al. GCC: Graph Contrastive Coding for Graph Neural Network Pre-Training. In KDD'20.

图自监督学习与预训练技术

掩码图自编码器GraphMAE (KDD'22)

对图数据来说，图自编码器(Graph Autoencoder, GAE)可以重建图的结构信息或者节点特征。大多数已有图自编码器主要关注链接预测、图聚类任务，因此通常选择重建图的结构信息。

不同于以往的图自编码器，GraphMAE发现仅仅重建节点特征便能够使得模型学习到充分的信息，从而有助于下游分类等任务。主要的改进包括：带掩码的节点特征重建，带重掩码的解码过程，使用放缩余弦误差作为损失函数。

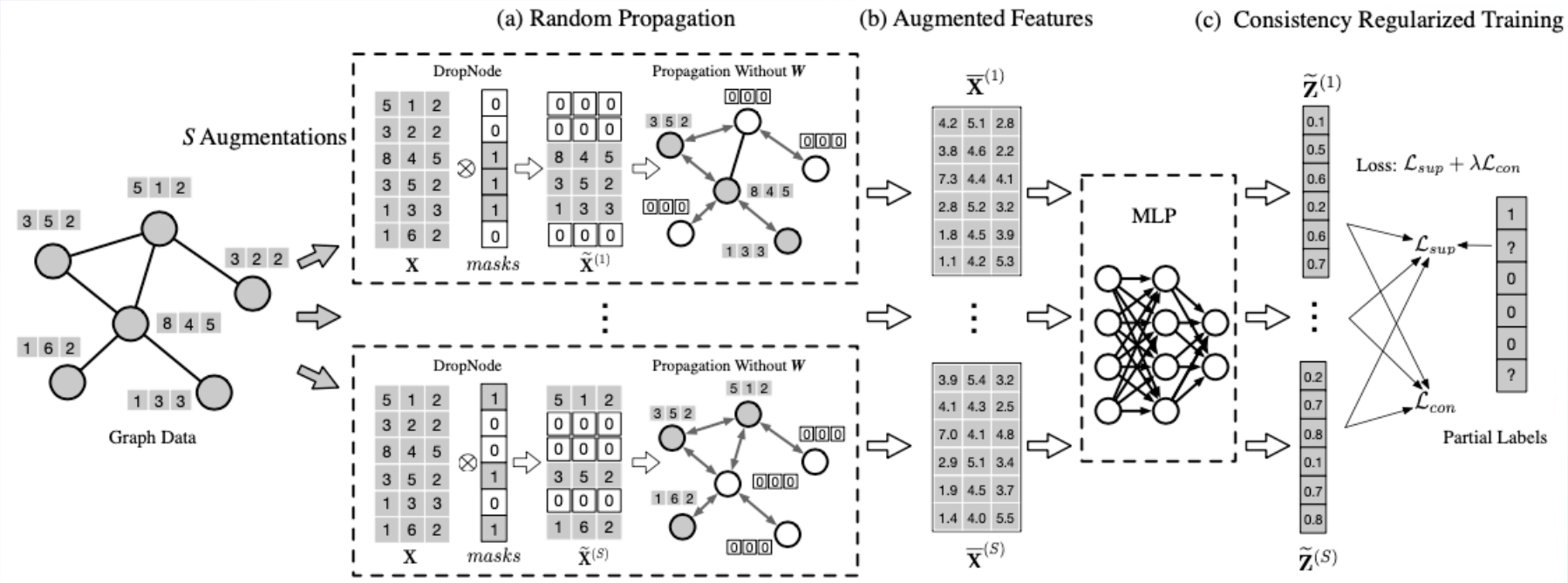


[1] Zhenyu Hou, Xiao Liu, Yukuo Cen, Yuxiao Dong, Hongxia Yang, Chunjie Wang, and Jie Tang. GraphMAE: Self-Supervised Masked Graph Autoencoders. In KDD'22.

