基于数据特征相关性和自适应差分隐私的深度学习方法研究

康海燕,王骁识
(北京信息科技大学信息安全系,北京 100192)

摘 要: 基于差分隐私的深度学习隐私保护方法中,训练周期的长度以及隐私预算的分配方式直接制约着深度学习模型的效用.针对现有深度学习结合差分隐私的方法中模型训练周期有限、隐私预算分配不合理导致模型安全性与可用性差的问题,提出一种基于数据特征相关性和自适应差分隐私的深度学习方法(deep learning methods based on data feature Relevance and Adaptive Differential Privacy,RADP).首先,该方法利用逐层相关性传播算法在预训练模型上计算出原始数据集上每个特征的平均相关性;然后,使用基于信息熵的方法计算每个特征平均相关性的隐私度量,根据隐私度量对特征平均相关性自适应地添加拉普拉斯噪声;在此基础上,根据加噪保护后的每个特征平均相关性,合理分配隐私预算,自适应地对特征添加拉普拉斯噪声;最后,理论分析该方法(RADP)满足ε-差分隐私,并且兼顾安全性与可用性.同时,在三个真实数据集(MNIST,Fashion-MNIST,CIFAR-10)上的实验结果表明,RADP方法的准确率以及平均损失均优于AdLM(Adaptive Laplace Mechanism)方法、DPSGD(Differential Privacy with Stochastic Gradient Descent)方法和DPDLIGDO(Differentially Private Deep Learning with Iterative Gradient Descent Optimization)方法,并且RADP方法的稳定性仍能保持良好.

关键词: 差分隐私;深度学习;逐层相关性传播;信息熵;隐私度量;隐私预算;拉普拉斯机制
基金项目: 国家社科基金(No.21BTQ079);教育部人文社科项目(No.20YJAZH046);国家自然科学基金(No.61370139)
中图分类号: TP309.2 **文献标识码:** A **文章编号:** 0372-2112(2024)06-1963-14
电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20220892

Research on the Deep Learning Method Based on Data Feature Relevance and Adaptive Differential Privacy

KANG Hai-yan, WANG Xiao-shi
(*Information Security Department*, *Beijing Information Science and Technology University*, *Beijing* 100192, *China*)

Abstract: In the deep learning privacy protection based on differential privacy, the length of the training period and the allocation of the privacy budget directly restrict the utility of the deep learning model. In the existing methods of deep learning combined with differential privacy, the model training cycle is limited and the budget allocation of a large number of feature privacy is unreasonable, which leads to poor security and availability of the model. We propose a method of deep learning methods based on data feature relevance and adaptive differential privacy (RADP). First, the method uses the layer-by-layer correlation propagation algorithm to calculate the average correlation of each feature parameter and the output result on the original data set on the pre-trained model and uses the information entropy-based method to calculate the average correlation of each feature parameter. According to the privacy metric, the Laplace noise is adaptively added to the average correlation; on this basis, according to the average correlation of each feature parameter, the privacy budget is allocated reasonably, Laplace noise is added to the feature parameters; finally, theoretical analysis shows that the method proposed in this paper satisfies ε-differential privacy and take into account security and availability. Based on the experimental results on 3 real datasets MNIST, Fashion-MNIST, and CIFAR-10, the accuracy and average loss of RADP are better than those of the AdLM (Adaptive Laplace Mechanism) method ,the DPSGD (Differential Privacy with Stochastic Gradient Descent) method and the DPDLIGDO (Differentially Private Deep Learning with Iterative Gradient Descent Optimization) method.

收稿日期:2022-07-26;修回日期:2023-04-01;责任编辑:梅志强

Moreover, the stability of RADP method can still be maintained well.

Key words: differential privacy; deep learning; layer-wise relevance propagation; entropy of information; privacy Metrics; privacy budget; laplace mechanism

Foundation Item(s): National Social Science Foundation of China (No. 21BTQ079); Humanities and Social Sciences Research Project of the Ministry of Education (No.20YJAZH046); National Natural Science Foundation of China (No.61370139)

1 引言

在大数据时代,深度学习在许多领域中取得了巨大成功,例如人脸识别、自然语言处理、金融预测和医疗. 然而,在图像识别、自然语言处理和一系列其他任务中使用带有敏感信息的数据进行训练会引发严重的隐私问题[1~5]. 例如,文献[6,7]表明攻击者可以通过成员身份推断攻击,使用位置或购买记录识别出数据集中的成员. 文献[8]表明攻击者可以通过提取攻击从具有十亿参数的GPT-2模型[9]中提取一个人的姓名、电子邮件地址和电话号码. 随着《网络安全法》、《数据安全法》、《个人信息保护法》、《关键信息基础设施安全保护条例》等法律条文的颁布,如何在大数据时代保护个人隐私日益重要. 传统的训练方式会暴露数据中的敏感信息,如何在保证训练模型可用性的同时保护个人隐私是主要的技术挑战.

许多深度学习隐私保护的研究结合了差分隐私及其变体技术[10~15],差分隐私是一种具有严格数学证明的方法,能够防止隐私信息的泄露[16~19]. 自文献[16]以来,基于差分隐私的隐私保护方法中出现了多种基于差分隐私的深度学习保护方法. 现存方法包括DPSGD[16,20],DP-Adam(Differential Privacy-Adaptive Moment Estimation)[21],DP-SGLD(Differential Privacy-Stochastic Gradient Langevin Dynamics)[22,23],DP-FTRL(Differential Privacy-Follow-The-Regularized-Leader)[24],DP-FedAvg(Differential Privacy-Federated Averaging)[18],ILM(Identical Laplace Mechanism)[25],GANobfuscator[26],DPSGD with tempered sigmoid[27]等方法. 其中文献[18,20~25,27,28]均是基于文献[16]展开工作,这些方法向梯度添加噪声,优势在于有严格的梯度裁剪策略,能够一定程度控制由于训练迭代次数造成的隐私预算消耗. 大多分布式深度学习模型是通过参与方传递梯度参数的方式运行,因此这些方法也能够很好的应用于分布式深度学习模型中,防止传递的梯度参数泄露隐私信息. ILM[25]则是一种向特征添加噪声的方法,通过平均分配隐私预算的方式向特征添加噪声,能够彻底避免向梯度添加噪声时隐私预算消耗的问题. AdLM[25]是对ILM的改进,对特征自适应地添加噪声,能够有效减少噪声对原数据精度的影响,但本方法更充分的体现自适应,两次加噪均采用自适应方法,保证更精确的控制噪声对原数据精度的影响.

本文的主要贡献总结如下:(1)提出一种基于数据特征相关性和自适应差分隐私的深度学习方法(deep learning methods based on data feature Relevance and Adaptive Differential Privacy,RADP). (2)设计一种基于差分隐私的数据特征加噪模型,该模型能够在满足ε-差分隐私的同时,保证隐私预算与训练周期完全无关. (3)针对文中两处不同的需加噪数据,提出基于信息熵的隐私度量方法和逐层相关性传播方法的自适应加噪机制.

2 相关工作

基于差分隐私的深度学习隐私保护方案已存在多种方法. 文献[28]提出一种deepee框架,该框架通过神经网络的并行机制,对梯度参数添加噪声,同时保持将神经网络的权重参数放入共享内存,以保持内存效率. 文献[29]提出了一种基于区块链的本地化差分隐私线上分布式学习模型,用于持有移动端设备用户的冠心病诊断. 采用自顶向下的树结构来包含自适应分区的病历. 利用随机响应机制的本地化差分隐私技术保证在没有任何可信第三方的情况下保护患者和医疗交易的隐私,并且确保在分布式学习框架下支持区块链的信息共享认证. 文献[30]提出PrivateDL(Private Deep Learning),该方法在确定隐私预算时考虑局部梯度的异质性和动态变化及其对精度的总体影响,通过基于采样的局部噪声校准灵敏度估计方法和关键数据识别技术,在满足隐私保证的同时提高模型的准确性. 文献[31]提出了一个分布式、安全、公平的联邦深度学习框架DeepChain,结合基于区块链的激励机制的同时,通过对梯度添加噪声为每个参与者保证数据隐私. 文献[32]提出安全隐私梯度计算模型SPGC(Secure and Private Gradient Computation). SPGC模型的主要思想是在安全多方计算中结合差分隐私的加噪机制. 文献[33]利用自编码器和生成对抗网络结合RDP(Rényi Differential Privacy)[13],为合成数据设计出一种差分隐私框架,考虑了数据特征,能够有效节省隐私预算. 文献[34]通过引入RDP,降低隐私预算成本,改进了PATE(Private Aggregation of Teacher Ensembles)模型,并且在实际效用方面优于原PATE[35]. 文献[36]提出了一种基于安全多方计算协议的分布式学习模型,各方协同学习后在全局模型的输出结果中添加拉普拉斯噪声. 文献[37]通过数学分析得出一个更严格的全局敏感度上界,并且设计了一个新的差分隐私框架,将噪声加入到神经网络输出层的单个神经

元中. 该方法能够进一步节约隐私预算. 文献[38]提出了一种DPNAS(Neural Architecture Search for Deep Learning with Differential Privacy)方法,该方法采用神经架构搜索方法,自动设计基于差分隐私的深度学习模型. 文献[39]设计了一种新的扰动迭代梯度下降优化PIGDO(Perturbed Iterative Gradient Descent Optimization)算法和新的MMA(MoMents Accountant)方法,该方法名为DPDLIGDO,获得了更严格的隐私损失界限,能够进一步节省隐私预算,进而更合理的添加噪声.

