

于蓓交接文档说明

- 问题背景

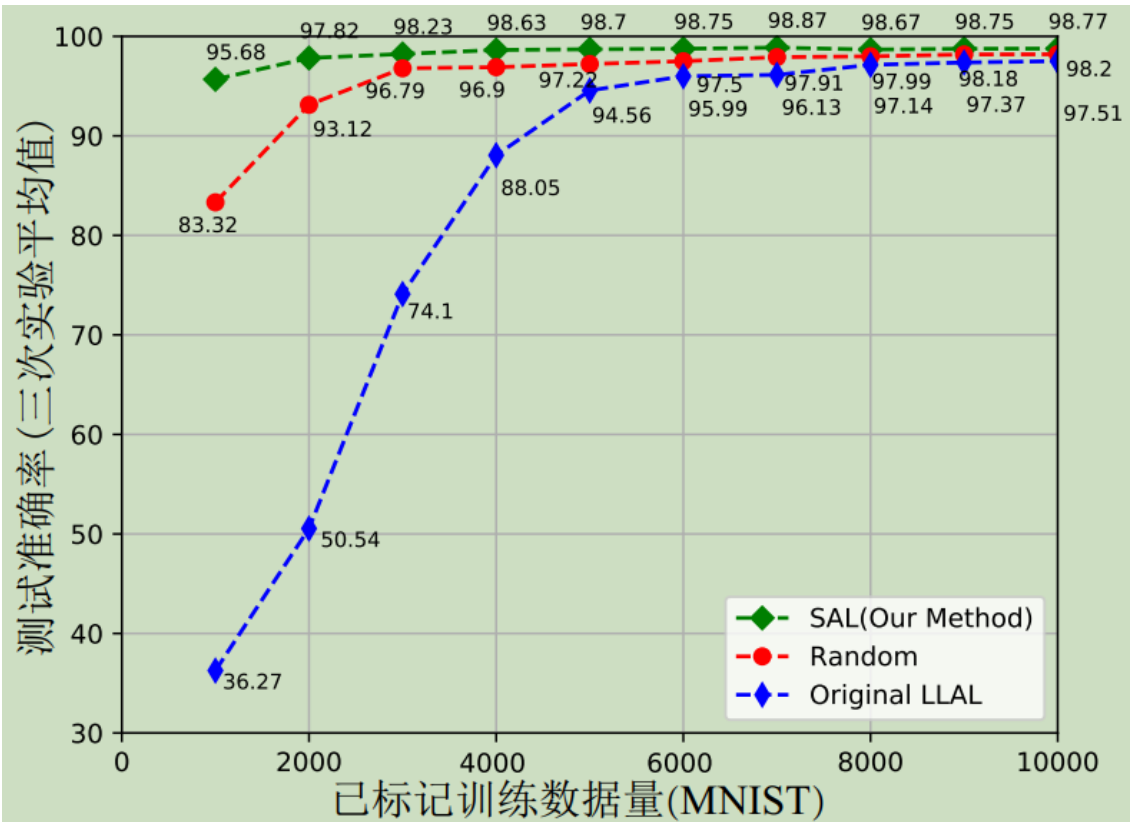
目前基于传统人工网络的主动学习策略研究较多，却仍没有基于脉冲神经网络的主动挑选输入样本的相关研究。同时，由于传统人工网络和脉冲神经网络在特征表达和信息处理机制等方面存在本质上差异，因此现有的主动学习方法很多不能直接用于脉冲神经网络，部分算法在适配脉冲神经网络过程中存在性能较差局限。为了减少训练深度脉冲学习模型所需要的已标记数据量，提升训练模型的效率，本文旨在研究脉冲神经网络上的主动学习方法。