图半监督学习技术

图随机神经网络GRAND(NeurIPS’20)

设计了一种随机传播策略来进行数据增强，然后利用一致性正则来优化无监督节点在不同数据增强下的预测结果。该方法减轻了过平滑和过拟合问题，相比于其他GNN模型泛化能力更强。

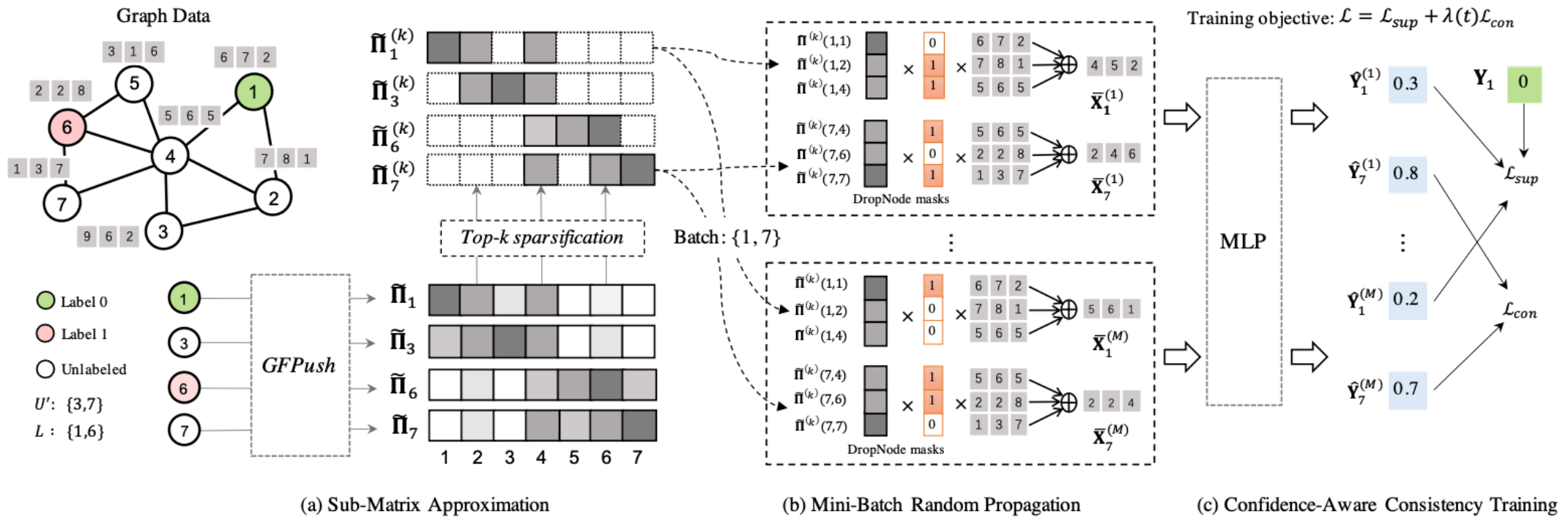


[1] Wenzheng Feng, Jie Zhang, et al. Graph Random Neural Networks for Semi-Supervised Learning on Graphs. In NeurIPS’20.

图半监督学习技术

可扩展的图随机神经网络GRAND+(WWW'22)

虽然GRAND在预测精度上取得了大幅提升，但是面临着扩展性差的问题。因此我们基于GRAND进一步提出GRAND+，分别在灵活性，可拓展性，和有效性方面进行了增强。

