对论文《SPIKFORMER: WHEN SPIKING NEURAL NETWORK MEETS TRANSFORMER》的复现

摘要

原文作者同时考虑到两种生物学上合理的结构：脉冲神经网络和注意力机制。脉冲神经网络为深度学习提供了一种节能的且可事件驱动的范式，注意力机制则具有捕获特征依赖关系的能力，使 Transformer 架构能够获得良好的性能。从直觉上探索这两种结构的结合是有意义的。原文作者利用脉冲神经网络的自注意力能力和生物学特性，提出了一种新颖的脉冲注意力机制 (SSA)，以及新的脉冲 Transformer 框架 (Spikformer)。Spikformer 模型中的 SSA 机制通过使用脉冲形式的 Qurry，Key 和 Value 矩阵来构建稀疏视觉特征，且不需要 softmax 运算。由于其计算具备稀疏性且避免了乘法运算，SSA 具有高效低耗的特点。实验结果表明在图像分类任务上，使用 SSA 的 Spikformer 的性能优于其他最先进的类似 SNN 框架。

关键词：脉冲神经网络；自注意力；图像分类

1 引言

深度学习已经解决了计算机视觉，语音识别和自然语言处理领域的许多问题。随着神经网络的规模不断扩大，传统深度神经网络由于包含大量的浮点数运算，其训练成本，包括硬件设备和能源开销，都成为了深度神经网络进一步发展的瓶颈之一。而新一代受大脑工作方式所启发的脉冲神经网络有望弥补这一问题。

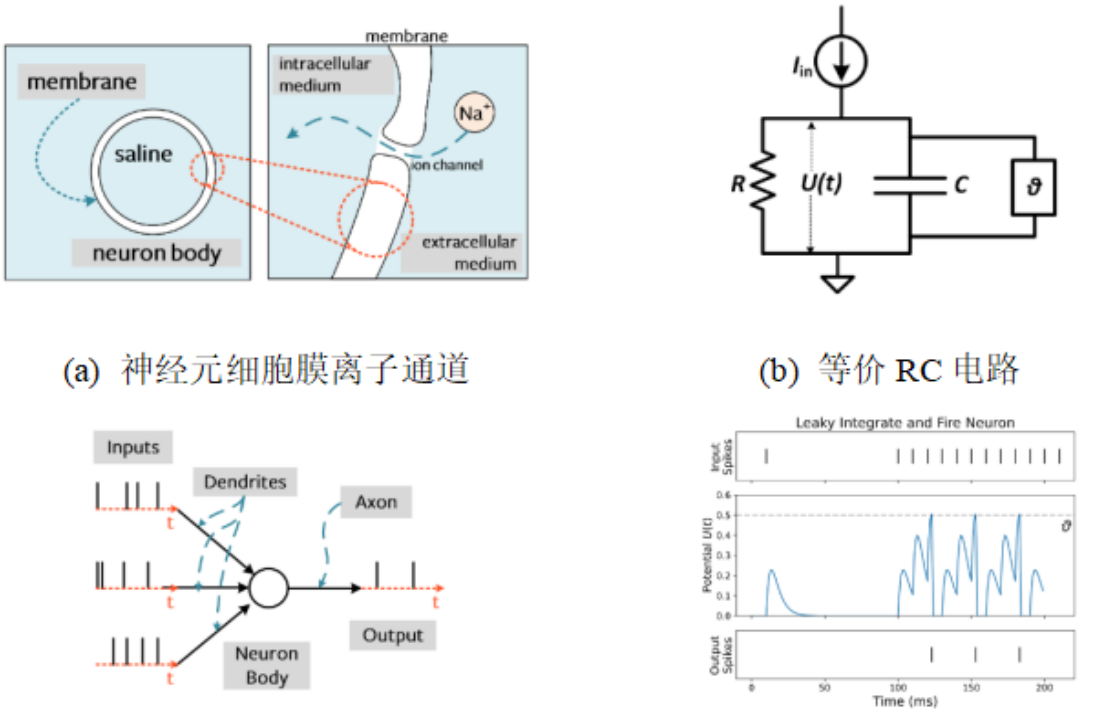
从 2012 年到 2019 年，运行最先进的深度学习模型所需要的计算能力以每年 10 倍的速度增长 [14]。数据生成的速度同样呈指数级增长。OpenAI 推出的大语言模型主干网络 GPT-3 包含了 1750 亿个可学习参数，训练消耗估计约为 190,000kWh [1] [3]。而反观人类自己的大脑，通常仅在 12 20W 的功率范围内运行。如果我们的大脑散发的热量也和最先进的深度学习模型一样多，那么人类早就在被人工智能灭亡之前，被自然选择所灭绝了 [9]。于是，在大脑工作特性的提示下，科研人员提出了类脑脉冲神经元建模和基于该神经元的脉冲神经网络 (Spiking Neural Networks，SNN)。

