arXiv:2303.04340v1 [cs.LG] 8 Mar 2023

面向网联自动驾驶汽车的隐私保护与不确定性感知联邦轨迹预测

预印本

Muzi Peng电气与计算机工程系 东北大学
波士顿，MA 02115
peng.mu@northeastern.edu

Jiangwei Wang电气与计算机工程系 康涅狄格大学 斯托尔斯，CT 06268
jiangwei.wang@uconn.edu

Dongjin Song计算机科学与工程系 康涅狄格大学 斯托尔斯，CT 06268
dongjin.song@uconn.edu

Fei Miao计算机科学与工程系 康涅狄格大学 斯托尔斯，CT 06268
fei.miao@uconn.edu

Lili Su东北大学电气与计算机工程系，马萨诸塞州波士顿，邮编 02115，
l.su@northeastern.edu

2023年3月9日

摘要

深度学习是自动驾驶车辆轨迹预测的首选方法。然而，其对数据的旺盛需求意味着必须拥有足够丰富且高质量的集中式数据集，这很容易导致隐私泄露。此外，对于安全关键型信息物理系统而言，不确定性感知日益重要，因为其预测模块严重依赖机器学习工具。本文通过在网联自动驾驶汽车上采用联邦学习，并使用不确定性感知全局目标，降低了数据收集要求并增强了不确定性感知。我们将该算法命名为 FLTP。我们进一步提出 ALFLTP，通过运用主动学习技术，自适应地选择参与客户端，从而提升 FLTP。我们考虑负对数似然（NLL）和偶然不确定性（AU）这两种不同的客户端选择指标。在 Argoverse 数据集上的实验表明，FLTP 的表现显著优于使用本地数据训练的模型。此外，在大多数轮次中，ALFLTP-AU 的回归损失收敛更快，在 NLL、最小平均位移误差和漏检率方面的表现也优于 FLTP；与 ALFLTP-NLL 相比，其逐轮表现更加稳定。

1 引言

准确预测周围物体的轨迹对自动驾驶至关重要。例如，为了避免碰撞，准确预测邻近车辆的并道或超车行为十分重要。近年来，深度学习已成为轨迹预测的首选方法 [1, 2, 3, 4, 5, 6]。据我们所知，由于深度学习对数据的需求量很大，现有的大多数方法都默认可以获取足够丰富且高质量的集中式数据集 [6, 4, 7, 8, 9, 10, 11, 12]。这一要求很容易导致隐私泄露

因为原始轨迹数据包含敏感信息，例如个人信息（家庭和公司地址以及驾驶日志）和车辆信息（车辆类型、品牌和外观） [13]。随着跨城市和/或国家获取有用信息的需求不断增加 [14] ，隐私威胁也会迅速加剧。

对于自动驾驶汽车等安全关键型信息物理系统而言，不确定性感知变得日益重要，因为其预测模块高度依赖机器学习工具 [15, 16, 17, 3, 18, 19, 20]。一般来说，不确定性主要分为两种。数据不确定性（偶然不确定性）描述的是内在且不可约的数据变异性。模型不确定性（认知不确定性）指观测模型的局限性，这些模型可用于学习潜在的真实机制。在轨迹预测领域，偶然不确定性通常通过模型输出中包含的方差估计来近似 [21, 3, 6, 18], ，而认知不确定性通常通过蒙特卡洛（MC）随机失活 [22, 3] 和集成模型 [23, 20]来建模。然而，利用中间不确定性量化进一步改进轨迹预测训练的潜力，在很大程度上仍未得到重视。

联邦学习（FL）是一种快速发展的隐私保护型去中心化学习框架，其中参数服务器（PS）与一组客户端协作训练一个共同模型 [24, 25]。在 FL 中，客户端不会将数据上传至 PS，而是基于各自的本地数据进行更新，并定期向 PS 报告本地更新。随后，PS 对这些更新进行有效聚合，以获得一个精细化模型，并将其广播给客户端，以便进一步更新模型。在网联自动驾驶汽车（CAVs）应用中，联邦学习可作如下理解：每辆自动驾驶汽车都是一个客户端，负责收集其驾驶场景的本地数据，而参数服务器则可视为一个计算中心。

我们的贡献可总结如下：
(1) 我们将联邦学习应用于网联自动驾驶汽车，使其能够协作训练 HiVT [6] ——一种轻量级 Transformer，从而放宽了对原始数据收集的要求；我们将由此得到的算法命名为基于联邦学习的轨迹预测（*FLTP*）。为实现不确定性量化，我们遵循已有研究，采用拉普拉斯混合分布的常用负对数似然（NLL）作为回归损失，并分别使用位置参数和尺度参数来解码预测轨迹及其对应的偶然不确定性。据我们所知，我们是首个将 FL 应用于 CAVs 协作轨迹预测的团队。

(2) 普遍观察到， [24, 25] （我们的初步实验结果也验证了这一点），客户端部分参与能够加快收敛并提高准确率。为进一步提升 FLTP 的性能，我们引入了 *ALFLTP* ，它采用新颖的主动学习技术，在每轮中谨慎选择参与客户端。我们分别将负对数似然（NLL）和偶然不确定性（AU）作为客户端选择指标。据我们所知，我们是首个考虑使用偶然不确定性作为客户端选择指标的研究。

(3) 在 Argoverse 数据集上进行的实验表明，FLTP 的表现显著优于使用本地数据训练的模型。此外，与 FLTP 相比，ALFLTP-AU 在训练回归损失方面收敛更快，并且在大多数轮次中，NLL、最小平均位移误差（minADE）和漏检率（MR）表现更好。与 ALFLTP-NLL 相比，它在各轮次中的表现也更加稳定。

2 相关工作

自动驾驶车辆轨迹预测。 轨迹预测流程通常包括三个子任务：输入表示、上下文聚合和输出表示。输入表示通常通过栅格化 [1, 2, 3] 或向量化 [4, 5, 6]来构建。上下文聚合模块用于捕捉交通中的对象交互，例如车辆与车辆、车辆与车道以及车辆与行人之间的交互；常用技术包括社会池化 [21, 26], 注意力机制 [7, 27, 9, 6] 或图神经网络（GNN） [28, 4, 29]。为对未来轨迹进行多模态预测，输出表示通常依赖于基于回归的方法 [21, 4, 6] 和基于提案的方法 [30, 31, 32, 9]等方法。