3　基本概念和相关知识

3.1　差分隐私

定义1　ε-差分隐私:给定数据集D和D',二者相互之间至多相差一条记录,即$|D\Delta D'|\leqslant 1$,则称D和D'为相邻数据集. 给定一个隐私算法Range(M)为M的输出范围,若算法M在数据集D和D'上任意输出结果$O(O\in\text{Range}(M))$满足式(1),则算法M满足ε-差分隐私.

$$\frac{\Pr[M(D)=O]}{\Pr[M(D')=O]}\leqslant e^{\varepsilon} \tag{1}$$

其中,概率$\Pr[\cdot]$由算法M的随机性控制,反映了隐私被披露的风险;隐私预算参数ε表示隐私保护程度,ε越小隐私保护程度越高.

定义2　全局敏感度:设f为查询函数,且$f: D\to R^k$,f的全局敏感度如式(2)所示:

$$\Delta f=\max_{D,D'}\left\|f(D)-f(D')\right\|_1 \tag{2}$$

其中,D和D'为相邻数据集,R代表输出结果,k表示输出维度.

定义3　拉普拉斯机制:对于数据集D上任意查询函数f,经隐私保护算法M使用拉普拉斯分布的概率密度函数laplace生成噪声扰动f,扰动后输出结果满足式(3),称算法M满足拉普拉斯机制的ε-差分隐私,其中Δf为全局敏感度,ε表示隐私预算.

$$M(D)=f(D)+\text{Laplace}\left(\frac{\Delta f}{\varepsilon}\right) \tag{3}$$

性质1　序列组合性[40]:对于数据集D,有一系列算法$M_1,M_2,\cdots,M_n$均满足ε-差分隐私,隐私预算分别为$\varepsilon_1,\varepsilon_2,\cdots,\varepsilon_n$,则由这一系列算法组合成的新算法$M\left(M_1(D),M_2(D),\cdots,M_n(D)\right)$,提供隐私预算为$\sum_{i=1}^{n}\varepsilon_i$的$\varepsilon$-差分隐私保护.

性质2　后处理免疫性[41]:对于数据集D,算法M满足ε-差分隐私,对任意算法A(A不一定满足ε-差分隐私),新算法$M'=A\left(M(D)\right)$仍满足ε-差分隐私.

3.2　逐层相关性传播算法

逐层相关性传播算法[42],从模型输出开始,逐层传播,直到模型输入为止,用于计算深度学习模型每个输入特征与模型输出结果的相关性.

模型输出结果记为$F_{x_i}(\theta)$,给定某一神经元p,p处于第l层,神经元p的相关性系数记为$R_p^{(l)}(x_i)$. 逐层相关性传播算法的目标是根据$l+1$层的神经元m发送给l层的神经元p的信息,对l层的神经元相关性进行分解,信息记为$R_{p\leftarrow m}^{(l,l+1)}(x_i)$.

假设已知$l+1$层神经元m的相关性$R_m^{(l+1)}(x_i)$,可将该相关性$R_m^{(l+1)}(x_i)$分解到第l层的所有神经元上,如式(4)所示:

$$R_m^{(l+1)}(x_i)=\sum_{p\in l}R_{p\leftarrow m}^{(l,l+1)}(x_i) \tag{4}$$

第l层神经元p的相关性$R_p^{(l)}(x_i)$可理解为,第$l+1$层中所有神经元的相关性分解后再进行求和,如式(5)所示:

$$R_p^{(l)}(x_i)=\sum_{m\in(l+1)}R_{p\leftarrow m}^{(l,l+1)}(x_i) \tag{5}$$

逐层相关性传播算法示意图如图1所示.

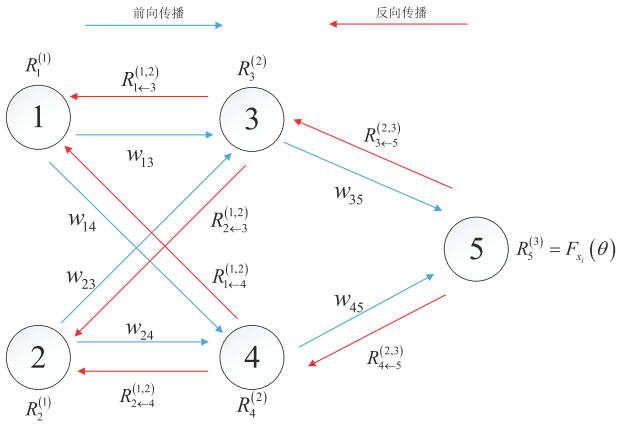


图1　逐层相关性传播算法示意图

式(5)中,信息$R_{p\leftarrow m}^{(l,l+1)}(x_i)$是根据单一仿射变换与全局仿射变换的比值得出的,如式(6)所示:

$$R_{p\leftarrow m}^{(l,l+1)}(x_i)=\begin{cases}\dfrac{z_{pm}(x_i)}{z_m(x_i)+\varepsilon}R_m^{(l)}(x_i), & z_m(x_i)\geqslant 0\\ \dfrac{z_{pm}(x_i)}{z_m(x_i)-\varepsilon}R_m^{(l)}(x_i), & z_m(x_i)\leqslant 0\end{cases} \tag{6}$$

其中,$z_{pm}(x_i)$为第l层神经元p对第$l+1$层神经元m的仿射变换,如式(7)所示. $z_m(x_i)$为第l层所有神经元对第$l+1$层神经元m的仿射变换,如式(8)所示:

$$z_{pm}(x_i)=w_{ij}\times p(x_i) \tag{7}$$

$$z_m(x_i)=\sum_{p\in l}z_{pm}(x_i)+b_m \tag{8}$$

给定输入特征值x_i的情况下,$p(x_i)$表示神经元p

的值，w_{ij}是第l层神经元p连接到第$l+1$层神经元m的连接权重，b_m是偏置项．引入偏置项防止$z_m(x_i)=0$，导致结果不可计算．

假设给定l层隐藏层，每个特征x_{ij}相对于输出结果$F_{x_i}(\theta)$的相关性$R_{x_{ij}}(x_i)$遵循式(9)：

$$F_{x_i}(\theta)=\sum_{m\in l}R_m^{(l)}(x_i)=\cdots=\sum_{x_{ij}\in x_i}R_{x_{ij}}(x_i) \tag{9}$$

最后，对$R_{x_{ij}}(x_i)$做归一化处理，设共有d维特征，归一化处理如式(10)所示：

$$p\left(R_{x_{ij}}(x_i)\right)=\frac{R_{x_{ij}}(x_i)}{\sum_{j=1}^{d}R_{x_{ij}}(x_i)} \tag{10}$$

归一化后保证$p\left(R_{x_{ij}}(x_i)\right)\in[0,1]$.

本文利用逐层相关性传播算法计算出每个输入特征对结果的特征平均相关性后，根据特征平均相关性自适应的对特征添加噪声，相关性越高的特征添加更少的噪声．

3.3 信息熵

定义4 信息熵[43]：假设X是一个离散随机变量，其取值空间为R，满足概率分布$p(x)=P(X=x),x\in R$，则变量X的信息熵$H(X)$定义如式(11)所示：

$$H(x)=-\sum_{x\in R}p(x)\cdot\log_2 p(x) \tag{11}$$

假设一个数据集有n个数据元组，每个元组有k个特征．x_{ij}表示第i个元组的第j个数据项(即第j个特征值)．$p(x_{ij})$表示x_{ij}出现在数据集中的概率．因此，数据集中第i个元组的第j个数据项的概率计算公式如式(12)所示：

$$p(x_{ij})=\frac{x_{ij}\text{出现的次数}}{\text{数据集中的元组个数}} \tag{12}$$

4 基于数据特征相关性和自适应差分隐私的深度学习方法(RADP)

本文提出基于数据特征相关性和自适应差分隐私的深度学习方法，其核心思想：首先，该方法利用逐层相关性传播算法在模型1上计算出原始数据集上特征和输出结果的特征平均相关性；其次，使用基于信息熵的方法计算特征平均相关性的隐私度量，根据隐私度量对特征平均相关性自适应地添加噪声；然后，根据加噪保护后的特征平均相关性，合理分配隐私预算，自适应地对特征添加噪声；最后，使用加噪后的特征训练模型2，将训练后的模型2用于预测任务．该方法对特征添加噪声，能够解决向梯度加噪造成的隐私预算受训练周期限制问题；两次自适应加噪能够解决平均分配隐私预算带来的隐私预算分配不合理问题；在4.3节通过证明说明多余噪声的存在，并在本方法中避免添加，解决添加多余噪声问题．

RADP的整体流程如图2所示，具体过程如算法1所示．主要包含三个阶段，阶段1：逐层相关性传播与特征平均相关性加噪阶段(算法1中的步骤1、2)；阶段2：特征加噪阶段(算法1中的步骤3、4)以及正式训练阶段(算法1中的步骤5~7)．4.2节和4.3节分别对逐层相关性传播与特征平均相关性加噪阶段(阶段1)和特征加噪阶段(阶段2)做了详细描述；阶段3：将加噪保护后的数据特征带入模型中训练，训练后得到可用于预测任务的模型．

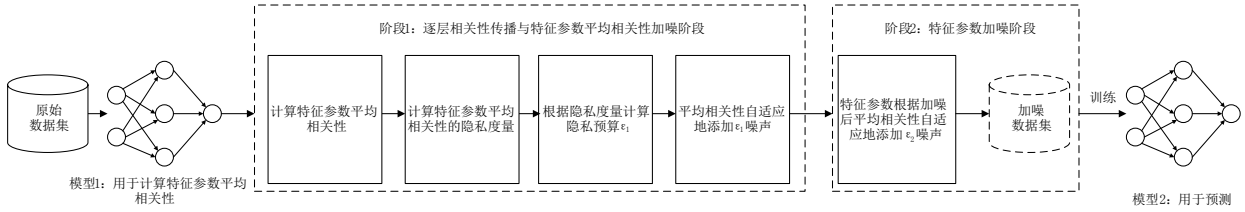


图2 RADP整体流程

4.1 逐层相关性传播与特征平均相关性加噪阶段(阶段1)

4.1.1 逐层相关性传播及特征平均相关性加噪算法

逐层相关性传播及特征平均相关性加噪算法的核心思想：根据模型1上的输出结果，利用逐层相关性传播算法计算出原始数据集上的特征平均相关性．然后，使用基于信息熵的方法计算每个特征平均相关性的隐私度量，根据隐私度量对特征平均相关性自适应地添加噪声．逐层相关性传播及特征平均相关性加噪算法(Rcp_and_Distorted)如算法2所示．

算法2主要分为4个部分，(1)通过逐层相关性传播算法计算数据特征的相关性，并归一化(算法2中的步骤1~9)；(2)根据所有数据的特征相关性计算特征平均相关性(算法2中的步骤10)；(3)计算特征平均相关性的隐私度量，根据隐私度量计算每个特征平均相关性的隐私预算，最终得出特征平均相关性总隐私预算(算法2中的步骤11、12)；(4)对特征平均相关性添加噪声(算法2中的步骤13~15)．下面对算法进行具体描述．

算法1　RADP

输入:原始数据集D,特征x,相关性全局敏感度ΔR,图片尺寸image_size,特征全局敏感度Δx_j,特征隐私预算ε_2.

