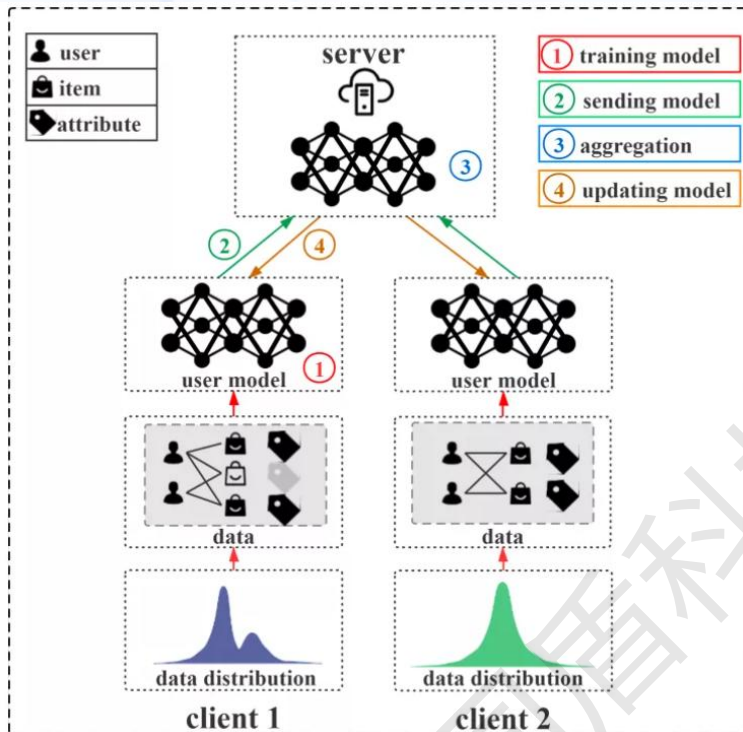


图联邦场景调研

分享人：虞晓阳



		场景任务	特点
GCFL	Federated Graph Classification over Non-IID Graphs	跨样本图分类	对比FedAvg, GCFL先进行分组, 各组内FedAvg; GCFL+采用序列DTW, 优化组内完全图边权重计算方式
FedSage	Subgraph Federated Learning with Missing Neighbor Generation	跨样本图节点分类	在FedAvg+GraphSage基础上, 本地GNN模型增加NeighGen模型补全缺失节点/边
FedGNN	Federated Graph Neural Network for Privacy-Preserving Recommendation	跨样本图推荐	通过item加密中心服务器得到邻域用户, 本地GNN模型, 引入LDP,pseudo int items隐私保护
CNFGNN	Cross-Node Federated Graph Neural Network for Spatio-Temporal Data Modeling	跨样本时空序列建模	各参与方本地M-N时序数据, 同时在空间上组成图谱 各参与方本地encoder-decoder模型(时序建模), 中心GNN模型(空间建模)



Algorithm 1 FederatedAveraging. The K clients are indexed by k ; B is the local minibatch size, E is the number of local epochs, and η is the learning rate.

Server executes:

```

initialize  $w_0$ 
for each round  $t = 1, 2, \dots$  do
     $m \leftarrow \max(C \cdot K, 1)$ 
     $S_t \leftarrow$  (random set of  $m$  clients)
    for each client  $k \in S_t$  in parallel do
         $w_{t+1}^k \leftarrow \text{ClientUpdate}(k, w_t)$ 
     $w_{t+1} \leftarrow \sum_{k=1}^K \frac{n_k}{n} w_{t+1}^k$ 
  
```

ClientUpdate(k, w): // Run on client k

```

 $\mathcal{B} \leftarrow$  (split  $\mathcal{P}_k$  into batches of size  $B$ )
for each local epoch  $i$  from 1 to  $E$  do
    for batch  $b \in \mathcal{B}$  do
         $w \leftarrow w - \eta \nabla \ell(w; b)$ 
    return  $w$  to server
  
```

两大挑战:

