纵向联邦　前置知识

丸一口

Content

relevant subtitle in this line

1

场景

你不干！有的是帕鲁干！

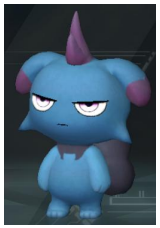
引子

数据格式为（身高/体重,[患有脂肪肝]）

捣蛋猫

(165cm,False)
(164cm,False)
(155cm,True)

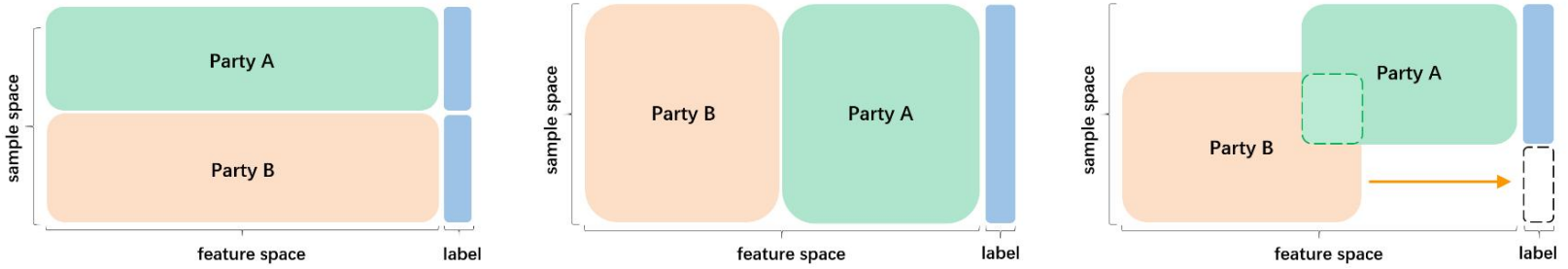
瞅什魔



趴趴鲶

(40kg)
(41kg)
(100kg)

数据格式对比



(a) Horizontal Federated Learning (b) Vertical Federated Learning (c) Federated Transfer Learning

Figure 1: Three categories of Federated Learning

横向联邦：(身高，体重，脂肪肝)

- 流浪商人手上：  (165cm,40kg,False)  (164cm,41kg,False)
- 黑市商人手上：  (155cm,100kg,True)

[1]Liu Y, Kang Y, Zou T, et al. Vertical Federated Learning: Concepts, Advances, and Challenges[J]. IEEE Transactions on Knowledge and Data Engineering, 2024.