输出:模型测试指标列表test_list.

1. 使用原始数据集D训练模型1,使用模型1计算特征平均相关性(不同迭代轮数的特征平均相关性不同,因此需要根据后续正式训练所需的迭代轮数计算特征平均相关性),得到训练后模型图sess,预测函数$\overline{y}$
2. $\overline{R_j(D)} \leftarrow$ Rcp_and_ Distorted (sess,$\overline{y}$,ΔR,image_size)　// 对特征平均相关性加噪,详见4.1节
3. $\beta_j \leftarrow$ RedistributeNoise($\overline{R_j(D)}$,image_size)　//导入每个特征加噪后的特征平均相关性,计算出β_j,详见4.2节
4. $\overline{x_j} \leftarrow$ Distorted_Inputs(β,x,image_size,Δx_j,ε_2)　//自适应的分配ε_2,给特征添加噪声得到$\overline{x_j}$,详见4.2节
5. 使用扰动后的特征$\overline{x_j}$训练模型2.
6. 将训练后的模型2用于预测任务.
7. Return test_list

算法2　Rcp_and_ Distorted**算法**

输入:原始数据集D,n项数据,d维特征,相关性全局敏感度ΔR,图片尺寸image_size.

输出:加噪后的特征平均相关性$\overline{R_j(D)}$.

1. 使用原始数据集D训练模型1,使用模型1计算特征平均相关性,得到训练后模型图sess,预测函数$\overline{y}$.
2. FOR n IN sess do//通过逐层相关性传播算法得出特征相关性
3. IF $\overline{y}$
4. THEN $R_{x_{ij}}(x_i) \leftarrow \text{fprop}(R_{x_{ij}}(x_i),n)$
5. IF n.op = "Pool"//n.op代表模型图sess节点的操作名称
6. THEN $R_{x_{ij}}(x_i) \leftarrow \text{fprop\_pool}(R_{x_{ij}}(x_i), n)$
7. IF n.op = "Conv"
8. THEN $R_{x_{ij}}(x_i) \leftarrow \text{fprop\_conv}(R_{x_{ij}}(x_i), n)$
9. $p\left(R_{x_{ij}}(x_i)\right) = \dfrac{R_{x_{ij}}(x_i)}{\sum_{j=1}^{d} R_{x_{ij}}(x_i)}$
10. $R_j(D) = \dfrac{\sum_{i=1}^{n} p\left(R_{x_{ij}}(x_i)\right)}{n}$
11. $\varepsilon_j = H\left(x_j\right) = -R_j(D) \cdot \log_2 R_j(D)$
12. $\varepsilon_1 = \sum_{j=1}^{d} \varepsilon_j = \sum_{j=1}^{d} H\left(x_j\right) = -\sum_{j=1}^{d} R_j(D) \cdot \log_2 R_j(D)$
13. FOR j IN range(0,image_size · image_size)
14. $\overline{R_j(D)} = R_j(D) + \text{laplace}(\dfrac{\Delta R}{\varepsilon_j})$//给特征平均相关性加噪
15. Return $\overline{R_j(D)}$

首先,使用原始数据集训练模型1,目的是得出模型图sess以及预测函数$\overline{y}$,遍历sess得出图中每一个神经元节点n,节点的操作名称记为n.op,根据n.op判断该节点的作用,采取不同的逐层传播计算方式,fprop(),fprop_pool(),fprop_conv()分别代表全连接(Fully connected)层时逐层传播计算方法,池化(Pool)层时逐层传播计算方法以及卷积(Convolution)层时逐层传播计算方法,例如n.op = "Pool"可以判断出该节点是池化层,使用池化层的逐层传播计算方法fprop_pool(),遍历模型图sess后,得出特征的相关性$R_{x_{ij}}(x_i)$,对$R_{x_{ij}}(x_i)$做归一化得到$p\left(R_{x_{ij}}(x_i)\right)$;然后,根据$p\left(R_{x_{ij}}(x_i)\right)$得到特征平均相关性$R_j(D)$,根据信息熵得出$R_j(D)$的隐私度量并计算每个特征平均相关性的隐私预算$\varepsilon_j$,进而得出特征平均相关性的总隐私预算$\varepsilon_1$;最后,通过输入的全局敏感度$\Delta R$和计算得出的隐私预算$\varepsilon_1$,对$R_j(D)$中的每项值添加噪声.

4.1.2　逐层相关性传播的全局敏感度分析及特征平均相关性加噪的隐私性证明

基于3.2节,本小节在此基础上对逐层相关性传播后得出的结果进行敏感度分析.

定理1　设共有d项特征,D个数据,i体现数据,j体现特征,x_i表示第i个数据,x_{ij}代表第i个数据的第j项特征的值,第j项特征的平均相关性为$R_j(D)$,$R_{x_{ij}}(x_i)$表示第i个数据的第j项特征的相关性. 特征平均相关性$R_j(D)$的计算公式如式(13)所示:

$$R_j(D) = \frac{1}{|D|} \sum_{x_i \in D} R_{x_{ij}}(x_i) \tag{13}$$

定理2　设相邻数据集D,D',其中仅第n个数据不同,D和D'两个数据集中的第n个数据分别记为x_n,x'_n,特征平均相关性分别记为$R_j\left(D_{x_n}\right)$,$R_j\left(D_{x'_n}\right)$,特征平均相关性的全局敏感度ΔR如式(14)所示:

$$\Delta R = \frac{1}{|D|} \tag{14}$$

证明

$$\Delta R = \frac{1}{|D|} \left\| \sum_{x_i \in D} R_j\left(D_{x_n}\right) - \sum_{x'_i} R_j\left(D_{x'_n}\right) \right\|_1 \qquad \text{step1}$$

$$= \frac{1}{|D|} \left\| R_j\left(D_{x_{nj}}\right) - R_j\left(D_{x'_{nj}}\right) \right\|_1 \qquad \text{step2}$$

$$\leqslant \frac{1}{|D|} \qquad \text{step3}$$

证毕,可知特征平均相关性的全局敏感度$\Delta R = \dfrac{1}{|D|}$.

因为所有相关性经过归一化后范围均是$R_j\left(D_{x_n}\right) \in [0,1]$,所以可以实现step2到step3的转换.

定理3　Rcp_and_ Distorted算法满足ε_1−差分隐私.

证明　Rcp_and_ Distorted算法(算法2)中步骤14为加噪过程,敏感度如式(14),具体证明过程如下所示.

首先,使用基于信息熵的隐私度量计算方法,计算

每个特征平均相关性的隐私预算.

每个特征平均相关性的信息熵如式(15)所示:

$$\varepsilon_j = H\left(x_{ij}\right) = -R_j(D)\cdot\log_2 R_j(D) \tag{15}$$

其次,根据每一个特征平均相关性的隐私预算ε_j,计算特征平均相关性的总隐私预算ε_1,ε_1的计算如式(16)所示:

$$\varepsilon_1 = \sum_{j=1}^{d}\varepsilon_j = \sum_{j=1}^{d}H\left(x_{ij}\right) = -\sum_{j=1}^{d}R_j(D)\cdot\log_2 R_j(D) \tag{16}$$

最后,证明Rcp_and_ Distorted算法满足ε_1-差分隐私,

$$\begin{aligned}\frac{P_D\left(\overline{R}_j\right)}{P_{D'}\left(\overline{R}_j\right)} &= \prod_{j=1}^{d}\frac{\exp\left(\varepsilon_j\left\|\frac{1}{|D|}\sum_{x_i\in D}R_j\left(x_i\right)-\bar{R}_j\right\|_1\cdot\Delta R^{-1}\right)}{\exp\left(\varepsilon_j\left\|\frac{1}{|D|}\sum_{x_i'\in D}R_j\left(x_i'\right)-\bar{R}_j\right\|_1\cdot\Delta R^{-1}\right)}\\ &\leqslant \prod_{j=1}^{d}\exp\left(\frac{\varepsilon_j}{|D|\Delta R}\left\|\sum_{x_i\in D}R_j\left(x_i\right)-\sum_{x_i'\in D'}R_j\left(x_i'\right)\right\|_1\right)\\ &\leqslant \prod_{j=1}^{d}\exp\left(\frac{\varepsilon_j}{|D|\Delta R}\left\|R_j\left(x_n\right)-R_j\left(x_n'\right)\right\|_1\right)\\ &\leqslant \prod_{j=1}^{d}\exp\left(\frac{\varepsilon_j}{|D|\Delta R}\right)=\exp\left(\frac{\sum_{j=1}^{d}\varepsilon_j}{|D|\Delta R}\right)=\exp\left(\varepsilon_1\right)\end{aligned}$$

证毕,可知Rcp_and_Distorted算法满足ε_1-差分隐私.