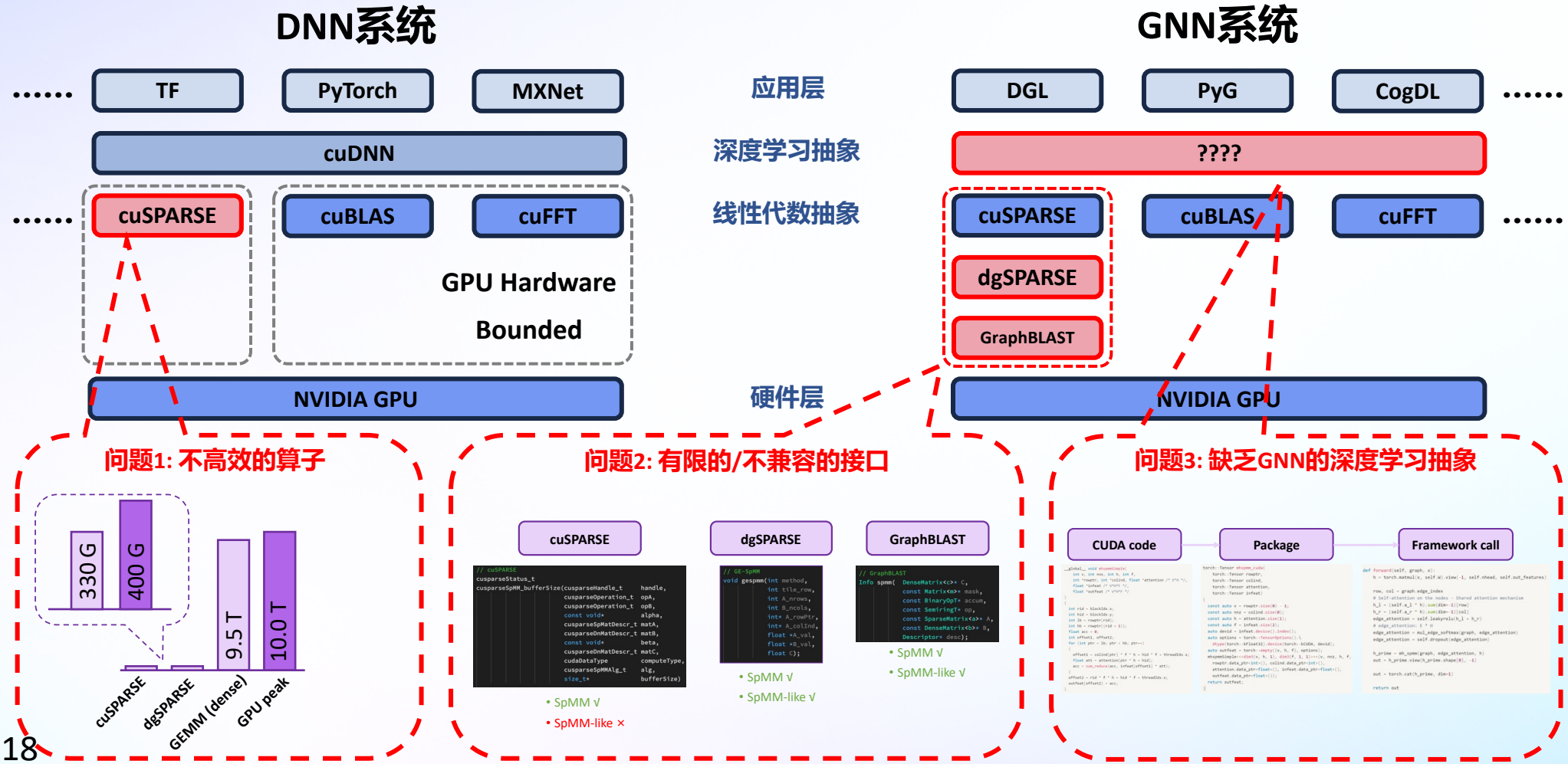


[1] Wenzheng Feng, et al. GRAND+: Scalable Graph Random Neural Networks. In WWW'22.