用于轨迹预测的联邦学习。 FedAvg 是首个也是应用最广泛的联邦学习算法[24, 25]。尽管联邦学习在分布式信息处理领域前景广阔，但现有研究中只有少数将联邦学习应用于与自动驾驶车辆轨迹预测相关的任务。Flow-FL [33] 研究了联网机器人团队的轨迹预测。ATPFL [34] 结合自动化机器学习与联邦学习，自动设计人类轨迹预测模型。据我们所知，目前尚无研究将联邦学习应用于网联自动驾驶汽车的轨迹预测。

联邦学习中的主动客户端选择。主动学习的核心思想是识别对模型训练更具信息价值的数据样本。受主动学习启发，已有少数研究设计了主动客户端选择策略，并证明这种有偏客户端选择能够加快收敛 [35, 36, 37] 。具体而言，[35, 36] 将本地损失作为主动学习指标，并优先选择主动学习指标较高的客户端。 [37] 采用贝叶斯主动学习，并将模型不确定性作为指标。然而，现有的客户端选择策略均未将偶然不确定性用作指标。

3 方法

3.1 问题定义

如图 1a 所示，驾驶场景数据（即一个数据样本）可用三元组 $S=(X,Y,\mathscr{M})$ 表示，其中 X 和 Y 分别是场景中各智能体（智能体可以是车辆或行人）的观测轨迹和未来轨迹集合，$\mathscr{M}$ 表示地图信息。令 m 表示场景中的智能体数量，则 X 和 Y 可分别表示为 $X=\{x1,...,xm\}$ 和 $Y=\{y1,...,ym\}$，其中 $x_i\in\mathbb{R}^{2\times T_{obs}}$ 和 $y_i\in\mathbb{R}^{2\times T_{pre}}$ 分别是智能体 i 的二维观测轨迹坐标和未来轨迹坐标，其长度分别为 T_{obs} 和 T^{pre}。特别地，对于任意给定的场景 S，在 m 个智能体中，有一个目标智能体，记为 i^*，是自车最关注的对象。在每个场景中，自车旨在利用 X 和 $\mathscr{M}$ 预测目标车辆的未来轨迹。

系统由一台服务器和 C 辆自动驾驶汽车组成，每辆车都可以使用激光雷达传感器和摄像头采集驾驶场景数据，以记录所有相邻车辆的轨迹。我们将每辆自动驾驶汽车称为一辆自车，它在我们的联邦学习框架中充当一个客户端。每个客户端 $c\in\mathscr{C}\triangleq\{1,\cdots,C\}$ 都拥有一个大小为 K_c 的本地数据集，记为 $\mathscr{D}_c=\{S^1,S^2,...,S^{K_c}\}$。令 m_c^k 表示客户端 c 的第 k 个样本中的智能体数量。令 $K=c\sum K_{cc=1}$ 表示样本总数。

3.2 HiVT：分层向量Transformer

分层向量变换器（HiVT）是一种集中式、轻量级、基于图的运动预测模型 [6]。为提高模型在场景中智能体数量增加时的可扩展性，HiVT 将问题分解为局部上下文提取和全局交互建模。HiVT 在 Argoverse运动预测基准上取得了最先进的性能。本文重点研究联邦学习框架下的 HiVT 训练。与集中式 HiVT 不同，每个客户端上的本地模型更新仅针对其本地目标函数进行。

损失函数：由于 HiVT 只需一次前向传播即可对场景中的所有智能体进行预测，因此所有智能体的预测结果都会用于计算训练损失。不过，在推理阶段，仅评估目标智能体的预测结果。同样，在第 3.4 节中，我们仅针对每个驾驶场景中的目标智能体进行价值计算。

HiVT 将拉普拉斯混合概率密度函数作为其损失函数的一部分，其中 $\hat{\mu}_{i,t,f}\in\mathbb{R}^2$ 和 $\hat{b}_{i,t,f}\in\mathbb{R}^2$ 分别表示每个智能体 $i\in\{1,...,m\}$ 在每个预测时间步 $t\in\{1,...,T_{pre}\}$ 上、针对每个混合分量 $f\in\{1,...,F\}$ 估计的位置参数和尺度参数。对于任意固定的 i 和 t，两个估计值 $\hat{\mu}_{i,t,f}$ 和 $\hat{b}_{i,t,f}$ 分别表示第 f 个预测轨迹及其对应的不确定性。HiVT解码器还会为每个智能体 $i\in\{1,...,m\}$ 在每个混合分量 $f\in\{1,...,F\}$ 输出混合模型的预测系数 $\hat{P}_{i,f}\in[0,1]$。

HiVT 仅优化 F 条轨迹中的最佳模式。具体而言，第 i 个智能体的最佳轨迹由以下公式确定：

$$f_{best_i}=\arg\min_{f\in\{1,...,F\}}\sum_{t=1}^{T_{pre}}\|y_{i,t}-\hat{\mu}_{i,t,f}\|_2. \tag{1}$$

回归损失 是拉普拉斯分布的负对数似然（NLL），如下所示：

$$L_{reg}=\frac{1}{m}\frac{1}{T_{pre}}\sum_{i=1}^{m}\sum_{t=1}^{T_{pre}}\left[\log(2\hat{b}_{i,t,f_{best_i}})+\frac{\|y_{i,t}-\hat{\mu}_{i,t,f_{best_i}}\|_1}{\hat{b}_{i,t,f_{best_i}}}\right] \tag{2}$$

分类损失 损失 L_{cls} 是用于优化混合系数的交叉熵损失，如下所示：

$$L_{cls}=\frac{1}{m}\sum_{i=1}^{m}\sum_{f=1}^{F}-P_{i,f}\log\hat{P}_{i,f} \tag{3}$$

with

$$P_{i,f}=\frac{\exp(-\sum_{t=1}^{T_{pre}}\|\hat{\mu}_{i,t,f}-y_{i,t}\|_2)}{\sum_{j=1}^{F}\exp(-\sum_{t=1}^{T_{pre}}\|\hat{\mu}_{i,t,j}-y_{i,t}\|_2)}$$

那么，模型权重为 $S\,w$ 的场景的最终损失为：

$$L(S,w)=L_{reg}+L_{cls}. \tag{4}$$

3.3 基于联邦学习的轨迹预测（FLTP）

第 3.2 节中的损失函数是针对单个驾驶场景数据定义的。在联邦学习中，由于客户端只能访问其本地数据，因此本地目标函数定义如下：

$$F_c(\mathscr{D}_c, w) = \frac{1}{K_c} \sum_{S \in \mathscr{D}_c} L(S, w) \tag{5}$$

我们在算法1中正式描述了FLTP。它遵循FedAvg [24]通用的服务器—客户端交互方式。不同于标准FedAvg，我们使用AdamW作为本地优化器，而不是使用随机梯度下降。
具体而言，在每次全局迭代中：