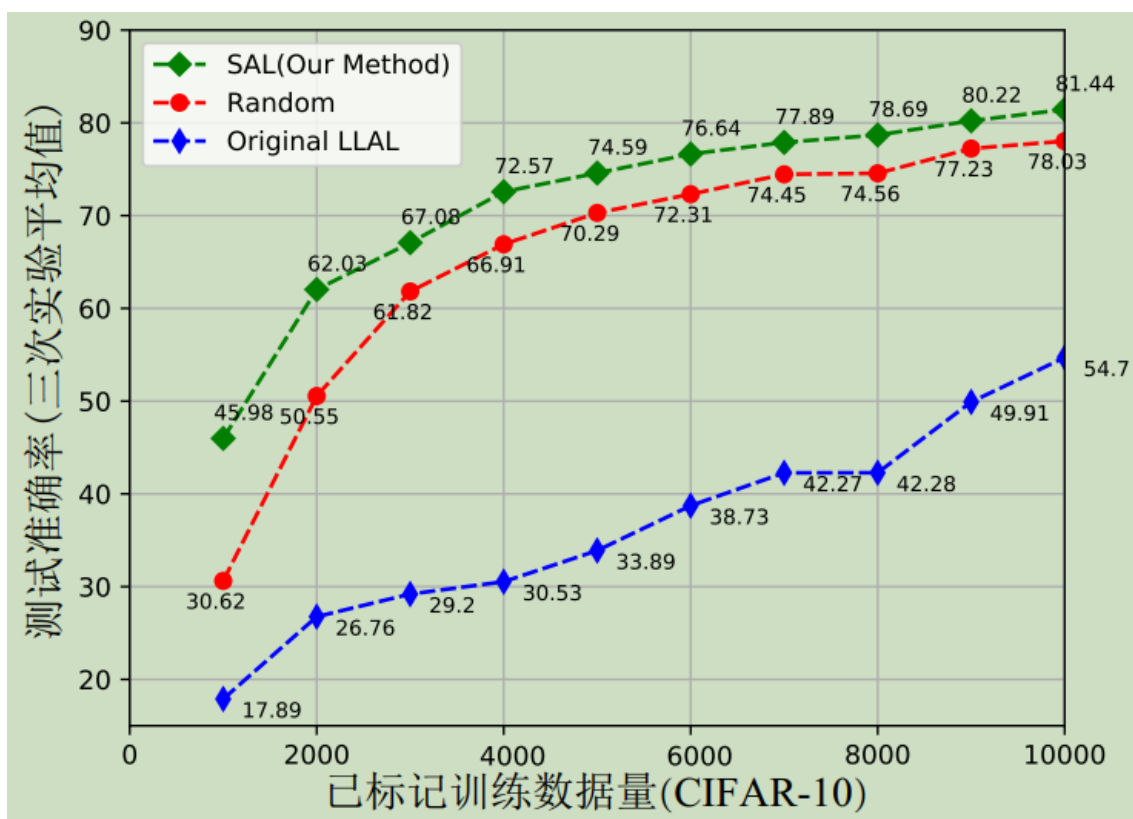
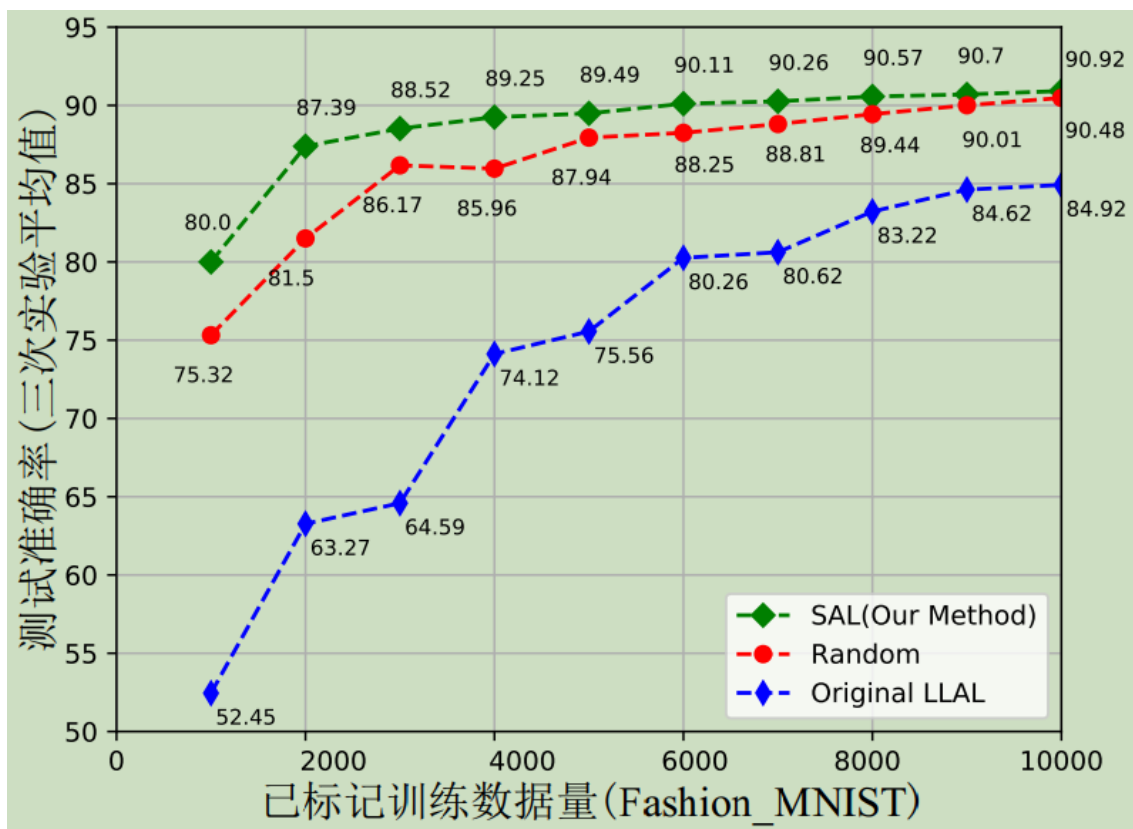
- 解决方法