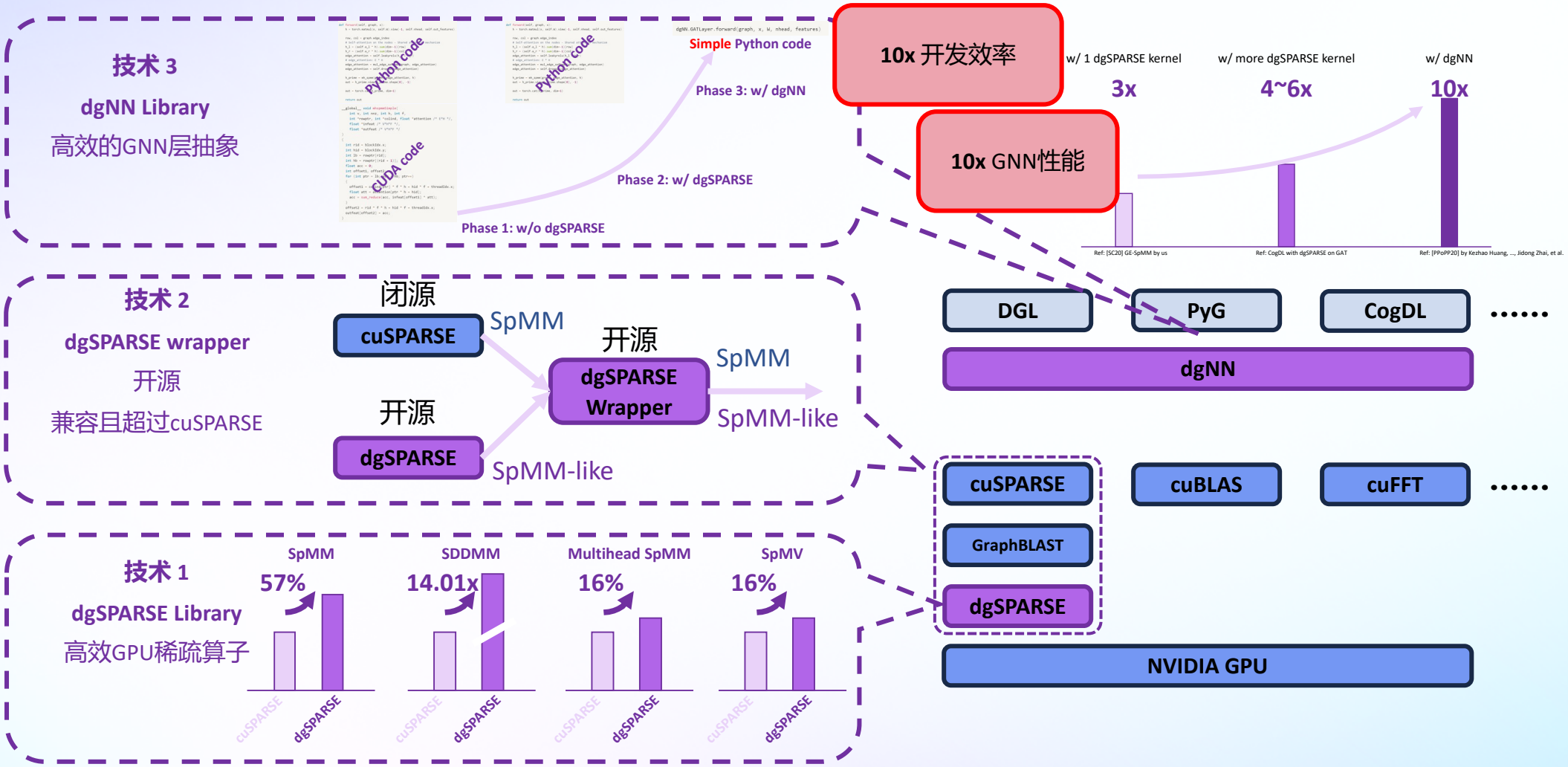
04

CogDL后端支持

现有GNN系统存在的问题

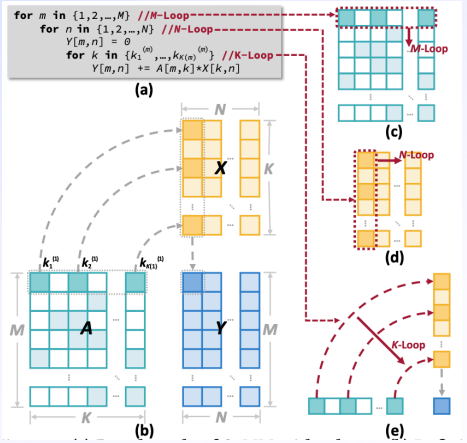


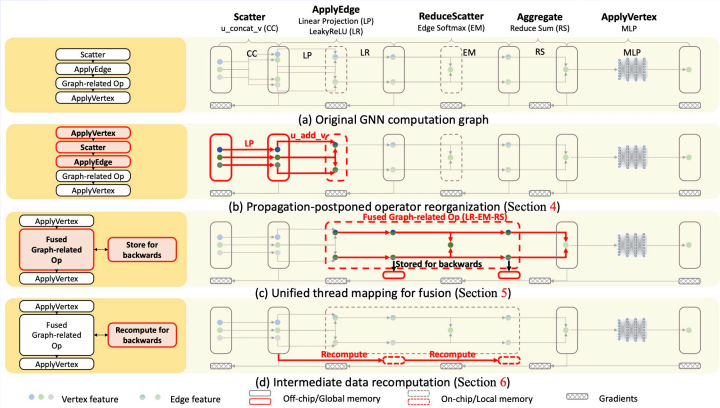
解决方案：dgSPARSE (Deep Graph SPARSE)

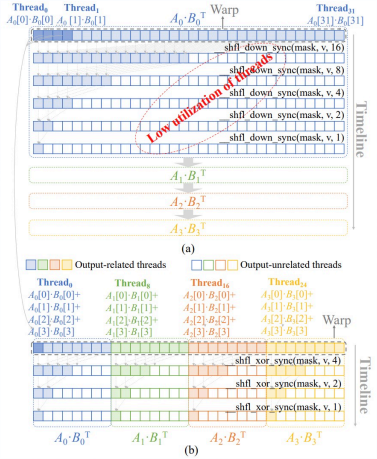


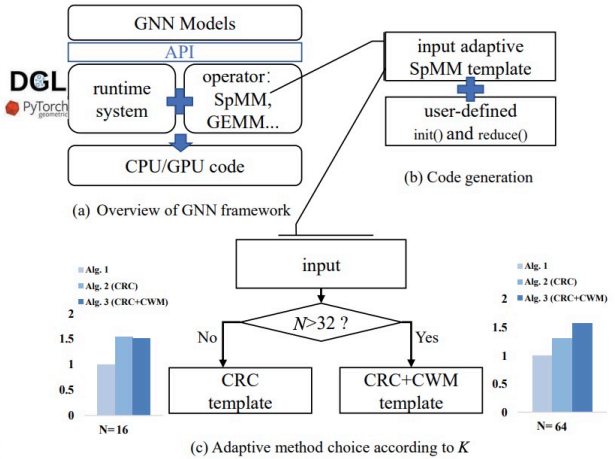
解决方案：dgSPARSE (Deep Graph SPARSE)

清华大学NICS实验室提出dgSPARSE项目（项目负责人：戴国浩博士后，汪玉教授）









……

[DAC 22]

输入数据特征自适应

相对NVIDIA后验最好方案平均加速26%~37%

[MLSys 22]

GNN计算图优化

相对DGL加速最多2.75x，内存节约最多7.75x

[ICCD 21]

高效并行规约

相对NVIDIA平均加速14x

[SC 20]