- 参数服务器首先根据概率向量$[K_1/K \cdots, Kc/K]$无放回抽样，随机选择 $\lfloor f1C \rceil$ 个客户端，其中$f_1 \in (0, 1]$是作为算法输入给定的客户端采样率。例如，设$C = 3$、$f_1 = \frac{2}{3}$、$K_1 = 1$,$K_2 = 2$和$K_3 = 7$。选择客户端1的概率为$\underline{0.1} + 0.2 \times \frac{0.1}{0.8} + 0.7 \times \frac{0.1}{0.3}$。
- 随后，参数服务器将当前模型w_r发送给每个选定的客户端 $\mathscr{L}_r$，以便利用其本地数据进一步改进模型。
- 各选定客户端并行地根据式（5）运行 AdamW，使用指定的小批量大小B，在本地数据集上训练 E 轮。具体而言，在 ClientUpdate 函数中，θ 和 Σ 分别是迄今为止观测到的小批量梯度的加权累积一阶矩和二阶矩，其中 $\beta_1, \beta_2 \in \in (0,1)$为动量参数。根据小批量大小$B$，在内部**for**循环的前几次迭代中，$\Sigma$ 的最小特征值可能为零或极小，从而导致w发生显著波动。因此，使用 $\varepsilon > 0$ 来平滑更新。
- 最后，收到本地更新w_{cr+1}后，参数服务器根据各模型对应的本地数据量对其进行聚合，从而得到w_{r+1}。

值得注意的是，该算法可以通过使用全局步长得到改进。我们将这一方向留待未来工作。

3.4 主动学习增强的 FLTP（ALFLTP）

我们使用主动学习，谨慎选择参与每次迭代的客户端。不同于现有文献[36], ，我们不直接将整个损失函数用作指标，而是考虑两种不确定性感知的客户端选择指标：负对数似然（NLL）和偶然不确定性（AU）。

在算法 2 中，每个客户端都有一个值变量 v_i。在每一轮 r（其中 $r > 2$），参数服务器会对客户端进行两次采样。首先，它像算法 1 中那样随机选择 $\lfloor f_2 C \rceil$ 个客户端。被选中的每个客户端 $\mathscr{L}_r$都会根据以下公式更新其本地值v

$$G_c(\mathscr{D}_c, w) = \frac{1}{K_c} \sum_{S \in \mathscr{D}_c} G(S, w)$$

其中

$$G(S,w) = \begin{cases} \frac{1}{T_{pre}} \sum_{t=1}^{T_{pre}} \left[\log(2\hat{b}_{i^*,t,f_{best_{i^*}}}) + \frac{\| y_{i,t} - \hat{\mu}_{i^*,t,f_{best_{i^*}}} \|_1}{\hat{b}_{i^*,t,f_{best_{i^*}}}} \right] & \text{if } NLL \\ \frac{1}{T_{pre}} \sum_{t=1}^{T_{pre}} \hat{b}_{i^*,t,f_{best_{i^*}}} & \text{if } AU \end{cases}$$

其中，$\hat{\mu}_{i^*,t,f_{besti^*}}$ 表示目标智能体在时间步 t 处、来自 F 条轨迹中最佳模式的预测位置，$\hat{b}_{i^*,t,f_{besti^*}}$ 表示相应的偶然不确定性。不属于 $\mathscr{L}_r$ 的所有客户端都会将其值$v_i = 0$重置。如果使用 NLL 作为选择指标，则参数服务器会选择集合 $\mathscr{L}_r$，即具有最高v_i的 $\lfloor f_1 C \rceil$ 个客户端；如果使用 AU 作为指标，则参数服务器会选择 $\mathscr{L}_r$，其中包含值v_i 最接近其中位数的 $\lfloor f_1 C \rceil$ 个客户端。当$r = 1$ 时（即算法 2 中的第 2–6 行），由于没有来自上一轮的全局模型用于计算值，客户端选择方法与 FLTP 中的类似。

将负对数似然（NLL）作为选择指标 我们选择 NLL 作为选择指标之一，原因有以下两点：由于 NLL 是损失函数的一部分，如果全局模型针对某个客户端的本地数据训练得不够充分，该客户端的 NLL 就会更高。本地数据呈非独立同分布（non-IID），涵盖不同的驾驶场景，因此，选择 NLL 较高的客户端，可以促使联邦学习中的全局模型在数据更难处理的客户端上进行更多本地训练。

算法 1FLTP

输入：初始模型 w_0,客户端数量 C，客户端采样率 $f_1 \in (0,1]$,本地数据量 $\{K1,\cdots,K_C\}$，步长η，批量大小 B，训练轮次数 E，权重衰减 λ，动量参数 $\beta_1,\beta_2 \in (0,1)$，平滑参数 ε；输出：w_R；1：初始化：$w \leftarrow w_0$；2：对于每一轮 $r = 1$ 到 R：3：根据概率向量 $[K_1/K \cdots, K_C/K]$，无放回抽样随机选取 $\lfloor f_1 C \rfloor$ 个客户端。令$\mathscr{L}_r$ 表示由此得到的客户端集合；4：向每个选定客户端 $\mathscr{L}_r$发送 w_{r-1}；5：对于每个并行执行的客户端 $c \in \mathscr{L}_r$：6：$w_r^c \leftarrow \text{ClientUpdate}(w_r, \eta, B, E\ \beta_1,\beta_2,\varepsilon)$；7：结束循环；8：$w_r \leftarrow \sum_{c\in\mathscr{L}_r} K_c K_r^{-1} w_r^c$，其中 $K_r \triangleq \sum_{c\in\mathscr{L}_r} K_c$；9：结束循环；10：返回 w_R；11：ClientUpdate($w\ \eta$, B, E $\beta_1,\beta_2,\varepsilon$)；12：初始化：一阶矩 $\theta \leftarrow 0$，二阶矩 $\Sigma \leftarrow 0$，计数器 $t \leftarrow 0$；13：对于每个本地训练轮次 $i = 1,\cdots,E$：14：将$\mathscr{B}_c \leftarrow$ 划分为批次，每批批量大小为 B；15：对于每个批次 $b \in \mathscr{B}_c$：16：$t \leftarrow t + 1$17：$g = \nabla F_c(b, w)$ 18：$w \leftarrow w - \eta\lambda g$ 19：$\theta \leftarrow \beta_1\theta + (1-\beta_1)g$, $\Sigma \leftarrow \beta_2\Sigma + (1-\beta_2)gg^\top$20：$\hat{}\theta \leftarrow \frac{\theta}{1-\beta^t 1}$, $\hat{}\Sigma \leftarrow \frac{\Sigma}{1-\beta^t 2}$ 21：$w \leftarrow w - \eta\left(\hat{} \Sigma^{1/2} + \varepsilon I\right)^{-1}\hat{\theta}$22：结束循环；23：结束循环；24：返回 w