4.2 特征加噪阶段(阶段2)

4.2.1 特征加噪算法

特征加噪算法的核心思想:根据加噪保护后的每个特征平均相关性,合理分配隐私预算,自适应地对特征添加噪声. 特征加噪算法(RedistributeNoise and Distorted_Inputs)如算法3所示.

该算法包含两个过程:(1)计算每一个特征分配到的隐私预算百分比β_j(算法3中的步骤1~5);(2)对特征加噪(算法3中的步骤6~9). 下面对算法3进行具体描述.

首先,通过循环计算出所有加噪特征平均相关性的总和sum_$\overline{R}_j(D)$;然后根据sum_$\overline{R}_j(D)$和$\overline{R}_j(D)$求出每个加噪特征平均相关性的百分比β_j;接下来,根据百分比β_j和特征的总隐私预算ε_2,计算每一个特征的隐私预算$\beta_j\cdot\varepsilon_2$;最后,对每一个特征进行扰动,得出加噪后特征$\overline{x_j}$.

算法3 RedistributeNoise and Distorted_Inputs算法

输入:原始数据集D,特征x,图片尺寸image_size,特征全局敏感度Δx_j,特征隐私预算ε_2.

输出:扰动后的特征$\overline{x_j}$.

1. sum_$\overline{R}_j(D)=0$
2. FOR j IN range(0, image_size·image_size):
3. sum_$\overline{R}_j(D)$+= $\overline{R}_j(D)$
4. FOR j IN range(0, image_size·image_size):
5. $\beta_j=\dfrac{\overline{R}_j(D)}{\text{sum\_}\overline{R}_j(D)}$
6. FOR x IN D:
7. FOR j IN range(0, image_size·image_size):
8. $\overline{x_j}=x_j+\text{Laplace}(\dfrac{\Delta x_j}{\beta_j\cdot\varepsilon_2})$
9. Return $\overline{x_j}$

4.2.2 特征的全局敏感度分析及隐私性证明

定理4 设数据批次大小为L,在相邻批次中,每项数据仅有特征x_{ij},x_{ij}'不同. 数据预处理时对特征做归一化处理,因此特征$x_{ij}\in[0,1]$. 特征的全局敏感度如式(17)所示:

$$\Delta x_j = L \tag{17}$$

证明 $$\Delta x_j = \sum_{i=1}^{L}\left\|x_{ij}-x_{ij}'\right\|_1 \leqslant L$$

证毕,可得结论特征的全局敏感度$\Delta x_j=L$.

定理5 RedistributeNoise and Distorted_Inputs算法满足ε_2-差分隐私.

证明 RedistributeNoise and Distorted_Inputs算法(算法3)中步骤8为加噪过程,敏感度如式(17)所示,具体证明如下:

$$\begin{aligned}\frac{p_L(z)}{p_{L'}(z)} &= \prod_{j=1}^{d}\frac{\exp\left(-\varepsilon_j\left|\sum_{x_i\in L}x_{ij}-\sum_{i=1}^{L}z_{ij}\right|\cdot\Delta x_j^{-1}\right)}{\exp\left(-\varepsilon_j\left|\sum_{x_i\in L'}x_{ij}'-\sum_{i=1}^{L}z_{ij}\right|\cdot\Delta x_j^{-1}\right)}\\ &\leqslant \prod_{j=1}^{d}\exp\left(\varepsilon_j\left|\sum_{x_i\in L'}x_{ij}'-\sum_{x_i\in L}x_{ij}\right|\cdot\Delta x_j^{-1}\right)\\ &= \prod_{j=1}^{d}\exp\left(\varepsilon_j\sum_{i=1}^{L}\left\|x_{ij}-x_{ij}'\right\|_1\cdot\Delta x_j^{-1}\right)=\exp\left(\sum_{j=1}^{d}\varepsilon_j\right)=\exp\left(\varepsilon_2\right)\end{aligned}$$

证毕,可知RedistributeNoise and Distorted_Inputs算法满足ε_2-差分隐私.

4.3 RADP方法隐私性分析

本小节主要从差分隐私定义角度,证明RADP方法如何满足ε-差分隐私,并且证明损失函数不需要添加噪声.

定理6 损失函数不需要添加噪声.

证明 由文献[26]可知,一般的想法是,还需要保护损失函数,使其不泄露标签y的隐私信息. 实现这一目的的方法是使用Maclaurin(麦克劳林)公式,将损失函数展开,给展开后公式中y_i的系数添加噪声,具体过程如下:

$$\begin{aligned}\mathrm{Loss}(\hat{y},y)&=-\sum_{i\in L}[y_i\ln\hat{y}_i+(1-y_i)\ln(1-\hat{y}_i)]\\&=-\sum_{i\in L}[y_i\ln\left(\frac{1}{1+\mathrm{e}^{-z}}\right)+(1-y_i)\ln\left(1-\frac{1}{1+\mathrm{e}^{-z}}\right)]\\&=-\sum_{i\in L}[y_i\ln(1+\mathrm{e}^{-z})+(1-y_i)\ln\left(1+\frac{1}{\mathrm{e}^{-z}}\right)]\end{aligned}$$

由麦克劳林公式得到

$$\ln(1+x)=\sum_{n=1}^{\infty}\frac{(-1)^{n-1}}{n}x^n$$

$$\therefore\ln(1+\mathrm{e}^{-z})=\sum_{n=1}^{\infty}\frac{(-1)^{n-1}}{n}\mathrm{e}^{-nz}$$

$$\ln\left(1+\frac{1}{\mathrm{e}^{-z}}\right)=\sum_{n=1}^{\infty}\frac{(-1)^{n-1}}{n}\cdot\frac{1}{\mathrm{e}^{-nz}}$$

$$\begin{aligned}\therefore\mathrm{Loss}(\hat{y},y)=\sum_{i\in L}\Bigg[&y_i\cdot\sum_{n=1}^{\infty}\frac{(-1)^{n-1}}{n}\mathrm{e}^{-nz}\\&+(1-y_i)\cdot\sum_{n=1}^{\infty}\frac{(-1)^{n-1}}{n}\cdot\frac{1}{\mathrm{e}^{-nz}}\Bigg]\end{aligned}$$

给y_i的每一项系数加噪声,相当于给每一项求和项加噪声. 通过上方一系列公式推导后不难看出,损失函数的麦克劳林展开项其实是$\hat{y}$,而$\hat{y}$是由特征x_i经过一层层神经网络后得出的,x_i已经经过ε_2加噪处理,由性质2可知,$\hat{y}$是经过差分隐私保护的. 因此,给$\hat{y}$经过麦克劳林展开后式子的系数加噪不会起到更好的保护效果,还可能降低模型效用.

证毕,可得结论,损失函数不需要添加噪声.

定理7　RADP方法满足ε-差分隐私.

证明　根据算法1描述,主要包含三个阶段:逐层相关性传播阶段,特征加噪阶段以及正式训练阶段. 其中逐层相关性传播阶段和特征加噪阶段设计添加噪声,根据性质1、定理3与定理5,可知RADP满足$(\varepsilon_1+\varepsilon_2)$-差分隐私,即满足$\varepsilon$-差分隐私.

证毕. 可得结论,RADP方法满足$(\varepsilon_1+\varepsilon_2)$-差分隐私.

4.4　时间复杂度分析

RADP方法中两次加噪的计算时间比起深度学习训练时间和LRP(Layer-wise Relevance Propagation)阶段花费的时间可以忽略不计,因此RADP方法主要花费的时间在模型1,模型2的训练和LRP阶段上,由前文可知模型1和模型2的训练轮数应一致,LRP阶段需要用到数据标签值和预测值,标签值和预测值的数量与数据量一致. 设数据量为D,训练轮数为e,深度学习网络神经元节点数为n,RADP方法的时间复杂度为$O(D(2e+n))$;AdLM方法和RADP方法仅在常数级别的计算上有区别,因此AdLM方法和RADP方法时间复杂度一致;DPSGD除了训练模型花费的时间外,每一次梯度下降都需要加噪. 设数据量为D,训练轮数为e,数据批次大小为L,DPSGD方法的时间复杂度为$O(De+(D/L)e)$;DPDLIGDO方法是在常数级别上对DPSGD方法进行改进,因此DPDLIGDO方法的时间复杂度与DPSGD方法一致.

5　实验与分析

5.1　实验环境与数据集

实验环境:实验平台是CPU:AMD R7-5800H,8核16线程,16 GB内存;GPU:NVIDIA RTX3060,6 GB显存;Win10系统. 所设计代码基于Tensorflow框架,采用Python实现.

实验数据集:采用三个真实数据集MNIST,Fashion-MNIST,CIFAR-10.

5.2　评价指标

本文采用通用的评价指标:准确率Accuracy和损失值Loss来作为评价标准. 设TP(True Positive)是实际为真,预测为真的预测样本数量,TN(True Negative)是实际为假,预测为假的预测样本数量,FP(False Positive)是实际为假,预测为真的预测样本数量,FN(False Negative)是实际为真,预测为假的预测样本数量. TP,TN,FP,FN的关系如表1所示.