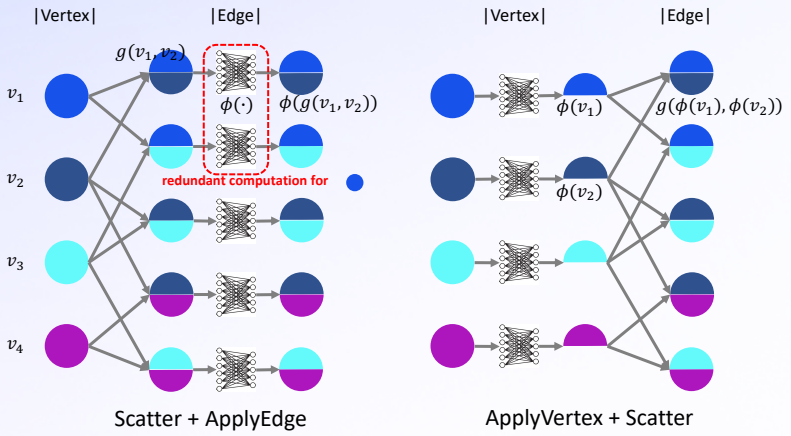
全局访存带宽优化

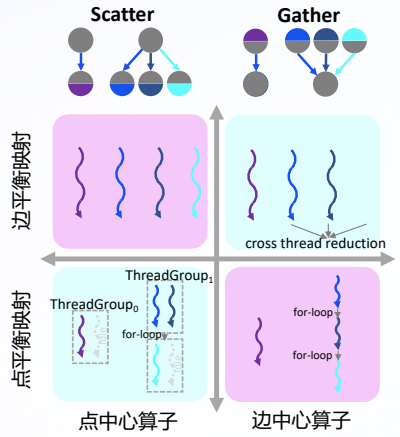
相对NVIDIA平均加速41%

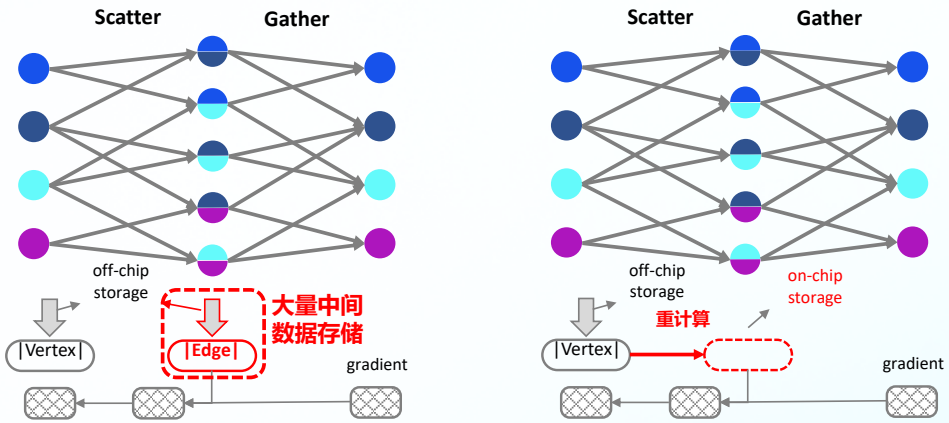
- [DAC 22] Dai, G., Huang, G., Yang, S., Yu, Z., Zhang, H., Ding, Y., Wang, Y., Xie, Y., Yang, H. Heuristic Adaptability to Input Dynamics for Sparse Matrix-Vector/Matrix Multiplication on GPUs. To appear in Design Automation Conference (DAC), 2022.
- [MLSys 22] Zhang, H., Yu, Z., Dai, G., Huang, G., Ding, Y., Xie, Y., & Wang, Y. Understanding GNN Computational Graph: A Coordinated Computation, IO, and Memory Perspective. To appear in Fifth Conference on Machine Learning and Systems (MLSys), 2022.
- [ACM-SRC 21] Huang, G., Dai, G., Ding, Y., Wang, Y., Xie, Y. Efficient Sparse Matrix Kernels based on Adaptive Workload-Balancing and Parallel-Reduction. In ACM Student Research Competition (ACM-SRC), 2021.
- [ICCD 21] Yu, Z., Dai, G., Huang, G., Wang, Y., & Yang, H. Exploiting Online Locality and Reduction Parallelism for Sampled Dense Matrix Multiplication on GPUs. In International Conference on Computer Design (ICCD), pp. 567-574, 2021.
- [MICRO-SRC 20] Huang, G., Dai, G., Wang, Y., & Yang, H. Towards Fast Graph Neural Network Training with Efficient and Framework-Compatible Sparse-Dense Matrix Multiplication. In MICRO-53 Student Research Competition (MICRO-SRC), 2020.
- [SC 20] Huang, G., Dai, G., Wang, Y., & Yang, H. GE-SpMM: General-purpose Sparse Matrix-Matrix Multiplication on GPUs for Graph Neural Networks. In International Conference for High Performance Computing, Networking, Storage and Analysis (SC), pp. 1-12, 2020.