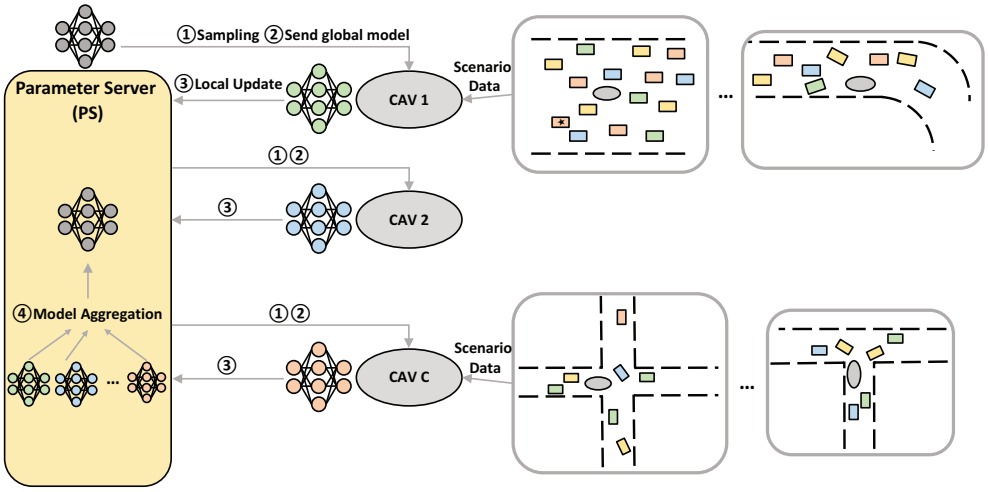
将偶然不确定性用作选择指标该指标的灵感来源于 [38],：在该方法中，采用增量主动学习进行人类轨迹预测，以评估候选数据样本，并选出更有价值的样本。具体而言，该方法会剔除含噪声和冗余的轨迹候选数据样本，而基于筛选后的数据样本训练的模型性能更佳。在我们的 ALFLTP 中，我们利用偶然不确定性来衡量数据噪声的程度。偶然不确定性高，意味着数据噪声很大；偶然不确定性低，则意味着数据较为简单，模型对其预测也更有把握。因此，我们倾向于选择偶然不确定性处于中位数的客户端，因为这些客户端的数据既有代表性，噪声也较少。

放宽更新 v 时所有客户端均参与的要求。为便于说明，在算法 2 的第 10–15 行中，我们让每个客户端都参与更新v_c。在实践中，只需让 $\mathscr{Q}_r$ 中的客户端执行值更新即可。由于值更新不依赖于v_c之前的任何值，更新后的值仅用于 PS 端排序，而且 PS 知道 $\mathscr{Q}_r$，因此 PS 可以将$v_c = 0$ 视为对所有$c \notin \mathscr{Q}_r$均成立。

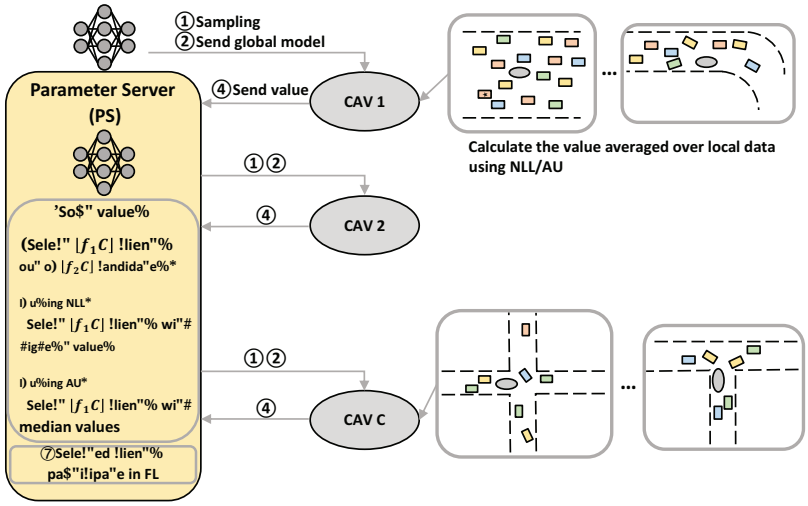
4 实验

4.1 实验设置

数据集： 我们使用 Argoverse Motion Forecasting v1.1 数据集进行训练和评估。为模拟联邦学习中的分布式轨迹数据，我们根据每个数据样本（驾驶场景）的城市标签，将训练集分配给 100 个客户端，其中 95521 个样本来自匹兹堡，110421 个样本来自迈阿密。具体而言，来自这两座城市的样本平均分配给 50 个客户端，每个客户端代表一辆能够在其所在区域收集和处理交通数据的自动驾驶车辆。验证集包含 39472 个样本。所有训练和验证场景均包含以 10 Hz 采样的 5 秒轨迹以及地图信息。Argoverse Motion



(a) FLTP框架



(b) ALFLTP框架

图 1：FLTP和ALFLTP框架。每个灰色椭圆代表联邦学习中的一个客户端（即一辆自车）。每个客户端使用其传感器和摄像头收集本地驾驶场景数据，进行本地更新，并与中央服务器交换模型权重。对于每个场景，自车旨在根据场景中智能体的历史轨迹和地图信息，预测目标车辆（星号标记）的未来轨迹。在ALFLTP中，每轮会根据候选客户端的负对数似然或偶然不确定性指标值，主动选择客户端参与计算。具体而言，若使用负对数似然作为指标，则选择数值最高的m 个客户端；若使用偶然不确定性作为指标，则选择数值处于m中位数的客户端。

Forecasting 预测任务是根据过去 2 秒的轨迹作为输入，预测焦点智能体未来 3 秒的轨迹。

模型与训练参数： 我们使用具有 64 个隐藏维度的 HiVT [6] 作为轨迹预测模型，该模型是一种轻量级的、基于 Transformer 的模型。我们为FLTP和ALFLTP中的本地HiVT模型采用与集中式HiVT相似的参数设置。具体而言，对于FLTP中的每个本地模型，学习率 η、权重衰减、丢弃率、本地批量大小B、本地训练轮数E和本地优化器分别设为 5×10^{-4}、 1×10^{-4}、0.1、32、4和AdamW。我们对FLTP和ALFLTP训练250轮。每轮中被选中参与通信的客户端比例f_1 设为0.1。