提出了一种深度脉冲主动学习算法。将主动学习模型嵌入进深度脉冲神经网络中，使用训练得到的主动损耗预测模型预测未标记池中样本的损失值，通过n次主动学习迭代筛选损失值最大的k*n个样本，使用优选的数据训练模型使模型能够使模型迅速收敛且最大程度提高其泛化性和鲁棒性，当训练数据较少时更能体现出本算法优势。

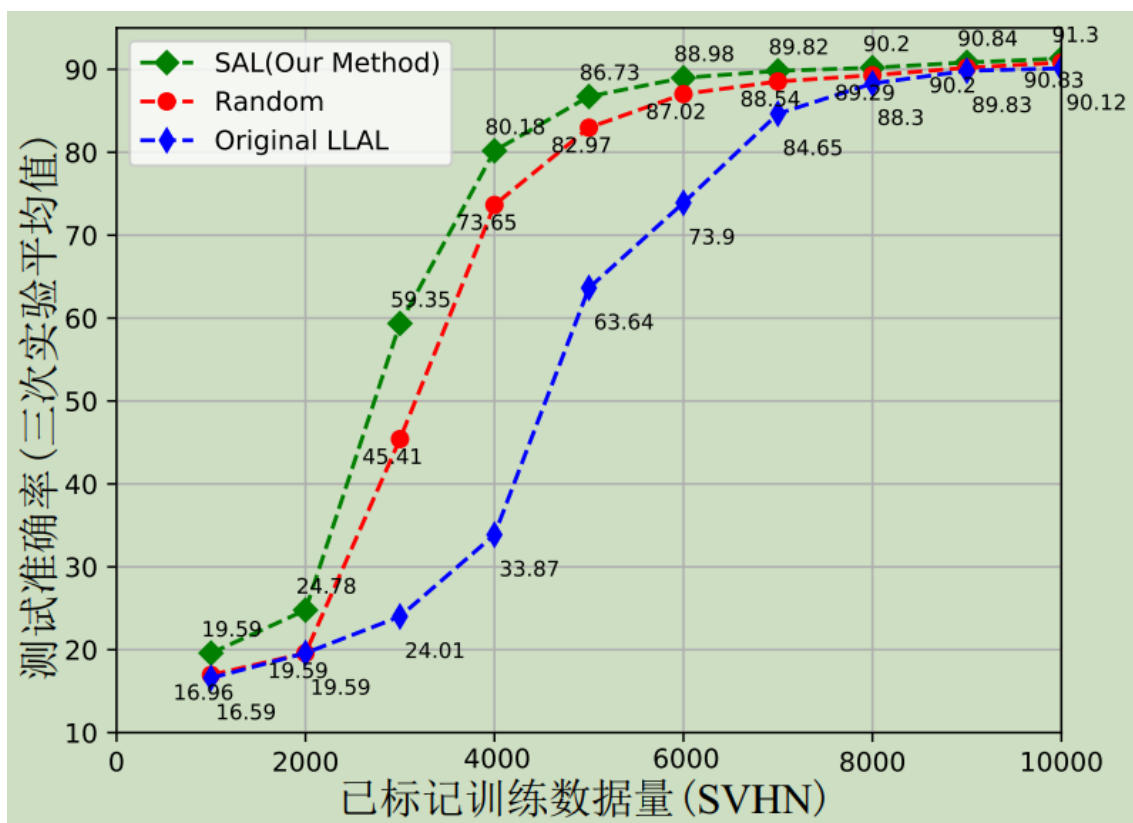
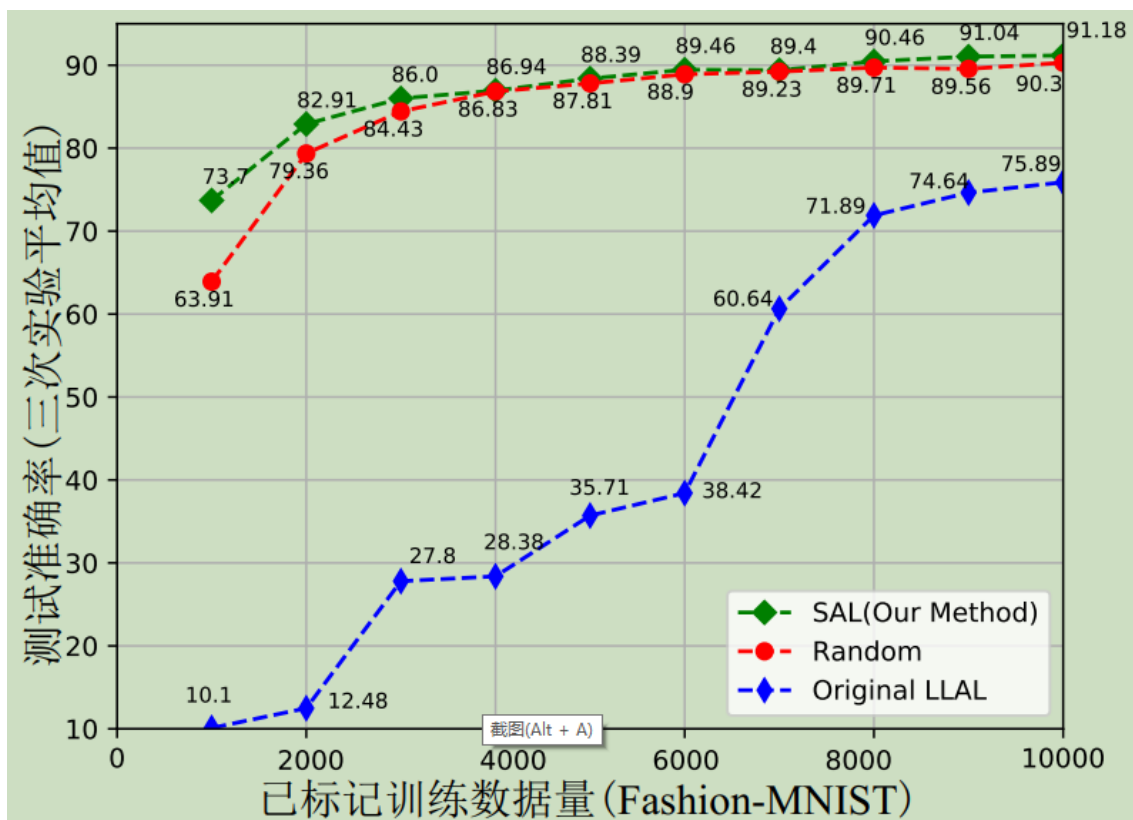
- 实验结果记录