表1　TP,TN,FP,FN的关系

实际类别	预测类别	
	Positive	Negative
Positive	TP	FN
Negative	FP	TN

定义5　准确率:该指标用于衡量分类器的分类效果与可用性,如式(18)所示:

$$\mathrm{Accuracy}=\frac{\mathrm{TP}+\mathrm{TN}}{\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN}}\tag{18}$$

定义6　预测函数:设θ为预测函数的参数,x为特征,共有n个特征,预测函数如式(19)所示:

$$h_\theta(x)=\theta_1x_1+\cdots+\theta_ix_i+\cdots+\theta_mx_m\tag{19}$$

定义7　损失值:该指标用于衡量分类器预测函数的可用性,设共有m个样本,每个样本的特征为$x^{(i)}$,标签为$y^{(i)}$,根据定义6,损失值如式(20)所示:

$$\mathrm{Loss}(\theta)=\frac{1}{m}\sum_{i=1}^{m}\left(h_\theta(x^{(i)})-y^{(i)}\right)^2\tag{20}$$

5.3　实验过程与结果分析

在三个数据集上通过实验衡量RADP、AdLM、DPSGD和DPDLIGDO四种方法在准确率Accuracy和损失值Loss两项指标上的可用性,并与不加噪声的实验结果(No_DP)做对比. 本文提出RADP方法中的特征平均相关性隐私预算ε_1为计算值,应根据隐私度量的

计算结果决定. 因此,将作为特征隐私预算的ε_2作为变量ε. 首先,根据数据集设置参数ε进行实验测试;其次,在很多深度学习应用差分隐私的方法中,使用了高斯机制. 一般认为高斯机制相比拉普拉斯机制更有助于提升模型性能,但本文所提方法不一定完全适用高斯机制,因此将测试在RADP方法的思想下应用高斯机制和拉普拉斯机制的效果对比;然后,测试RADP方法在不同激活函数中的效果;最后,根据不同的数据集、实验设备性能,设计不同的神经网络,共分为5个测试:

测试1 在MNIST数据集上,分别设$\varepsilon=0.5$,$\varepsilon=1.5$,$\varepsilon=2$,通过不同ε值下四种方法在准确率Accuracy和损失值Loss两项指标上的实验结果,对比四种方法的可用性;所用神经网络结构,采用三层卷积,一层全连接层的结构,每层卷积核分别为4×4,8×8,16×16,全连接层设置32个神经元.

测试2 在Fashion-MNIST数据集上,分别设$\varepsilon=1.5$,$\varepsilon=3$,$\varepsilon=5$,通过不同ε值下4种方法在准确率Accuracy和损失值Loss两项指标上的实验结果,对比4种方法的可用性;所用神经网络结构,采用三层卷积,一层全连接层的结构,每层卷积核分别为16×16,32×32,64×64,全连接层设置128个神经元.

测试3 在CIFAR-10数据集上,分别设$\varepsilon=2$,$\varepsilon=8$,$\varepsilon=12$,通过不同ε值下四种方法在准确率Accuracy和损失值Loss两项指标上的实验结果,对比4种方法的可用性;所用神经网络结构:采用五层卷积,两层全连接层的结构,每层卷积核分别为8×8,16×16,32×32,64×64,128×128,全连接层第一层设置256个神经元,第二层设置128个神经元.

测试4 在MNIST数据集上设置$\varepsilon=1.5$,在Fashion-MNIST数据集上,设置$\varepsilon=3$,在CIFAR-10数据集上,设置$\varepsilon=8$. 分别测试RADP方法应用拉普拉斯机制和高斯机制的实验结果,其中高斯机制设置$\delta=10^{-3}$(按照高斯机制的定义,δ表示高斯机制不满足差分隐私的概率,因此δ在实际应用中应设计的很小). 在每个数据集中,神经网络结构与测试1~3一致.

测试5 在MNIST数据集上设置$\varepsilon=1.5$,在Fashion-MNIST数据集上,设置$\varepsilon=3$,在CIFAR-10数据集上,设置$\varepsilon=8$. 分别测试RADP方法在4种常用激活函数relu,sigmoid,leaky_relu,tanh中的效果. 在每个数据集中,神经网络结构与测试1~3一致.

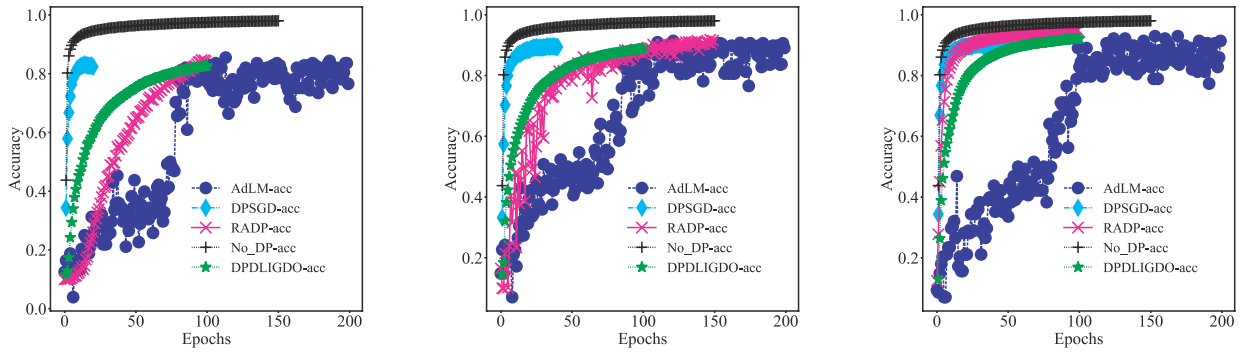
三种数据集所用神经网络结构如表2所示.

表2 实验数据集所用神经网络结构

数据集名称	神经网络结构	每层卷积核	全连接层神经元
MNIST	三层卷积,一层全连接层	第一层4×4 第二层8×8 第三层16×16	32
Fashion-MNIST	三层卷积,一层全连接层	第一层16×16 第二层32×32 第三层64×64	128
CIFAR-10	五层卷积,两层全连接层	第一层8×8 第二层16×16 第三层32×32 第四层64×64 第五层128×128	第一层256 第二层128

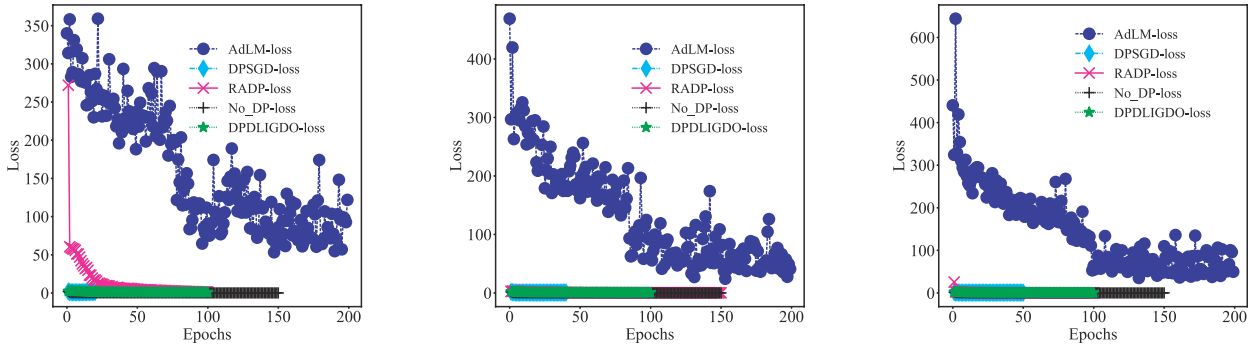
测试1分析 由图3、图4可得如下结论:在较为简单的数据集中训练模型时,RADP方法在准确率Accuracy和损失值Loss两项指标上的结果优于DPSGD方法、AdLM方法和DPDLIGDO方法,并且RADP方法的稳定性良好. 具体原因如下:在MNIST数据集上,根据图3(a)~图3(c)显示,当ε从0.5变化到2,RADP准确率稍高于DPSGD、AdLM和DPDLIGDO,以图3(c)为例,RADP方法准确率最高在0.949左右,DPSGD方法准确率最高在0.903左右,AdLM方法准确率最高在0.914左右,DPDLIGDO方法准确率最高在0.921左右. 但是AdLM方法的准确率有明显的震荡,其原因是AdLM方法给损失函数的参数添加了额外的噪声,对梯度下降过程造成了较大的干扰,进而影响了模型学习过程的稳定性. 根据图4(a)~图4(c),RADP的损失值明显优于AdLM,稍好于DPSGD和DPDLIGDO. 在损失值中同样显示出AdLM方法的不稳定性. 以图4(c)为例,RADP、DPSGD和DPDLIGDO方法的损失值都分别收敛到了0.367,0.736和0.392左右,而AdLM则在50左右,并且AdLM方法的损失值有明显的震荡,原因是加噪过程造成的梯度下降不稳定. 由于MNIST数据集特征简单,所以四种方法的准确率均能达到80%以上.