评估指标：我们使用负对数似然、最小平均位移误差（minADE）、最小最终位移误差（minFDE）和漏检率（MR）对模型性能进行定量评估。minADE 衡量最佳预测轨迹（即终点误差最小的轨迹）与真实值之间的平均 L2距离。minFDE 衡量最佳预测轨迹与真实值之间的终点 L2距离。MR

算法 2ALFLTP

输入：初始模型 $\mathbf{w}_0$, 步长 η，训练轮数 $\mathbf{E}$，客户端采样率 $\mathbf{f}_1 \in (0,1]$, 候选采样率 $\mathbf{f}_2 \in (0,1]$, 本地数据量 $\{\mathbf{K1}, \cdots, \mathbf{K}_C\}$，权重衰减 λ，动量参数 $\beta_1, \beta_2 \in (0, 1)$，平滑参数 ε；输出：$\mathbf{w}_R$1：初始化 $\mathbf{w} \leftarrow \mathbf{w}_0$2：参数服务器根据概率向量 $[\mathbf{K}_1/\mathbf{K} \cdots, \mathbf{K}_C/\mathbf{K}]$ 无放回抽样，随机抽取 $\lfloor \mathbf{f}_1\mathbf{C} \rfloor$ 个客户端。令 $\mathscr{L}_r$ 表示抽取出的客户端集合；3：对每个客户端 $\mathbf{c} \in \mathscr{L}_1$ 并行执行 4：$\mathbf{w}_1^c \leftarrow$ 客户端更新$(\mathbf{w}\ \eta, \mathbf{B}, \mathbf{E}\ \beta_1, \beta_2, \varepsilon)$ 5：结束循环 6：$\mathbf{w}_1 \leftarrow \sum \mathbf{c}_{\in \mathscr{L}_1} \mathbf{Kc}^{-}\mathbf{K1}\ \mathbf{w}_1^c$，其中 $\tilde{}\mathbf{K}_1 \triangleq \sum \mathbf{c}_{\in \mathscr{L}_1} \mathbf{K}_c$7：对于每一轮 $\mathbf{r} = 2$ 至 $\mathbf{R}$，执行 8：参数服务器按第 2 行的方法随机选择 $\lfloor \mathbf{f}_2\mathbf{C} \rfloor$ 个客户端。令 $\mathscr{Q}_r$ 表示抽取出的客户端集合；9：参数服务器向 $\mathscr{Q}_r$ 中的每个客户端广播 $\mathbf{w}_{r-1}$；10：对每个客户端 $\mathbf{c}$ 并行执行 11：如果 $\mathbf{c} \in \mathscr{Q}_r$，则 12：$\mathbf{v}_c \leftarrow \mathbf{G}_c(\mathscr{D}_c, \mathbf{w}_{r-1})$；13：否则 14：$\mathbf{v}_c \leftarrow 0$15：将 $\mathbf{v}_c$ 报告给参数服务器；16：结束循环 17：参数服务器对 $\{\mathbf{v}_c\}_{c \in \mathscr{C}}$ 排序 18：如果 $\mathbf{u}$ = 负对数似然 19：参数服务器选择数值 $\mathbf{v}_i$ 最高的 $\lfloor \mathbf{f}_1\mathbf{C} \rfloor$ 个客户端，并将所得集合记为 $\mathscr{L}_r$；20：否则 21：参数服务器选择数值最接近中位数值 $\mathbf{v}_i$ 的 $\lfloor \mathbf{f}_1\mathbf{C} \rfloor$ 个客户端，并将所得集合记为 $\mathscr{L}_r$；22：结束判断 23：对每个客户端 $\mathbf{c} \in \mathscr{L}_r$ 并行执行 24：$\mathbf{w}_r^c \leftarrow$ 客户端更新$(\mathbf{w}\ \eta, \mathbf{B}, \mathbf{E}\ \beta_1, \beta_2, \varepsilon)$ 25：结束循环 26：$\mathbf{w}_r \leftarrow \sum \mathbf{c}_{\in \mathscr{L}\mathbf{r}}\ \mathbf{Kc}^{-}\mathbf{Kr}\ \mathbf{w}_{r-1}^c$，其中 $\tilde{}\mathbf{K}_r \triangleq \sum \mathbf{c}_{\in \mathscr{L}\mathbf{r}}\ \mathbf{K}_c$。27：结束循环

用于衡量所有预测轨迹的端点误差均大于 2 米的场景所占比例。

4.2 FLTP 与本地数据训练的比较

我们定量比较 FLTP 的全局模型与任意选取的一个客户端的本地模型；该客户端不参与通信，仅使用其本地数据进行更新。这里我们选择客户端 0；选择其他任何客户端都会得到相同结果。如图 2 和表1所示，FLTP 的表现显著优于未参与联邦学习的客户端，这表明 FLTP 能够利用多源交通数据的有效性，尽管它无法显式访问原始本地数据。具体而言，约经过 50 轮后，随着训练轮次增加，客户端 0 的本地模型开始表现变差，这表明未使用联邦学习的本地模型泛化能力较差。

4.3 FLTP 与 ALFLTP 的比较

本节将采用两种主动客户端选择指标且具有不同偏差程度的 ALFLTP 框架与 FLTP 进行比较。从图3和图4可以看出:

- **训练损失的收敛速度：**ALFLTP-NLL 和 ALFLTP-AU 在 $f_2 = 0.30$ 下的回归损失比在 $f_2 = 0.15$ 和 FLTP 下收敛得更快。
- **逐轮验证性能：**ALFLTP-NLL 和 ALFLTP-AU 在$f_2 = 0.15$ 和$f_2 = 0.30$方面均优于 FLTP，漏检率更低。漏检率衡量的是端点误差