解决方案：dgSPARSE (Deep Graph SPARSE)

例：GNN计算图优化 [MLSys 22]







训练加速2.75x

技术：算子重排

思想：优化计算图算子执行顺序

效果：训练加速最多2.75x

访存降低6.89x

技术：统一映射

思想：对算子统一映射并进行融合

效果：访存数据量降低最多6.89x

内存节约7.75x

技术：数据重算

思想：反向传播时重计算中间数据

效果：内存开销最多降低7.75x

解决方案：dgSPARSE (Deep Graph SPARSE)

较NVIDIA商用库加速1个量级，节省7倍内存（~20GB ⇒ ~2GB）

单卡GPU大规模图神经网络成为可能

基于访存规则化思想设计

使用中端工艺硬件达到高端工艺硬件更好性能

RTX 3090 (7nm, 24 GB)

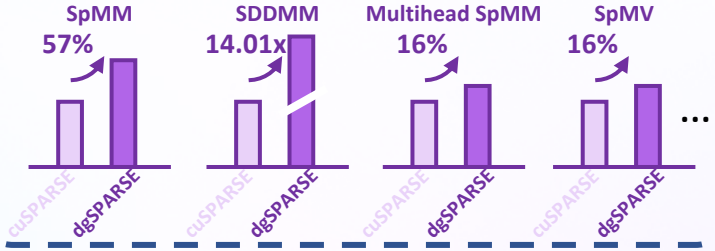
1.18倍加速

7.75倍内存节省

RTX 2080 (12nm, 8 GB)

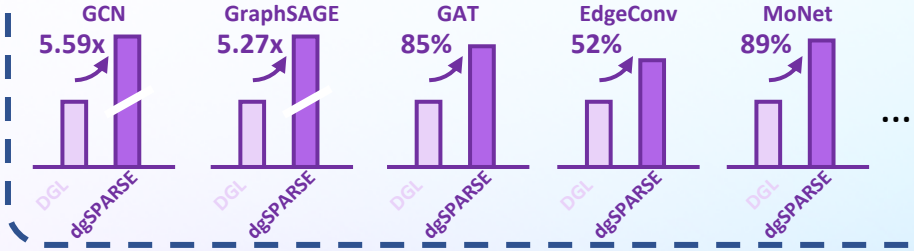
高效稀疏算子

相较NVIDIA商用库加速多达一个量级



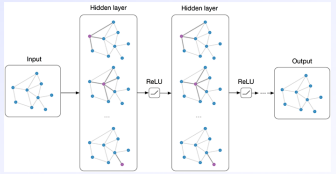
高效图神经网络计算图

相较最好开源框架加速多达5倍



解决方案：dgSPARSE (Deep Graph SPARSE)