测试2和测试3分析 由图5~图8可得到下面结论,在较为复杂的数据集中训练模型时,RADP方法在准确率Accuracy和损失值Loss两项指标上的结果优于DPSGD方法、AdLM方法和DPDLIGDO方法,并且RADP方法的稳定性仍能保持良好. 具体原因为:在Fashion-MNIST数据集和CIFAR-10数据集上,根据图5、图7显示,RADP准确率优于DPSGD、AdLM和DPDLIGDO,以图5(c)为例,RADP方法准确率最高在0.858左右,DPSGD方法准确率最高在0.823左右,AdLM方法准确率最高在0.84左右,DPDLIGDO方法准确率最高在0.818左右. 根据图6、图8显示,在损失值上RADP同样表现稳定,稍好于DPSGD和DPDLIGDO,明显优于AdLM,并且AdLM方法损失值难以收敛的特点随着数据集训练难度的增加而愈发明显,以图8(c)为例,RADP,DPSGD和DPDLIGDO方法损失值收敛在



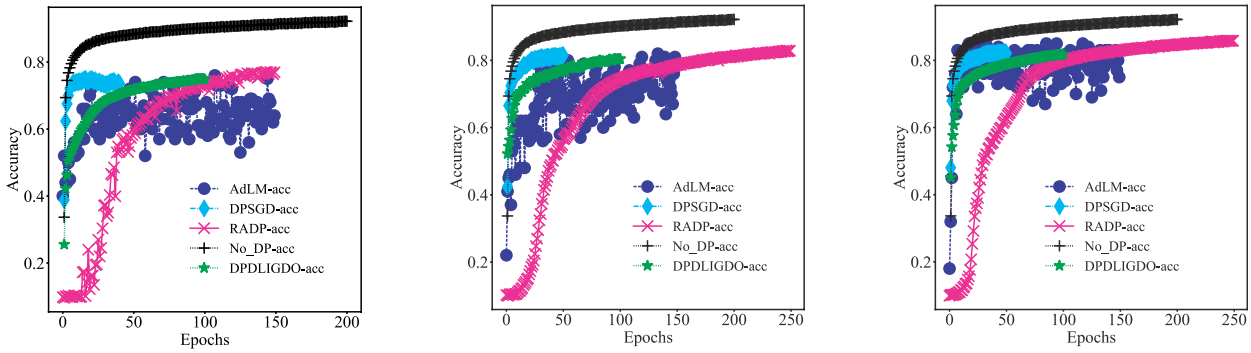
(a) $\varepsilon=0.5$　(b) $\varepsilon=1.5$　(c) $\varepsilon=2$

图3　在MNIST数据集上的准确率(Accuracy)



(a) $\varepsilon=0.5$　(b) $\varepsilon=1.5$　(c) $\varepsilon=2$

图4　在MNIST数据集上的损失值(Loss)



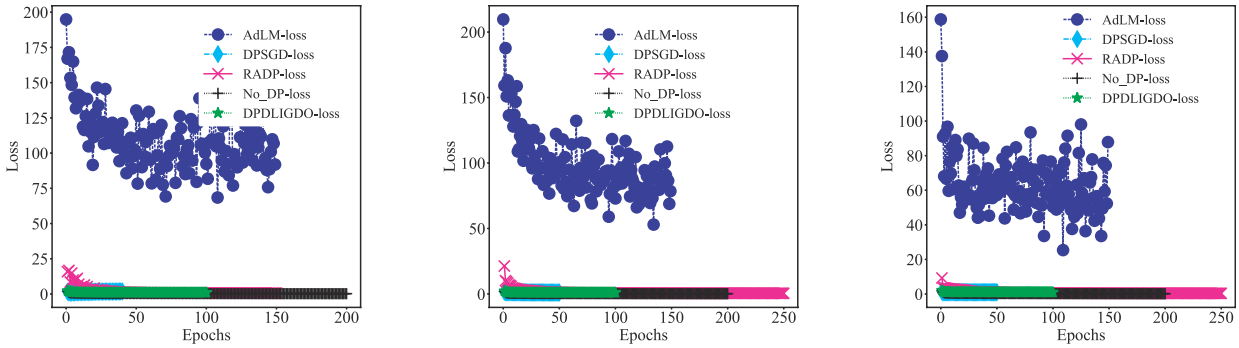
(a) $\varepsilon=1.5$　(b) $\varepsilon=3$　(c) $\varepsilon=5$

图5　在Fashion-MNIST数据集上的准确率(Accuracy)

1.124, 1.875 和 2.01 左右,而 AdLM 方法的损失值则在 120 上下震荡. 四种方法在准确率和损失值两项指标上显示出的特征与在 MNIST 数据集显示的特征一致. 在训练难度最大的 CIFAR-10 数据集上,RADP 方法相比其他三种方法在准确率指标上明显表现更好,RADP 方法的损失值也没有呈现如 AdLM 的不稳定,说明 RADP 方法既可以应用在较为简单的数据集上,也可以应用于较复杂的数据集上,并且方法稳定性较好.

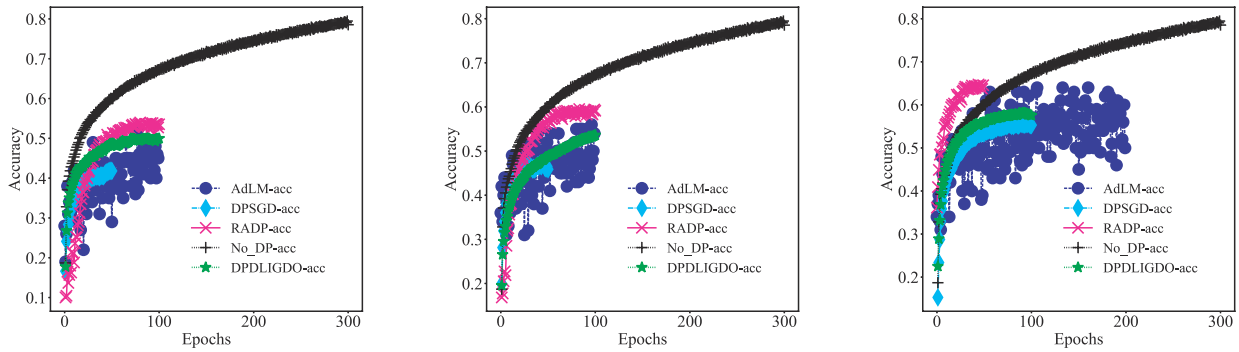
RADP 方法区别于 AdLM 方法的优势在于:(1) AdLM 方法共添加 3 次噪声,第一次采用平均加噪的方式添加在特征平均相关性上,第二次根据加噪保护后的特征平均相关性自适应地对特征加噪,第三次添加在损失函数中;而 RADP 方法仅添加两次噪声,第一次采用信息熵的隐私度量方式计算特征平均相关性的隐私度量,根据隐私度量自适应地对特征平均相关性添加噪声,第二次根据加噪保护后的特征平均相关性可以自适应地对特征加噪;(2) RADP 方法不对损失函数添加噪声,这在

4.3 节进行了数学分析与证明. RADP 方法对 AdLM 方法的优势在于:(1)对特征平均相关性同样采取自适应加噪的方式;(2)通过数学分析,证明不需要对损失函数加噪. 这两个优势使得本方法相比于 AdLM 方法能够进一步精细化加噪过程,提升模型效用.



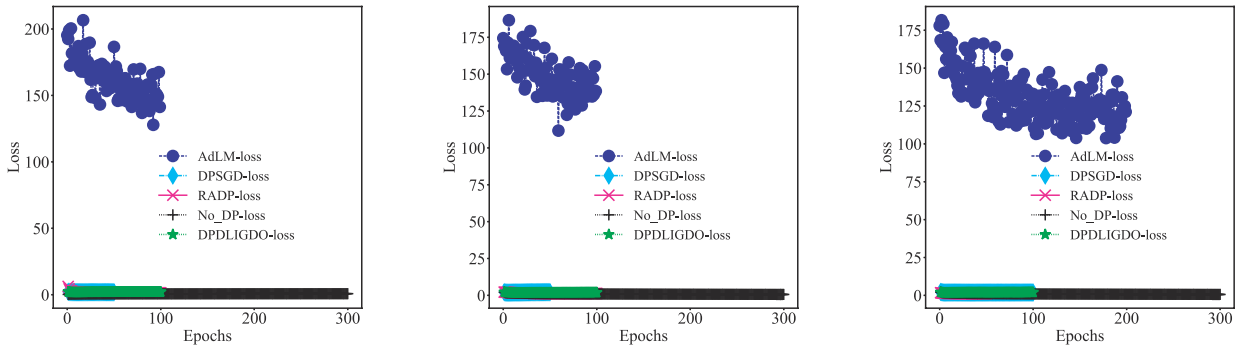
(a) $\varepsilon=1.5$ (b) $\varepsilon=3$ (c) $\varepsilon=5$

图 6 在 Fashion-MNIST 数据集上的损失值(Loss)



(a) $\varepsilon=1.5$ (b) $\varepsilon=3$ (c) $\varepsilon=5$

图 7 在 CIFAR-10 数据集上的准确率(Accuracy)



(a) $\varepsilon=2$ (b) $\varepsilon=8$ (c) $\varepsilon=12$

图 8 在 CIFAR-10 数据集上的损失值(Loss)

测试 4 分析 由图 9~图 11 可得如下结论:在 RADP 方法中,拉普拉斯机制比高斯机制更适用. 具体原因如为,如图 9 所示,设置高斯机制和拉普拉斯机制的隐私预算和敏感度均为 1,其中高斯机制的 $\delta=10^{-3}$,输出 100 000 次噪声结果,从图中的结果可以看出拉普拉斯机制的输出结果更多分布在 0 周围,即拉普拉斯机制输出了更少的噪声,而高斯机制则输出了更多噪声,因此在相同隐私预算和敏感度的情况下,高斯机制添加的噪声比拉普拉斯机制更大.

但是高斯机制优于拉普拉斯机制的方面在于针对

向量值函数,高斯机制应用L_2范数计算敏感度,而拉普拉斯机制应用L_1范数计算敏感度,根据欧氏距离,L_2范数小于等于L_1范数. 因此,高斯机制天然在敏感度上的优势,使得它在很多情况下相比拉普拉斯机制允许更多的隐私预算. 而在本方法中,对特征的敏感度计算方式如定理4所示,换成高斯机制的L_2范数敏感度计算方式则如式(21)所示.

$$\Delta x_j=\sum_{i=1}^{L}\left\| x_{ij}-x'_{ij}\right\|_2=\sqrt{\sum_{i=1}^{L}\left(x_{ij}-x'_{ij}\right)^2}\leqslant\sqrt{L^2}=L \quad (21)$$

在本方法的思想中,高斯机制的敏感度正好等于拉普拉斯机制的敏感度. 因此高斯机制敏感度的优势在本方法的思想中不能体现. 所以在测试4中,会出现高斯机制的应用效果不如拉普拉斯机制的情况.