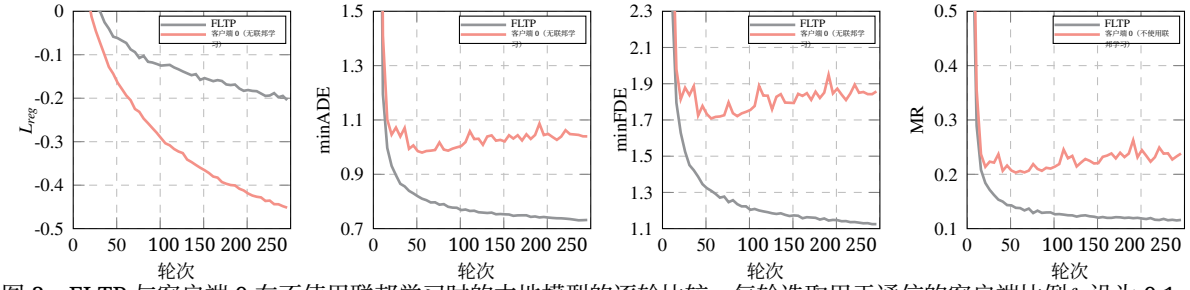


图 2：FLTP 与客户端 0 在不使用联邦学习时的本地模型的逐轮比较。每轮选取用于通信的客户端比例f_1 设为 0.1。

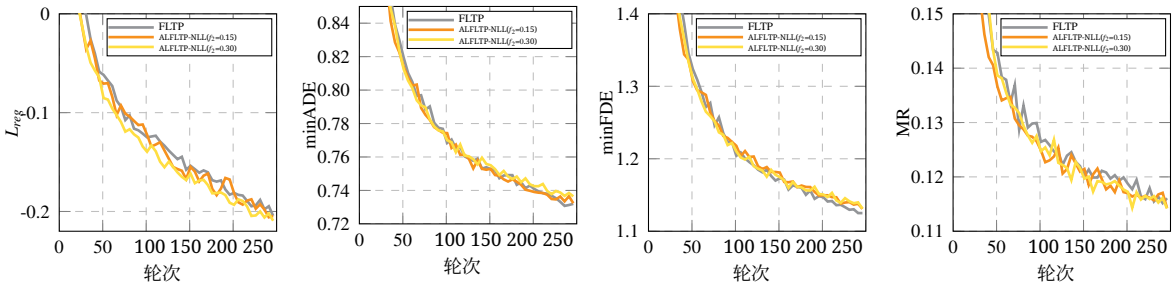


图3：FLTP与ALFLTP-NLL的逐轮比较

大于 2 米的场景所占比例。较低的漏检率表明，在推理阶段，ALFLTP-NLL 和 ALFLTP-AU 对各种交通场景具有更强的鲁棒性。

- **偏差选择对NLL的影响：**约100轮后，由于偏差选择，FLTP在$f_2 = 0.15$ 和$f_2 = 0.30$ 两种设置下的最小平均位移误差和最小最终位移误差均优于ALFLTP-NLL。值得注意的是，采用$f_2 = 0.30$ 的ALFLTP-NLL表现不如采用$f_2 = 0.15$的版本，因为前者引入了更多偏差。
- **偏差选择对偶然不确定性的影响：**与采用$f_2 = 0.15$ 和$f_2 = 0.30$ 的ALFLTP-NLL以及采用$f_2 = 0.15$的ALFLTP-AU相比，采用$f_2 = 0.30$ 的ALFLTP-AU取得了与FLTP相当的最小平均位移误差和最小最终位移误差，并且在多数轮次中的漏检率表现更好，这表明较大的$f2$有助于ALFLTP-AU找到拥有更多代表性数据的客户端。

表1展示了特定轮次中全局模型的详细性能，我们可以看到：

- 在第50轮中，采用$f_2 = 0.30$ 的 ALFLTP-AU 在负对数似然、最小平均位移误差和最小最终位移误差方面优于其他框架。
- 在第150轮中，采用$f_2 = 0.15$ 和$f_2 = 0.30$ 的 ALFLTP-AU 在最小平均位移误差方面优于其他框架。
- 在第250轮，即全局模型完成训练时，采用$f_2 = 0.30$ 的 ALFLTP-AU 在负对数似然、最小平均位移误差和漏检率方面优于其他框架。
- 尽管第250轮中基于联邦学习的 HiVT 模型性能略逊于集中式 HiVT，但其通过避免与服务器进行数据交换，保护了人工驾驶车辆的隐私。

总而言之，在大多数轮次中，ALFLTP-AU 在回归损失方面收敛更快，并且在负对数似然、最小平均位移误差和漏检率方面的性能优于 FLTP。此外，ALFLTP-AU 相比 ALFLTP-NLL 展现出更优且更稳定的逐轮性能。

5 结论

本文提出了一种面向网联自动驾驶汽车的隐私保护与不确定性感知轨迹预测框架，该框架基于联邦学习，并采用不确定性感知全局目标。我们将该框架称为 FLTP：它放宽了必须收集驾驶场景原始数据以构建大型集中式数据集的要求，使网联自动驾驶汽车能够收集本地交通数据，并在不进行显式数据交换的情况下协同训练轨迹预测模型，从而保护交通参与者的隐私。我们进一步提出了主动学习增强型 FLTP（ALFLTP），用于