CogDL与dgSPARSE深度配合

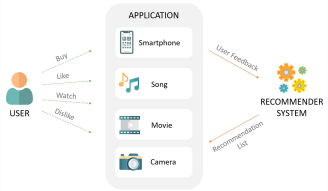


GNN研究者

- 易用实验接口
- 快速验证想法

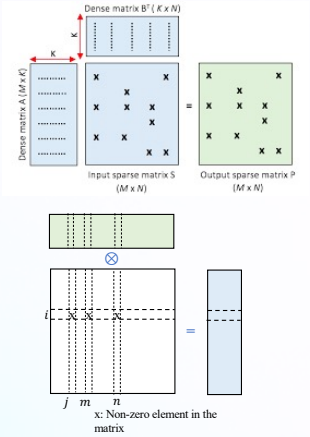
CogDL

应用开发者



- 易用应用接口
- 快速应用部署

- 更快计算速度
- 更高开发效率



- 统一算子接口
- 快速框架部署

算子开发者

dgSPARSE

- 统一稀疏接口
- 硬件稀疏支持

硬件开发者

05

CogDL生态建设

CogDL+评测基准

北京智源大会 BAAI CONFERENCE 2022.5.31-6.02

IGB (IJCAI'22)

我们发现几个常用数据集上通过超参过拟合验证集以获取其标签信息的现象可能非常严重。为了解决这个问题，我们提出了IGB数据集，每个数据集都由很多同分布子图构成，让测评更加稳定鲁棒。

GRB (NeurIPS'21 D&B)

图的对抗攻击能够影响到图机器学习模型的鲁棒性，但是缺乏一个统一和标准的评估方法。我们提出了可扩展的、通用的、统一的、可复现的图对抗鲁棒性基准，来促进不同攻击/防御方法的公平对比。

HGB (KDD'21)

我们发现某些异构图研究在实验设置、评估流程等方面存在一些问题。因此，针对异构图研究，我们提出了一个统一的评测数据集和评估流程。我们一共收集和评测了3个任务上11个广泛使用的数据集。

[1] Ziang Li, Ming Ding, Weikai Li, Zihan Wang, Ziyu Zeng, Yukuo Cen, and Jie Tang. Rethinking the Setting of Semi-supervised Learning on Graphs. In IJCAI'22.
[2] Qinkai Zheng, Xu Zou, Yuxiao Dong, Yukuo Cen, Da Yin, Jiarong Xu, Yang Yang, and Jie Tang. Graph Robustness Benchmark: Benchmarking the Adversarial Robustness of Graph Machine Learning. In NeurIPS'21 D&B.
[3] Qingsong Lv, Ming Ding, Qiang Liu, Yuxiang Chen, Wenzheng Feng, Siming He, Chang Zhou, Jian-guo Jiang, Yuxiao Dong, and Jie Tang. Are we really making much progress? Revisiting, benchmarking and refining the Heterogeneous Graph Neural Networks. In KDD'21.

CogDL+Jittor框架

Jittor的开发团队来自清华大学计算机系的图形学实验室（实验室负责人：胡事民教授）

计图（Jittor）

一个完全基于动态编译（Just-in-time），内部使用创新的元算子和统一计算图的深度学习框架。元算子和Numpy一样易于使用，并且超越Numpy能够实现更复杂更高效的操作。而统一计算图则是融合了静态计算图和动态计算图的诸多优点，在易于使用的同时，提供高性能的优化。基于元算子开发的深度学习模型，可以被计图实时的自动优化并且运行在指定的硬件上，如CPU，GPU，TPU。

CogDL+Jittor

CogDL可以提供基于Jittor框架的GNN层（例如GCNLayer），方便Jittor用户搭建相应的GNN模型。
目前的调用方式为：from cogdl.layers.jittor import GCNLayer。

Jittor 计图

https://cg.cs.tsinghua.edu.cn/jittor/

北京智源大会 BAAI CONFERENCE 2022.5.31-6.02

谢谢！

欢迎大家继续关注我们！

https://github.com/THUDM/cogdl

CogDL交流QQ群：792564770

CogDL主页: cogdl.ai