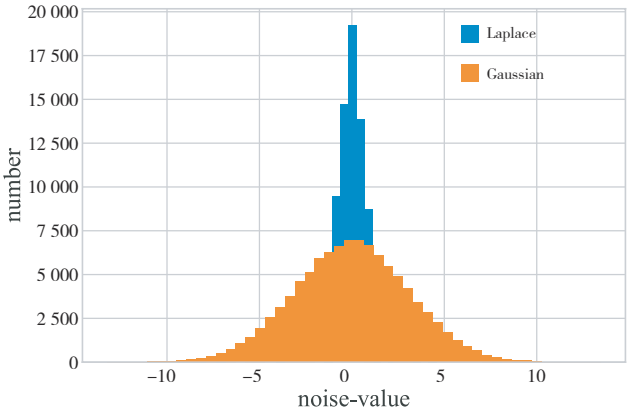
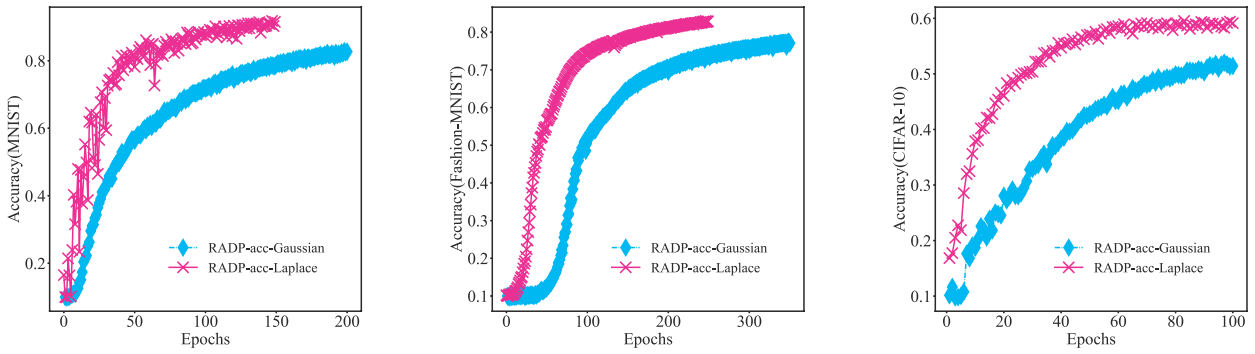
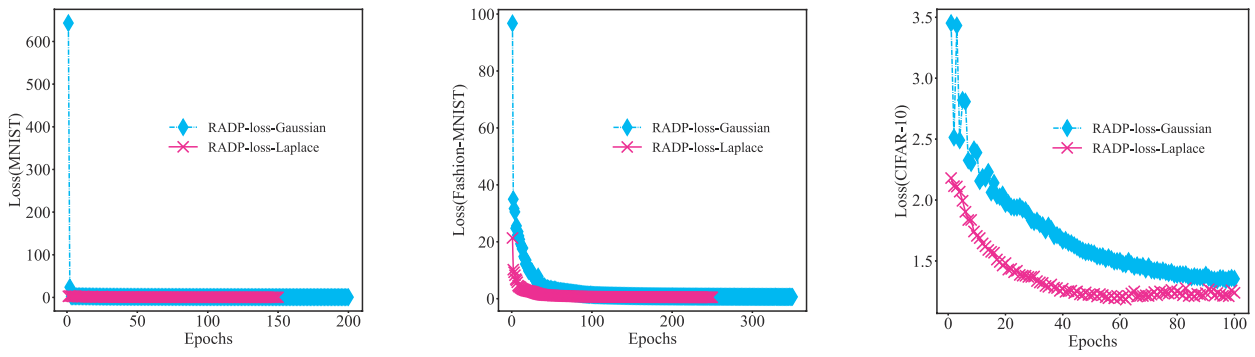


图9　高斯机制和拉普拉斯机制输出100 000次噪声的结果



(*a*) MNIST数据集　(*b*) Fashion-MNIST数据集　(*c*) CIFAR-10数据集

图10　拉普拉斯机制和高斯机制在三个数据集上的准确率(Accuracy)

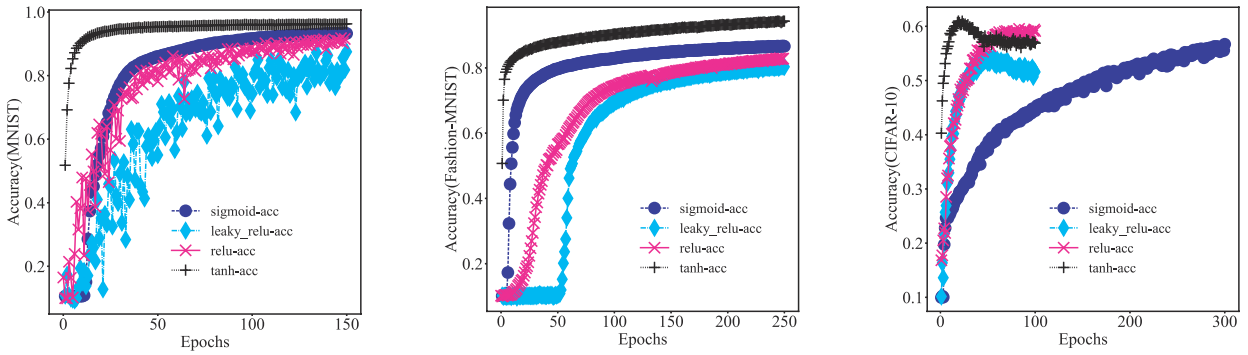


(*a*) MNIST数据集　(*b*) Fashion-MNIST数据集　(*c*) CIFAR-10数据集

图11　拉普拉斯机制和高斯机制在三个数据集上的损失值(Loss)

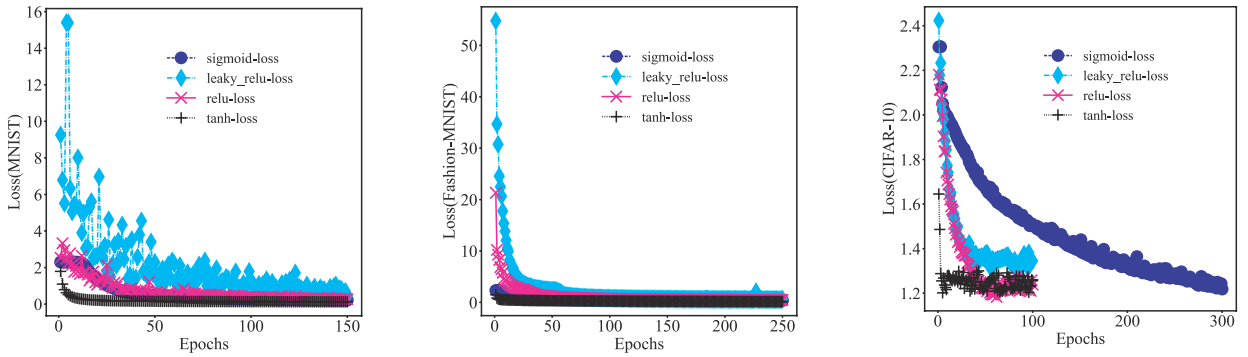
测试5分析　由图12、图13可得如下结论:本实验所采用的神经网络结构中,饱和激活函数(sigmoid,tanh)比非饱和激活函数(relu,leaky_relu)训练效果更好. 具体原因为:当函数自变量趋于正无穷和负无穷时,函数的导数皆趋近于0,称该函数为为饱和函数. 测试5选用了两个饱和激活函数——sigmoid,tanh,两个非饱和激活函数——relu,leaky_relu. 如图12、图13所示,在三个数据集中sigmoid和tanh整体表现优于relu和leaky_relu. 早期神经网络结构比较简单,饱和激活函数应用非常广泛,随着神经网络的层数不断加深,结构不断复杂,为了

解决梯度消失问题并加快模型收敛速度,非饱和激活函数逐渐被越来越广泛的使用. 本测试出现饱和激活函数训练效果强于非饱和激活函数的现象可能是因为所使用的神经网络层数较浅,结构较简单.



(*a*) MNIST数据集 (*b*) Fashion-MNIST数据集 (*c*) CIFAR-10数据集

图12 四种激活函数在三个数据集上的准确率(Accuracy)



(*a*) MNIST数据集 (*b*) Fashion-MNIST数据集 (*c*) CIFAR-10数据集

图13 四种激活函数在三个数据集上的损失值(Loss)

6 结束语

针对基于差分隐私的深度学习隐私保护存在的问题,本文提出了一种基于逐层相关性传播,信息熵和自适应差分隐私的深度学习隐私保护方法RADP. 该方法根据逐层相关性传播算法计算出训练样本的平均特征相关性. 在此基础上,结合信息熵给特征平均相关性自适应地添加噪声. 然后,再使用加噪保护的特征平均相关性为特征自适应地添加噪声. 采用一种与梯度无关的加噪策略,提高了模型的可用性,稳定性与安全性. 最后,通过真实数据集上的实验结果表明RADP有比较高的可用性.