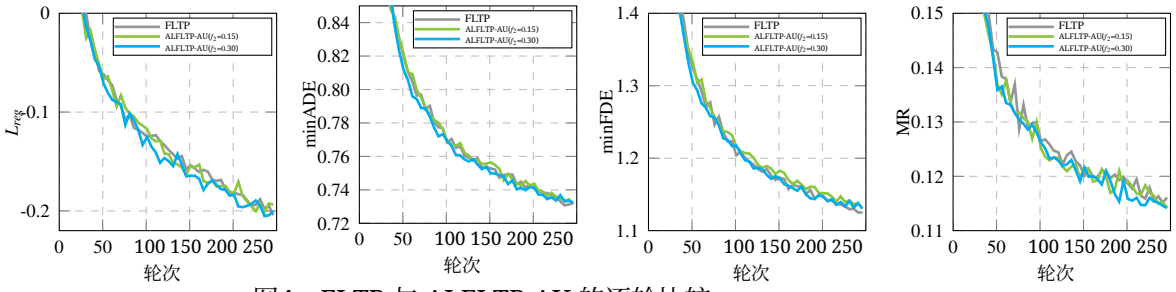


图4：FLTP 与 ALFLTP-AU 的逐轮比较

模型	轮次	负对数似然(↓)	最小平均位移误差(↓)	最小最终位移误差(↓)	漏检率(↓)
集中式 HiVT [6]	-	0.467	0.685	1.028	0.104
客户端 0 不使用联邦学习	50	0.897	0.992	1.732	0.207
FLTP	50	0.629	0.818	**1.318**	0.140
ALFLTP-NLL(联邦学习$_2$=0.15)	50	0.634	0.821	**1.318**	**0.138**
ALFLTP-NLL(f_2=0.30)	50	0.636	0.816	1.319	0.141
ALFLTP-AU(f_2=0.15)	50	0.629	0.819	1.325	0.139
ALFLTP-AU(f_2=0.30)	50	**0.625**	**0.813**	**1.318**	0.141
客户端 0（无联邦学习）	150	1.098	1.026	1.822	0.229
FLTP	150	**0.551**	0.751	**1.170**	0.123
ALFLTP-NLL(联邦学习$_2$=0.15)	150	0.552	**0.750**	1.179	0.122
ALFLTP-NLL(f_2=0.30)	150	0.564	0.753	**1.170**	0.120
ALFLTP-AU(f_2=0.15)	150	0.555	0.753	1.177	**0.119**
ALFLTP-AU(f_2=0.30)	150	0.554	0.752	**1.170**	**0.119**
客户端 0（无联邦学习）	250	1.259	1.059	1.896	0.245
FLTP	250	0.527	0.730	**1.122**	**0.114**
ALFLTP-NLL(f_2=0.15)	250	0.532	0.732	1.139	0.116
ALFLTP-NLL(f_2=0.30)	250	0.536	0.733	1.126	**0.114**
ALFLTP-AU(f_2=0.15)	250	0.531	0.731	1.131	0.115
ALFLTP-AU(f_2=0.30)	250	**0.526**	**0.729**	1.126	**0.114**

表1：Argoverse 验证集上的性能。每轮选择用于通信的客户端比例 f_1 设置为 0.1。

FLTP 中的客户端选择。我们采用两种不确定性感知指标——负对数似然（NLL）和偶然不确定性（AU）——在 FLTP 中主动选择客户端，以实现客户端部分参与。在 Argoverse 数据集上进行的实验表明，FLTP 的表现显著优于使用本地数据训练的模型。此外，在大多数轮次中，ALFLTP-AU 的回归损失收敛速度更快，且在 NLL、最小平均位移误差（minADE）和漏检率（MR）方面的表现优于 FLTP；与 ALFLTP-NLL 相比，其逐轮表现也更加稳定。

参考文献

[1] Mayank Bansal、Alex Krizhevsky 和 Abhijit Ogale。Chauffeurnet：通过模仿最佳驾驶者来学习驾驶并合成最差情况。*arXiv* 预印本 *arXiv:1812.03079*，2018年。

[2] Henggang Cui、Vladan Radosavljevic、Fang-Chieh Chou、Tsung-Han Lin、Thi Nguyen、Tzu-Kuo Huang、Jeff Schneider 和 Nemanja Djuric。利用深度卷积网络进行自动驾驶的多模态轨迹预测。发表于*2019*年国际机器人与自动化会议 *(ICRA)*，第 2090–2096 页。IEEE，2019年。

[3] Nemanja Djuric、Vladan Radosavljevic、Henggang Cui、Thi Nguyen、Fang-Chieh Chou、Tsung-Han Lin、Nitin Singh 和 Jeff Schneider。《面向自动驾驶交通参与者的不确定性感知短期运动预测》。载于*IEEE/CVF*冬季计算机视觉应用会议论文集，第2095–2104页，2020年。

[4] Ming Liang、Bin Yang、Rui Hu、Yun Chen、Renjie Liao、Song Feng 和 Raquel Urtasun。《学习用于运动预测的车道图表示》。载于《计算机视觉—ECCV 2020：第16届欧洲会议》，英国格拉斯哥，2020年8月23日至28日，论文集，第二部分16，第541–556页。施普林格，2020年。[5] Jiyang Gao、Chen Sun、Hang Zhao、Yi Shen、Dragomir Anguelov、Congcong Li 和 Cordelia Schmid。《VectorNet：从向量化表示中编码高精地图和智能体动态》。载于IEEE/CVF计算机视觉与模式识别会议论文集，第11525–11533页，2020年。[6] Zikang Zhou、Luyao Ye、Jianping Wang、Kui Wu 和 Kejie Lu。《HiVT：用于多智能体运动预测的分层向量变换器》。载于IEEE/CVF计算机视觉与模式识别会议论文集，第8823–8833页，2022年。[7] Jiquan Ngiam、Benjamin Caine、Vijay Vasudevan、Zhengdong Zhang、Hao-Tien Lewis Chiang、Jeffrey Ling、Rebecca Roelofs、Alex Bewley、Chenxi Liu、Ashish Venugopal 等。《场景变换器：用于预测多个智能体轨迹的统一架构》。arXiv预印本 arXiv:2106.08417，2021年。[8] Junru Gu、Chen Sun 和 Hang Zhao。《DenseTNT：基于密集目标集的端到端轨迹预测》。载于IEEE/CVF国际计算机视觉会议论文集，第15303–15312页，2021年。[9] Yicheng Liu、Jinghuai Zhang、Liangji Fang、Qinhong Jiang 和 Bolei Zhou。《使用堆叠式变换器进行多模态运动预测》。载于IEEE/CVF计算机视觉与模式识别会议论文集，第7577–7586页，2021年。[10] Xinshuo Weng、Boris Ivanovic、Kris Kitani 和 Marco Pavone。《这究竟是谁的跟踪轨迹？通过基于亲和度的轨迹预测提高对跟踪误差的鲁棒性》。载于IEEE/CVF计算机视觉与模式识别会议论文集，第6573–6582页，2022年。[11] Qingzhao Zhang、Shengtuo Hu、Jiachen Sun、Qi Alfred Chen 和 Z Morley Mao。《自动驾驶车辆轨迹预测的对抗鲁棒性》。载于IEEE/CVF计算机视觉与模式识别会议论文集，第15159–15168页，2022年。[12] Mohammadhossein Bahari、Saeed Saadatnejad、Ahmad Rahimi、Mohammad Shaverdikondori、Amir Hossein Shahidzadeh、Seyed-Mohsen Moosavi-Dezfooli 和 Alexandre Alahi。《车辆轨迹预测虽有效，但并非处处适用》。载于IEEE/CVF计算机视觉与模式识别会议论文集，第17123–17133页，2022年。[13] Dorothy J Glancy。《自动驾驶汽车中的隐私》。《Santa Clara法律评论》，52:1171，2012年。[14] Arendse Huld。《滴滴如何触犯中国网络安全监管机构的规定？了解12亿美元罚款》，2022年。[15] Man Zhang、Bran Selic、Shaukat Ali、Tao Yue、Oscar Okariz 和 Roland Norgren。《理解信息物理系统中的不确定性：概念模型》。载于《建模基础与应用：第12届欧洲会议（ECMFA 2016），作为STAF 2016的一部分举行》，奥地利维也纳，2016年7月6日至7日，论文集12，第247–264页。施普林格，2016年。[16] Hsu-kuang Chiu、Jie Li、Rares Ambrus 和 Jeannette Bohg。《用于自动驾驶的概率三维多模态多目标跟踪》。载于2021年IEEE国际机器人与自动化会议（ICRA），第14227–14233页。IEEE，2021年。[17] Sanbao Su、Yiming Li、Sihong He、Songyang Han、Chen Feng、Caiwen Ding 和 Fei Miao。《自动驾驶协同检测的不确定性量化》。arXiv预印本 arXiv:2209.08162，2022年。[18] Bohan Tang、Yiqi Zhong、Ulrich Neumann、Gang Wang、Siheng Chen 和 Ya Zhang。《多智能体轨迹预测中的协同不确定性》。神经信息处理系统进展，34:6328–6340，2021年。[19] Guopeng Li、Zirui LI、Victor Knoop 和 Hans van Lint。《UQNet：通过非参数化且可泛化的方法量化轨迹预测中的不确定性》。可见于SSRN 4241523，2022年。[20] Xiaolin Tang、Kai Yang、Hong Wang、Jiahang Wu、Yechen Qin、Wenhao Yu 和 Dongpu Cao。《预测不确定性感知的自动驾驶车辆决策》。《IEEE智能车辆汇刊》，7(4):849–862，2022年。[21] Nachiket Deo 和 Mohan M Trivedi。《用于车辆轨迹预测的卷积社交池化》。载于IEEE计算机视觉与模式识别会议研讨会论文集，第1468–1476页，2018年。[22] Yarin Gal 和 Zoubin Ghahramani。《将随机失活作为贝叶斯近似：表示深度学习中的模型不确定性》。载于国际机器学习会议，第1050–1059页。PMLR，2016年。[23] Balaji Lakshminarayanan、Alexander Pritzel 和 Charles Blundell。《使用深度集成进行简单且可扩展的预测不确定性估计》。神经信息处理系统进展，30，2017年。