数据异构 (主要是用户间数据Non-IID)

系统异构 (设备间通信和计算能力的差异)

	特点	备注
FedProx	通过加入 proximal term 修正项, 提高了整体收敛的稳定性。 通过对本地设备动态调整迭代轮数, 保证了对系统异构的容忍性。	$\min_w h_k(w; w^t) = F_k(w) + \frac{\mu}{2} \ w - w^t\ ^2$ 使得本地更新不要太过远离初始 global model
LG-FedAvg	模型: Local + global global部分进行FedAvg - 通信参数量更小 - 底层参数是参与方个性化的	
ASFGNN	把节点类别信息加密发送到中心, 统计总的分布, 计算各参与方与中心数据分布JS散度值, 差异化更新参数	<pre> for $i \in \mathcal{P}$ <i>in parallel</i> do $js_t^i \leftarrow JS(Q_t^i Q_t^s)$ $\overline{W}_{l,t}^i \leftarrow js_t^i \cdot \overline{W}_{l,t}^i + (1 - js_t^i) \cdot W_{l,t}^s$ </pre>
FedAMP	会让权重的相似度分布比较接近的贡献会更大一些, 然后不那么相似的话, 那么参与方之间的模型的贡献就会稍小一些。	$\mathbf{u}_i^k = \left(1 - \alpha_k \sum_{j \neq i}^m A' \left(\ \mathbf{w}_i^{k-1} - \mathbf{w}_j^{k-1}\ ^2 \right) \right) \cdot \mathbf{w}_i^{k-1}$ $+ \alpha_k \sum_{j \neq i}^m A' \left(\ \mathbf{w}_i^{k-1} - \mathbf{w}_j^{k-1}\ ^2 \right) \cdot \mathbf{w}_j^{k-1}$ $= \xi_{i,1} \mathbf{w}_1^{k-1} + \dots + \xi_{i,m} \mathbf{w}_m^{k-1},$

联邦场景:

跨样本图分类联邦, 对比FedAvg, GCFL先进行分组, 各组内FedAvg; GCFL+采用序列DTW, 优化组内完全图边权重计算方式。

直观方案 (GNN+FedAvg) :

存在的问题:

各client图结构, 节点特征存在异质性, 可以认为是Non IID数据;

§ 图结构异质:

使得模型难以学习到不同client之间重要的相似图模式

§ 节点特征异质:

使得模型难以学习到在不同客户端之间具有泛化能力的消息传递机制

结果:

直接FedAvg, 结果可能适得其反

主要思想:

联合寻找具有相似结构和特征的图所在的客户端组成簇，并在同一簇内的客户端之间使用**FedAvg**训练图模型。

步骤:

基于**自顶向下的bi-partitioning**机制，在第 t 轮时基本流程如下：

- 中心服务器接收客户端簇 $\{\mathbb{C}_1, \mathbb{C}_2, \dots, \mathbb{C}_m\}$ 的 m 个梯度集合 $\{\{\Delta\theta_{i1}\}, \{\Delta\theta_{i2}\}, \dots, \{\Delta\theta_{im}\}\}$;
- 对于**满足式4和5**的簇计算其**cluster-wise**余弦相似度矩阵，该矩阵用于构建该簇内所有客户端间的全连接图(图中节点为客户端);
- 在上述全连接图上运用Stoer-Wagner最小割算法，将簇 $\mathbb{C}_k$ 划分为 $\{\mathbb{C}_{k1}, \mathbb{C}_{k2}\}$ 。

分割条件:

$$\delta_{mean} = \left\| \sum_{i \in [n]} \frac{|\mathcal{G}_i|}{|\mathcal{G}|} \Delta\theta_i \right\| < \varepsilon_1.$$

$$\delta_{max} = \max(\|\Delta\theta_i\|) > \varepsilon_2 > 0.$$

出发点:

GCFL存在两个问题: **(1)**每一轮聚合的梯度范数波动很大, **(2)**不同客户端的梯度范数数值范围相差很大, 表明不同客户端梯度更新的方向以及大小是发散的, 使得**GCFL**的聚合操作容易被噪声误导或忽略更重要的客户端梯度信息。

改进:

- 在GCFL基础上, 中心服务器维护一个多变量时序(multi-variant time-series)矩阵 $\mathbf{Q} \in \mathbb{R}^{(n,d)}$, 其中为 n 客户端数量, d 为当前维护的梯度序列长度。在每一轮 t 中, 中心服务器移除序列头部的梯度信息, 将刚到达的梯度信息 $\|\Delta\theta_{i1}\|$ 插入序列末尾 $\mathbf{Q}(i,:) \in \mathbb{R}^d$ 。
- GCFL+沿用GCFL的聚类标准(即式4和式5), 中央服务器对满足标准的簇计算距离矩阵 β , 该矩阵中每个元素为簇中客户端对应梯度序列的DTW(dynamic time warping)距离, 计算公式如下:

$$\beta_k(p, q) = \text{dist}(Q(p,:), Q(q:)), p, q \in \text{idx}(\{\mathbf{S}_i\})$$

其中 $\text{idx}(\{\mathbf{S}_i\})$ 为簇中 $\mathbf{C}_k$ 所有客户端 $\{\mathbf{S}_i\}$ 的索引。

- GCFL+根据矩阵 β 对簇进行bi-partitioning。

Table 3: Performance on the single-dataset-multi-client setting. We present the average accuracy and minimum gain over self-train on all clients, as well as the ratio of clients which get improved.

Dataset (# clients)	NCI1 (30)			PROTEINS (10)			IMDB-BINARY (10)		
Accuracy	average	min gain	ratio	average	min gain	ratio	average	min gain	ratio
self-train	0.6468(± 0.053)	—	—	0.7213(± 0.058)	—	—	0.7654(± 0.057)	—	—
FedAvg	0.6474(± 0.076)	-0.1333	14/30	0.7490(± 0.034)	-0.0615	6/10	0.7596(± 0.049)	-0.0800	5/10
FedProx	0.6437(± 0.072)	-0.2400	16/30	0.7556(± 0.036)	-0.0923	7/10	0.7746(± 0.048)	-0.0600	6/10
GCFL	0.7326(± 0.052)	-0.0462	26/30	0.7739(± 0.043)	-0.0545	8/10	0.8256(± 0.059)	0.0182	10/10
GCFL+	0.7422 (± 0.053)	-0.1143	28/30	0.7776 (± 0.037)	-0.0154	9/10	0.8299 (± 0.052)	0.0167	10/10

Table 4: Performance on the multi-dataset-multi-client setting. Metrics are the same as Table 3.

Dataset (# domains)	MOLECULES (1)			BIOCHEM (2)			MIX (3)		
Accuracy	average	min gain	ratio	average	min gain	ratio	average	min gain	ratio
self-train	0.7543(± 0.017)	—	—	0.7129(± 0.016)	—	—	0.7001(± 0.034)	—	—
FedAvg	0.7524(± 0.026)	-0.0132	3/7	0.6944(± 0.027)	-0.1467	4/10	0.6886(± 0.023)	-0.1233	5/13
FedProx	0.7668(± 0.032)	-0.0054	5/7	0.7053(± 0.026)	-0.1000	5/10	0.6897(± 0.026)	-0.1367	5/13
GCFL	0.7661(± 0.016)	0.0010	7/7	0.7172(± 0.019)	-0.0700	7/10	0.7056(± 0.019)	-0.1400	10/13
GCFL+	0.7745 (± 0.030)	0.0010	7/7	0.7312 (± 0.031)	-0.0300	8/10	0.7121 (± 0.021)	-0.0233	10/13

联邦场景:

跨样本节点分类联邦，在GraphSage+FedAvg基础上，增加NeighGen补全缺失节点/边

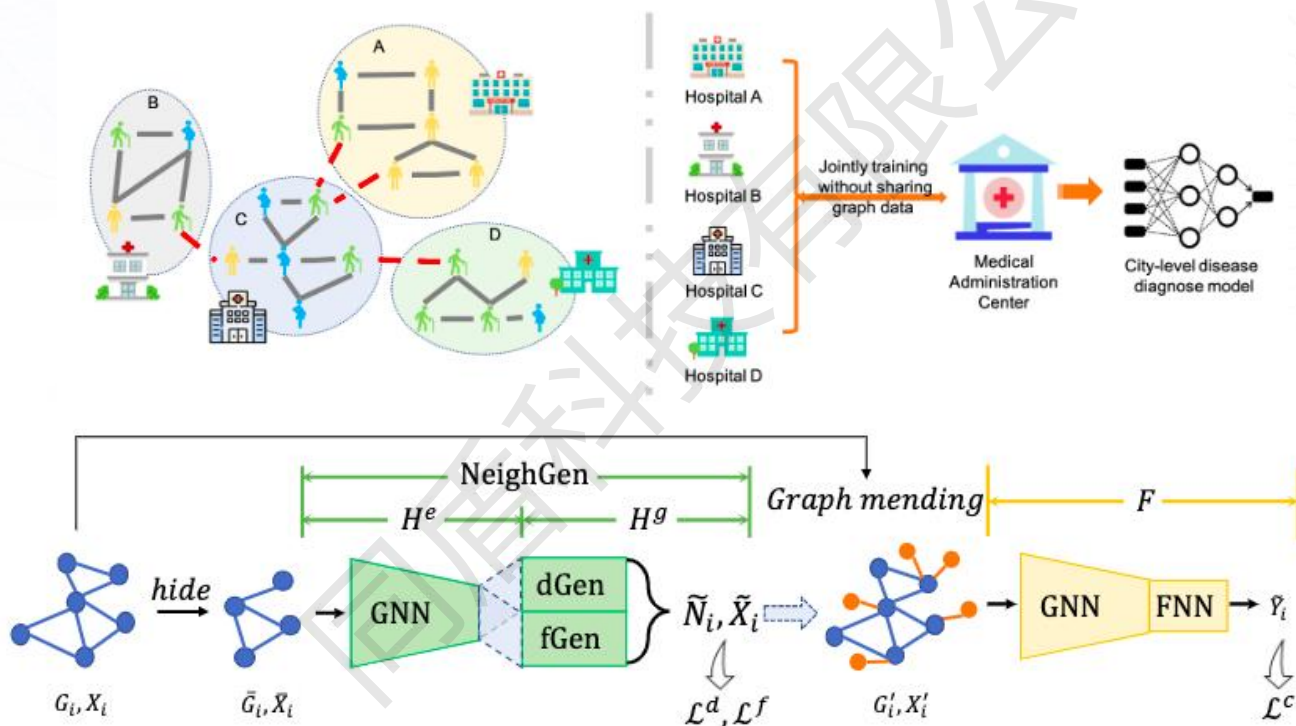


Figure 2: Joint training of missing neighbor generation and node classification.

联邦场景:

跨样本图推荐, 通过item加密中心服务器得到邻域用户, 本地GNN模型, 引入LDP,pseudo int items 隐私保护

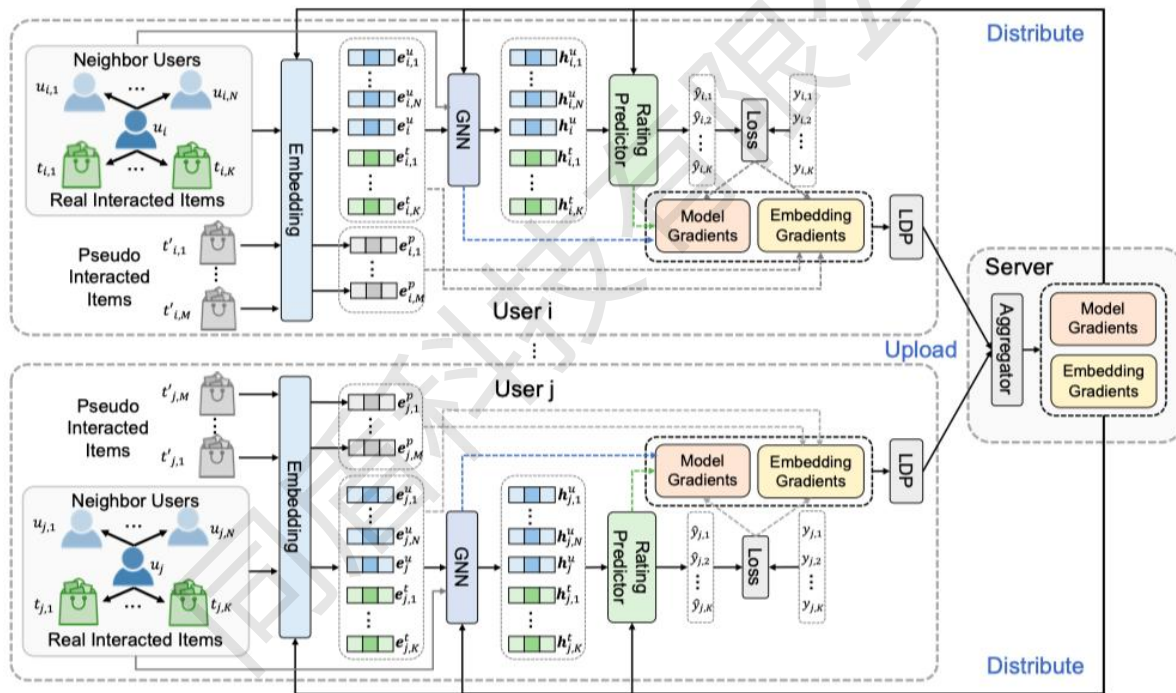


Figure 2: The framework of our FedGNN approach.

Privacy Preserving Graph Expansion:

原只有一阶，增加领域用户，高阶关系

Pseudo Interacted items:

Gaussian distribution 实际item梯度的均值/
方差

LDP:

$$g_i = clip(g_i, \delta) + Laplace(0, \lambda),$$

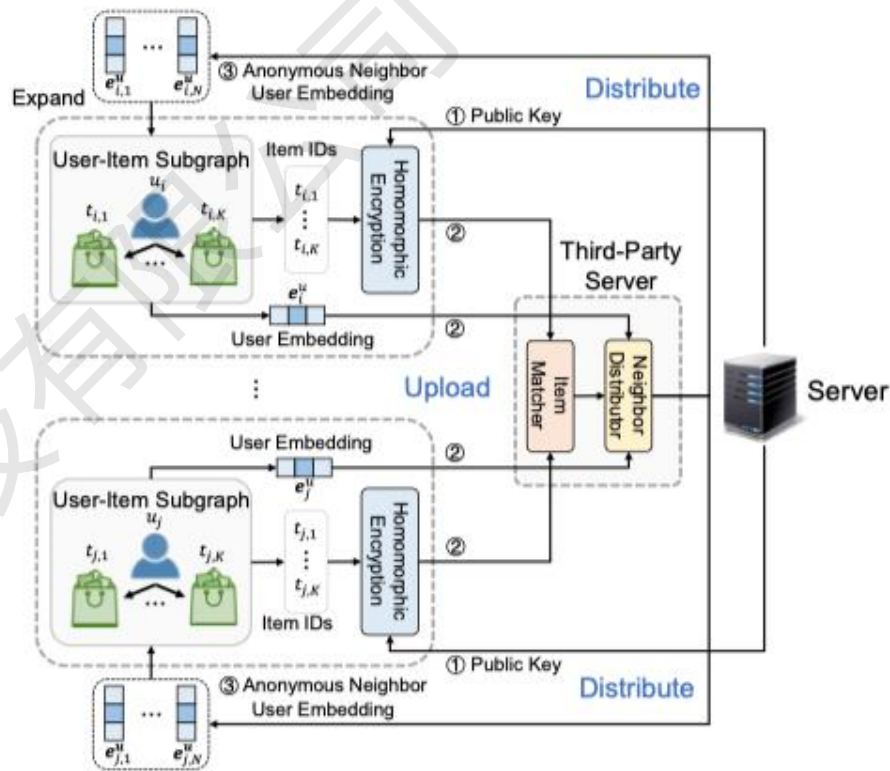
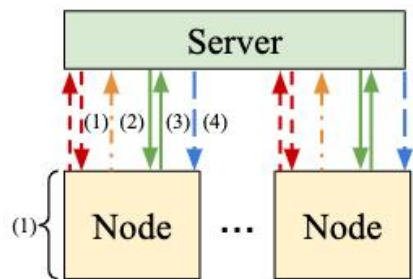