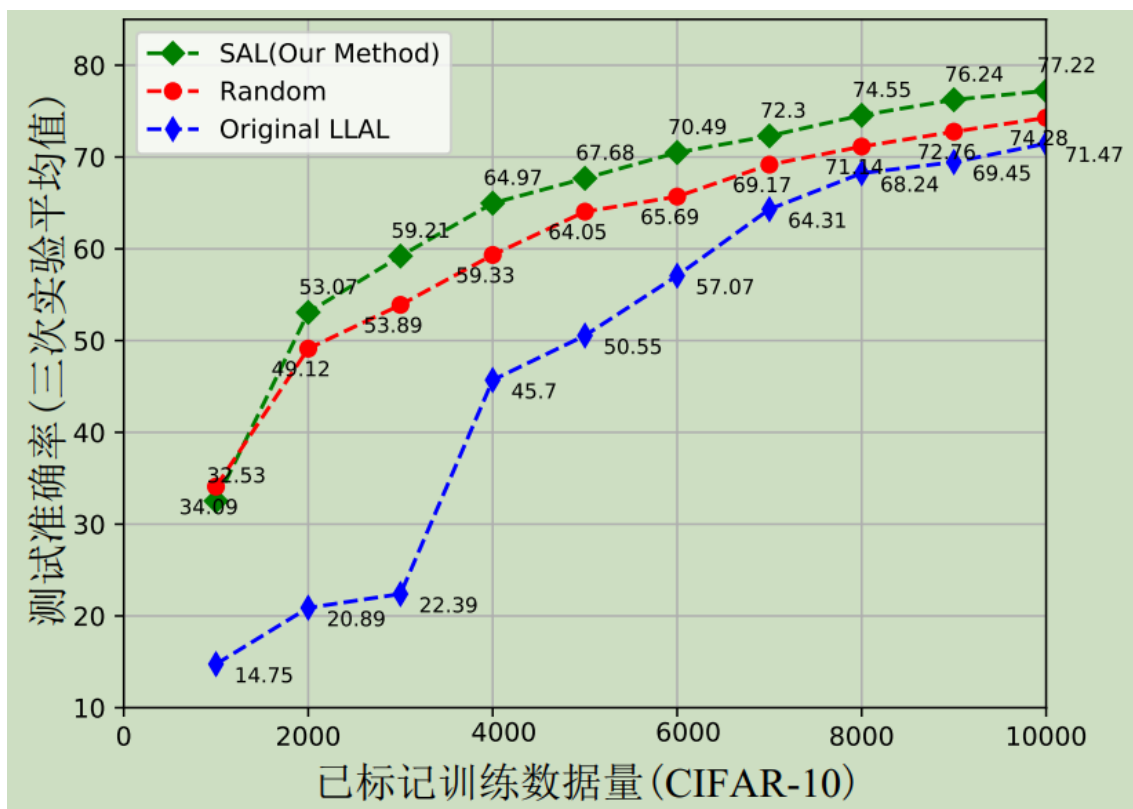
在CIFARNet模型上的主动学习结果：





在ResNet模型上的实验结果：





- 是否有相关论文在投或被接受

有，小修修回，邮件已显示线上接收成功。

论文latex链接: <https://www.overleaf.com/9253177166ctyzgbgpphpj>

- 代码运行说明

整个代码由两个大文件夹组成，其中，Jelly-CIFARNet-MNIST中的模型适合跑28\times28尺寸的输入数据，如：MNIST、Fashion-MNIST。Jelly-Grad-SpikingNN中的模型适合32*32尺寸的输入数据，如：cifar10、SVHN。

在每一个文件夹中都同时包含CIFARNet和ResNet模型：

- CIFARNet模型的文件分别为：

- main_jelly_active.py main函数，无需参数直接运行该代码即可开始模型训练
 - models/Jelly_active.py 为5层的SNN分类器
 - mdoels/Jelly_active_lossnet.py 为对应的主动损失预测模型
 - configs/CIFARNet_config.py 为模型训练相关参数

- ResNet模型的文件分别为：

- main_jelly_resnet.py main函数，无需参数直接运行该代码即可开始模型训练
 - models/Jelly_resnet.py 为20层的SNN分类器
 - mdoels/Jelly_resnet_lossnet.py 为对应的主动损失预测模型
 - configs/Resnet_config.py 为模型训练相关参数

需要注意在运行SAL（深度脉冲主动学习）代码时，将main文件中190多行的内容改为SAL对应的部分（已经标注好了），将330多行的部分对应也使用SAL的平方差公式（已经标注好了）。同样的在运行LLAL（原始主动学习基线）时，同样前面两个位置对应修改为LLAL的内容（已经标注好了）。

注意每次运行代码前都将main文件中创建模型保存目录的路径重新修改，以防止覆盖之前训练好的模型。

注意每次运行代码前都检查配置文件中经常修改的参数是否设置好，如：数据集尺寸，实验运行次数、主动学习迭代次数、每次选择的数据量大小k

- 代码运行依赖的环境

tqdm	4.54.1
torch	1.7.1+cu101
numpy	1.19.4
torchvision	0.8.2+cu101
sklearn	0.0
matplotlib	3.3.3
spikingjelly	0.0.0.0.3
python	3.7
CUDA	release 10.0, V10.0.130
conda	4.10.3