[24] Brendan McMahan、Eider Moore、Daniel Ramage、Seth Hampson 和 Blaise Aguera y Arcas。基于去中心化数据的深度网络高效通信学习。载于人工智能与统计学，第1273–1282页。PMLR，2017年。[25] Peter Kairouz、H Brendan McMahan、Brendan Avent、Aurélien Bellet、Mehdi Bennis、Arjun Nitin Bhagoji、Kallista Bonawitz、Zachary Charles、Graham Cormode、Rachel Cummings 等。联邦学习的进展与未解难题。《机器学习基础与趋势》，14(1–2):1–210，2021年。[26] Haoran Song、Wenchao Ding、Yuxuan Chen、Shaojie Shen、Michael Yu Wang 和 Qifeng Chen。Pip：面向自动驾驶的规划感知轨迹预测。载于计算机视觉—*ECCV 2020*：第*16*届欧洲会议，英国格拉斯哥，*2020*年*8*月*23*日至*28*日，论文集，第*21*部分*16*，第598–614页。施普林格，2020年。[27] Ye Yuan、Xinshuo Weng、Yanglan Ou 和 Kris M Kitani。Agentformer：用于社会时空多智能体预测的智能体感知变换器。载于*IEEE/CVF*国际计算机视觉会议论文集，第9813–9823页，2021年。[28] Xin Li、Xiaowen Ying 和 Mooi Choo Chuah。Grip：基于图的交互感知轨迹预测。载于*2019*年*IEEE*智能交通系统会议（*ITSC*），第3960–3966页。IEEE，2019年。[29] Hyeongseok Jeon、Junwon Choi 和 Dongsuk Kum。Scale-net：通过边增强图卷积神经网络，在交互车辆数量随机变化的情况下实现可扩展的车辆轨迹预测网络。载于*2020*年*IEEE/RSJ*智能机器人与系统国际会议（*IROS*），第2095–2102页。IEEE，2020年。[30] Yuning Chai、Benjamin Sapp、Mayank Bansal 和 Dragomir Anguelov。Multipath：用于行为预测的多重概率锚点轨迹假设。*arXiv*预印本 *arXiv:1910.05449*，2019年。[31] Tung Phan-Minh、Elena Corina Grigore、Freddy A Boulton、Oscar Beijbom 和 Eric M Wolff。Covernet：使用轨迹集进行多模态行为预测。载于*IEEE/CVF*计算机视觉与模式识别会议论文集，第14074–14083页，2020年。[32] Hang Zhao、Jiyang Gao、Tian Lan、Chen Sun、Ben Sapp、Balakrishnan Varadarajan、Yue Shen、Yi Shen、Yuning Chai、Cordelia Schmid 等。TNT：目标驱动的轨迹预测。载于机器人学习会议，第895–904页。PMLR，2021年。[33] Nathalie Majcherczyk、Nishan Srishankar 和 Carlo Pinciroli。Flow-FL：面向多机器人系统时空预测的数据驱动联邦学习。载于*2021*年*IEEE*国际机器人与自动化会议（*ICRA*），第8836–8842页。IEEE，2021年。[34] Chunnan Wang、Xiang Chen、Junzhe Wang 和 Hongzhi Wang。ATPFL：联邦学习框架下的自动轨迹预测模型设计。载于*IEEE/CVF*计算机视觉与模式识别会议论文集，第6563–6572页，2022年。[35] Jack Goetz、Kshitiz Malik、Duc Bui、Seungwhan Moon、Honglei Liu 和 Anuj Kumar。主动联邦学习。*arXiv*预印本 *arXiv:1909.12641*，2019年。[36] Yae Jee Cho、Jianyu Wang 和 Gauri Joshi。联邦学习中的客户端选择：收敛性分析与选择权策略。*arXiv*预印本 *arXiv:2010.01243*，2020年。[37] Pengfei Li、Yunfeng Zhao、Liandong Chen、Kai Cheng、Chuyue Xie、Xiaofei Wang 和 Qinghua Hu。智能电网联邦学习中基于不确定性度量的主动客户端选择。载于*2022*年*IEEE*智能物联网国际会议（*SmartIoT*），第148–153页。IEEE，2022年。[38] Yi Xi、Dongchun Ren、Mingxia Li、Yuehai Chen、Mingyu Fan 和 Huaxia Xia。通过增量主动学习实现对多个交互行人的鲁棒轨迹预测。载于神经信息处理：第*28*届国际会议，*ICONIP 2021*，印度尼西亚巴厘岛沙努尔，*2021*年*12*月*8*日至*12*日，论文集，第*V*部分*28*，第141–150页。施普林格，2021年。