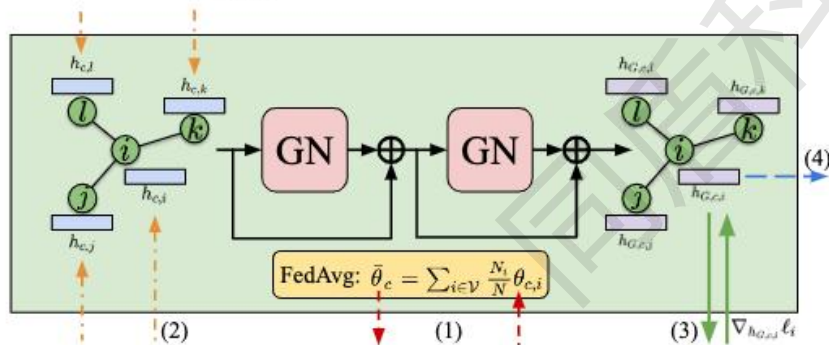
Figure 3: The framework of the privacy-preserving user-item graph expansion method.

联邦场景:

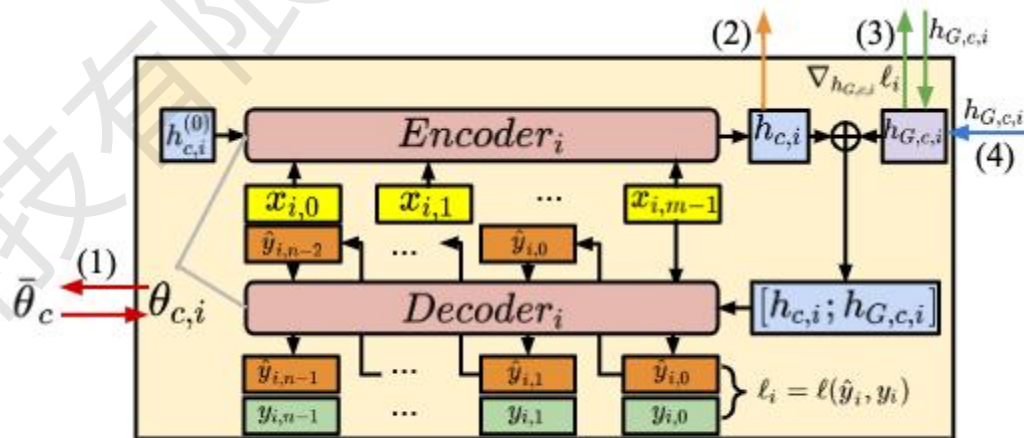
跨样本时空序列建模, 各参与方本地encoder-decoder模型 (时序建模), 中心GNN模型 (空间建模)



(a) Overview of the training procedure.



(b) Server-side Graph Network (GN).



(c) Encoder-decoder on the i -th node.



T H A N K S