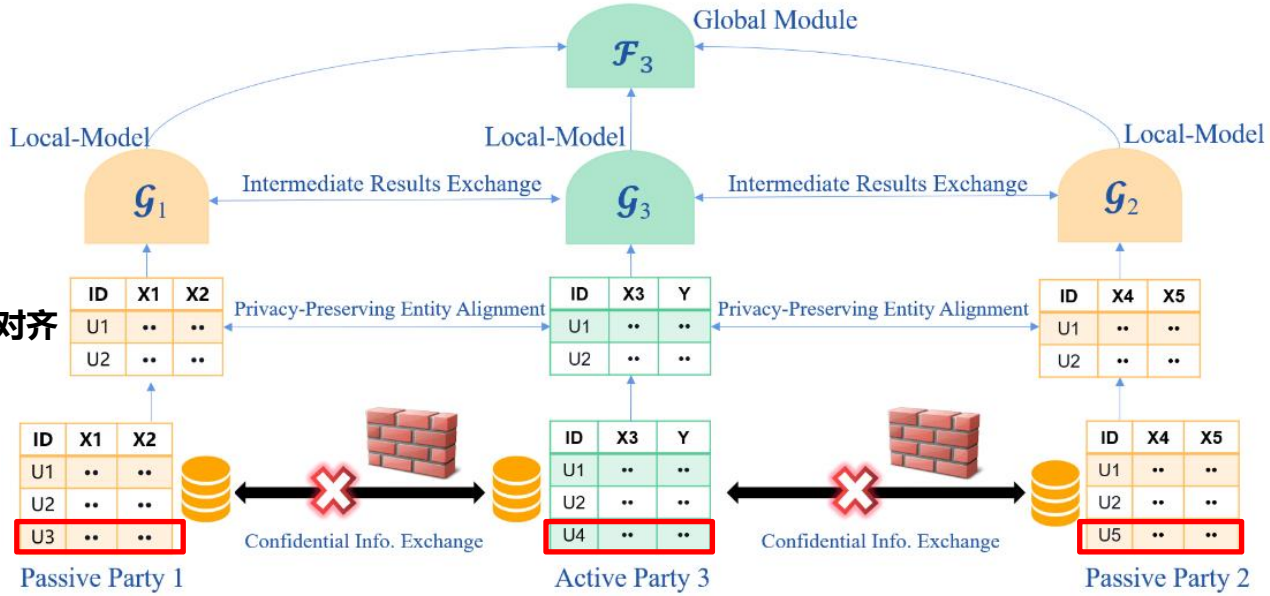
2

算法

帕鲁脂肪肝的预测与研究：
帕鲁商人(仅有尺)与黑市商人(仅有秤)联合训练

VFL

数据对齐



符号	含义
N	样本数为 N
K	K 个参与方
H_k	本地模型输出 $H_k = \mathcal{G}_i(\mathbf{x}_k, \theta_k)$
θ_k	被动方模型
ψ_K	主动方模型
$\mathcal{F}_K$	global module $\mathcal{F}_K$
$\mathcal{G}_i$	local model $\mathcal{G}_i$ (我觉得叫module好一点)
$\mathbf{x}, \mathbf{y}$	$\mathcal{D} \triangleq \{(\mathbf{x}_i, y_i)\}_{i=1}^N$

VFL

$$\nabla_{\theta_k}\ell = \frac{\partial \ell}{\partial \theta_k} = \sum_i \frac{\partial \ell}{\partial H_{i,k}} \frac{\partial H_{i,k}}{\partial \theta_k}$$

Algorithm 1 A General VFL Training Procedure.

Input: learning rates η_1 and η_2

Output: Model parameters θ_1, θ_2 ... θ_K, ψ_K

Party 1,2,...,K, initialize θ_1, θ_2, ... θ_K, ψ_K.
for each iteration $j = 1, 2, ...$ **do**
  Randomly sample a mini-batch of samples $\mathbf{x} \subset \mathcal{D}$
  for each party k=1,2,...,K in parallel **do**
    Party k computes $H_k = \mathcal{G}_k(\mathbf{x}_k, \theta_k)$;
    Party k sends $\{H_k\}$ to party K;
  end for
  Active party K updates $\psi_K^{j+1} = \psi_K^j - \eta_1 \frac{\partial \ell}{\partial \psi_K}$;
  Active party K computes and sends $\frac{\partial \ell}{\partial H_k}$ to all other parties;
  for each party k=1,2,...,K in parallel **do**
    Party k computes $\nabla_{\theta_k}\ell$ with Equation (3);
    Party k updates $\theta_k^{j+1} = \theta_k^j - \eta_2 \nabla_{\theta_k}\ell$;
  end for
end for

3

疑问

一场误会：三个我阅读过程中产生的疑问，和解答

Q&A：一场误会

Algorithm 1 A General VFL Training Procedure.

Input: learning rates η_1 and η_2

Output: Model parameters θ_1, θ_2 ... θ_K, ψ_K

1: Party 1,2,...,K, initialize θ_1, θ_2, ... θ_K, ψ_K.
2: **for** each iteration $j = 1, 2, ...$ **do**
3: Randomly sample a mini-batch of samples $\mathbf{x} \subset \mathcal{D}$
4: **for** each party k=1,2,...,K in parallel **do**
5: Party k computes $H_k = \mathcal{G}_k(\mathbf{x}_k, \theta_k)$;
6: Party k sends $\{H_k\}$ to party K;
7: **end for**
8: Active party K updates $\psi_K^{j+1} = \psi_K^j - \eta_1 \frac{\partial \ell}{\partial \psi_K}$;
9: Active party K computes and sends $\frac{\partial \ell}{\partial H_k}$ to all other parties;
10: **for** each party k=1,2,...,K in parallel **do**
11: Party k computes $\nabla_{\theta_k} \ell$ with Equation (3);
12: Party k updates $\theta_k^{j+1} = \theta_k^j - \eta_2 \nabla_{\theta_k} \ell$;
13: **end for**
14: **end for**

1.第9行这个求导，有预测值，还要知道label才能求啊?

- Label可以是共享的(我一直以为Label是所有人都有的)
- 可以是只有活跃方拥有的(本篇的场景)

2.被动方有label的话，为啥不在本地自己做反向传播？(可能不存在这个问题哈)

- 因为单个Party的数据维度不够，或者说想吃别的client手上的维度

3.联邦学习中的最重要的问题，某个client怎么吃到别的client的数据?

- 这里下发的是 $\frac{\partial l}{\partial H_k}$，这里，$\partial l$是用完整维度的数据计算的损失，包含了所有client的数据维度

谢谢~

丸一口 2024.2