参考文献

[1] 康海燕, 冀源蕊. 基于本地化差分隐私的时序位置发布方案研究[J]. 电子学报, 2022, 50(9):2 222-2232.
KANG H Y, JI Y R. Research on time-serial location data publication based on local differential privacy[J]. Acta Electronica Sinica, 2022, 50(9): 2222-2232. (in Chinese)

[2] 周纯毅, 陈大卫, 王尚, 等. 分布式深度学习隐私与安全攻击研究进展与挑战[J]. 计算机研究与发展, 2021, 58(5): 927-943.
ZHOU C Y, CHEN D W, WANG S, et al. Research and challenge of distributed deep learning privacy and security attack[J]. Journal of Computer Research and Development, 2021, 58(5): 927-943. (in Chinese)

[3] ROCHER L, HENDRICKX J M, DE MONTJOYE Y A. Estimating the success of re-identifications in incomplete datasets using generative models[J]. Nature Communications, 2019, 10: 3069.

[4] 刘艺璇, 陈红, 刘宇涵, 等. 联邦学习中的隐私保护技术[J]. 软件学报, 2022, 33(3): 1057-1092.
LIU Y X, CHEN H, LIU Y H, et al. Privacy-preserving techniques in federated learning[J]. Journal of Software, 2022, 33(3): 1057-1092. (in Chinese)

[5] 刘睿瑄, 陈红, 郭若杨, 等. 机器学习中的隐私攻击与防御[J]. 软件学报, 2020, 31(3): 866-892.
LIU R X, CHEN H, GUO R Y, et al. Survey on privacy attacks and defenses in machine learning[J]. Journal of Software, 2020, 31(3): 866-892. (in Chinese)

[6] SHOKRI R, STRONATI M, SONG C Z, et al. Membership inference attacks against machine learning models[C]// 2017 IEEE Symposium on Security and Privacy. Piscataway: IEEE, 2017: 3-18.

[7] CARLINI N, LIU C, ERLINGSSON Ú, et al. The secret sharer: Evaluating and testing unintended memorization in neural networks[C]//Proceedings of the 28th USENIX Conference on Security Symposium. New York: ACM, 2019: 267-284.

[8] CARLINI N, TRAMER F, WALLACE E, et al. Extracting training data from large language models[EB/OL]. (2021-06-15)[2022-05-20]. https://arxiv.org/abs/2012.07805.

[9] RADFORD A, WU J, CHILD R, et al. Language models are unsupervised multitask learners[EB/OL]. (2019-01-01)[2022-06-09]. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[10] DWORK C, MCSHERRY F, NISSIM K, et al. Calibrating noise to sensitivity in private data analysis[C]//Proceedings of the Third Conference on Theory of Cryptography. New York: ACM, 2006: 265-284.

[11] DWORK C. Differential privacy: A survey of results[C]// Proceedings of the 5th International Conference on Theory and Applications of Models of Computation. New York: ACM, 2008: 1-19.

[12] DWORK C, ROTH A. The algorithmic foundations of differential privacy[J]. Foundations and Trends in Theoretical Computer Science, 2014, 9(3-4): 211-407.

[13] MIRONOV I. Rényi differential privacy[C]//2017 IEEE 30th Computer Security Foundations Symposium (CSF). Piscataway: IEEE, 2017: 263-275.

[14] 陈思, 付安民, 柯海峰, 等. MCDP: 基于神经网络的多集群分布式差分隐私数据发布方法[J]. 电子学报, 2020, 48(12): 2297-2303.
CHEN S, FU A M, KE H F, et al. MCDP: Multi-cluster differential privacy data publishing method based on neural network[J]. Acta Electronica Sinica, 2020, 48(12): 2297-2303. (in Chinese)

[15] DONG J S, ROTH A, SU W J. Gaussian differential privacy[J]. Journal of the Royal Statistical Society Series B: Statistical Methodology, 2022, 84(1): 3-37.

[16] ABADI M, CHU A, GOODFELLOW I, et al. Deep learning with differential privacy[C]//Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2016: 308-318.

[17] MCSHERRY F, TALWAR K. Mechanism design via differential privacy[C]//48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07). Piscataway: IEEE, 2007: 94-103.

[18] BRENDAN MCMAHAN H, RAMAGE D, TALWAR K, et al. Learning differentially private recurrent language models[EB/OL]. (2017-10-17) [2022-05-20]. http://export.arxiv.org/abs/1710.06963.

[19] GEYER R C, KLEIN T, NABI M. Differentially private federated learning: A client level perspective[EB/OL]. (2017-12-20) [2022-05-20]. http://export. arxiv. org/abs/1712.07557.

[20] BASSILY R, SMITH A, THAKURTA A. Private empirical risk minimization: Efficient algorithms and tight error bounds[C]//2014 IEEE 55th Annual Symposium on Foundations of Computer Science. Piscataway: IEEE, 2014: 464-473.

[21] BU Z Q, DONG J S, LONG Q, et al. Deep learning with Gaussian differential privacy[EB/OL]. (2019-11-26) [2022-05-20]. http://export.arxiv.org/abs/1911.11607.

[22] WANG Y X, FIENBERG S E, SMOLA A J. Privacy for free: Posterior sampling and stochastic gradient Monte Carlo[C]//Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37. New York: ACM, 2015: 2493-2502.

[23] LI B, CHEN C Y, LIU H, et al. On connecting stochastic gradient MCMC and differential privacy[EB/OL]. (2017-12-25)[2022-05-20]. https://arxiv.org/pdf/1712.09097.pdf.

[24] KAIROUZ P, MCMAHAN B, SONG S, et al. Practical and private (deep) learning without sampling or shuffling[EB/OL]. (2021-12-10) [2022-05-20]. https://arxiv.org/abs/2103.00039.

[25] PHAN N, WU X T, HU H, et al. Adaptive Laplace mechanism: Differential privacy preservation in deep learning[C]//2017 IEEE International Conference on Data Mining (ICDM). Piscataway: IEEE, 2017: 385-394.

[26] XU C G, REN J, ZHANG D Y, et al. GANobfuscator: Mitigating information leakage under GAN via differential privacy[J]. IEEE Transactions on Information Forensics and Security, 2019, 14(9): 2358-2371.

[27] PAPERNOT N, THAKURTA A, SONG S, et al. Tempered sigmoid activations for deep learning with differential privacy[J]. Proceedings of the AAAI Conference on

Artificial Intelligence, 2021, 35(10): 9312-9321.
[28] ZILLER A, USYNIN D, BRAREN R, et al. Medical imaging deep learning with differential privacy[J]. Scientific Reports, 2021, 11: 13524.
[29] LIU X, ZHOU P, QIU T, et al. Blockchain-enabled contextual online learning under local differential privacy for coronary heart disease diagnosis in mobile edge computing[J]. IEEE Journal of Biomedical and Health Informatics, 2020, 24(8): 2177-2188.
[30] HAN R, LI D, OUYANG J, et al. Accurate differentially private deep learning on the edge[J]. IEEE Transactions on Parallel and Distributed Systems, 2021, 32(9): 2231-2247.
[31] WENG J S, WENG J, ZHANG J L, et al. DeepChain: Auditable and privacy-preserving deep learning with blockchain-based incentive[J]. IEEE Transactions on Dependable and Secure Computing, 2021, 18(5): 2438-2455.
[32] IWAHANA K, YANAI N, CRUZ J P, et al. SPGC: An integrated framework of secure computation and differential privacy for collaborative learning[C]//Data Privacy Management, Cryptocurrencies and Blockchain Technology. New York: ACM, 2021: 89-105.
[33] TORFI A, FOX E A, REDDY C K. Differentially private synthetic medical data generation using convolutional GANs[J]. Information Sciences, 2022, 586: 485-500.
[34] Papernot N, Song S, Mironov I, et al. Scalable private learning with PATE[EB/OL]. (2018-02-24)[2022-05-20]. http://export.arxiv.org/abs/1802.08908.
[35] 谭作文, 张连福. 机器学习隐私保护研究综述[J]. 软件学报, 2020, 31(7): 2127-2156.
TAN Z W, ZHANG L F. Survey on privacy preserving techniques for machine learning[J]. Journal of Software, 2020, 31(7): 2127-2156. (in Chinese)
[36] JAYARAMAN B, WANG L X, EVANS D, et al. Distributed learning without distress: Privacy-preserving empirical risk minimization[C]//Proceedings of the 32nd International Conference on Neural Information Processing Systems. New York: ACM, 2018: 6346-6357.
[37] LU Z G, ASGHAR H J, KAAFAR M ALI, et al. A differentially private framework for deep learning with convexified loss functions[J]. IEEE Transactions on Information Forensics and Security, 2022, 17: 2151-2165.
[38] CHENG A D, WANG J X, ZHANG X S, et al. DPNAS: Neural architecture search for deep learning with differential privacy[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2022, 36(6): 6358-6366.
[39] DING X F, CHEN L, ZHOU P, et al. Differentially private deep learning with iterative gradient descent optimization[J]. ACM/IMS Transactions on Data Science, 2022, 2(4):1-27.
[40] DWORK C, ROTHBLUM G N, VADHAN S. Boosting and differential privacy[C]//2010 IEEE 51st Annual Symposium on Foundations of Computer Science. Piscataway: IEEE, 2010: 51-60.
[41] KIFER D, LIN B R. Towards an axiomatization of statistical privacy and utility[C]//Proceedings of the twenty-ninth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems. New York: ACM, 2010: 147-158.
[42] BACH S, BINDER A, MONTAVON G, et al. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation[J]. PLoS One, 2015, 10 (7): e0130140.
[43] SHANNON C E. A mathematical theory of communication[J]. Bell System Technical Journal, 1948, 27(4): 623-656.

作者简介

康海燕 男, 1971 年 7 月生, 河北灵寿人, 博士, 教授, 研究方向为网络安全与隐私保护等.
E-mail: kanghaiyan@126.com

王骁识 男, 1997 年 12 月生, 北京人, 北京信息科技大学网络空间安全专业在读硕士研究生, 研究方向为网络安全与隐私保护.