与使用连续十进制值来传递传统深度学习的模型不同，SNN 使用离散脉冲序列来计算和传输信息。脉冲神经元接受连续值并将其转换为脉冲序列。使用脉冲神经网络架构的深度学习模型，由于其脉冲神经元活动的时序性和稀疏性，在计算总量和能源消耗上展现出了较强的优越性。不仅有望解决当前的设备瓶颈问题，也在边缘设备部署领域具有巨大优势。

2 相关工作

2.1 脉冲神经网络

与使用连续十进制传递信息的传统深度学习模型不同，SNN 使用离散的脉冲序列来计算和传输信息。脉冲神经元接受连续值并将其转化为脉冲序列，常见的模型包括 Integrate-and-Fire(LIF) 神经元模型 [16]，PLIF 神经元模型 [4] 等。得到深度 SNN 模型有两种主要的方法，分别是将 ANN 转化为 SNN 和直接训练。若使用将 ANN 转化为 SNN 的方法，可将任务表现优秀的高性能 ANN 模型中的 ReLU 激活层使用脉冲神经元来替换，将其转化为 SNN [10] [15]。转换后的 SNN 需要较大的时间步长才能准确逼近 ReLU 激活层，这会导致整个模型有较大的延迟 [5]。如果使用直接训练的方法，SNN 可在模拟时间步的维度上展开，并以随时间反向传播的方法来训练。由于脉冲神经元的事件触发机制，其脉冲是不可微的，为了解决这个问题，可以使用替代梯度函数来进行反向传播 [8]。现在已有很多模型从 ANN 转化为了 SNN 形式，但是关于使用 SNN 实现自注意力机制的研究还是空白。



(a) 神经元细胞膜离子通道　(b) 等价 RC 电路

(c) 脉冲输入神经元产生输出脉冲　(d) 达到阈值后的输出脉冲

图 1. LIF 神经元模型

作为 SNN 的基本单元，脉冲神经元接受电流并积累膜电位，与放电阈值比较之后确定是否产生脉冲。论文作者在此工作中统一使用 LIF 脉冲神经元模型，如图 1所示。LIF 的动态模型可以使用一下公式描述：

$$H[t] = V[t-1] + \frac{1}{\tau}(X[t] - (V[t-1] - V_{reset})) \tag{1}$$

$$S[t] = \Theta(H[t] - V_{th}) \tag{2}$$

$$V[t] = H[t](1 - S[t]) + V_{reset}S[t] \tag{3}$$

其中 τ 是膜时间常数，$X[t]$ 表示在每个时间步 t 的输入电流。当膜电位 $H[t]$ 累积并超过了放电阈值 V_{th} 时，脉冲神经元将发出脉冲 $S[t]$。$\Theta(v)$ 是 Heaviside 阶跃函数，当 $v \geq 0$ 时，$\Theta(v) = 1$；否则 $\Theta(v) = 0$。$V[t]$ 表示触发事件后的膜电位，如果神经元没有产生脉冲，则其膜电位等于 $H[t]$，否则就等于复位电位 V_{reset}。

2.2 Vision Transformer

对于图像分类任务，标准视觉 Transformer（Vision Transformer, ViT）包括块分割模块，变换编码器层和线性分类头。Transformer 编码器由自注意力层和多层感知机层组成。自注意力机制是 ViT 模型成功的核心。通过查询矩阵 Query 和键矩阵 Key 的点积以及 softmax 函数对图像块特征值进行加权，自注意力可以捕获全局依赖性和兴趣表征 [7] [11]。现在已有一些工作来改进 ViT 的结构。使用卷积层进行图像批块分割已被证明能够有效加速收敛并环节 ViT 的数据匮乏问题 [17] [6]。有一些方法旨在降低自注意力的计算复杂度或者提高其视觉依赖性建模能力 [13] [18] [12]。该论文重点探讨 SNN 中自注意力机制的有效性，并提出了有效的脉冲 Transformer 模型用于完成图像分类任务。

3 文章方法

3.1 整体结构

论文作者提出的 Spikeformer 框架，其整体结构与 ViT 基本保持一致，由 SPS、EncoderBlock 和 ClassificationHead 三个主要部分组成，其中 EncoderBlock 包含了脉冲注意力 SSA 和多层感知机模块，如图 2所示。EncoderBlock 模块可以根据实际任务需求修改数量，以达到模型与数据相匹配。

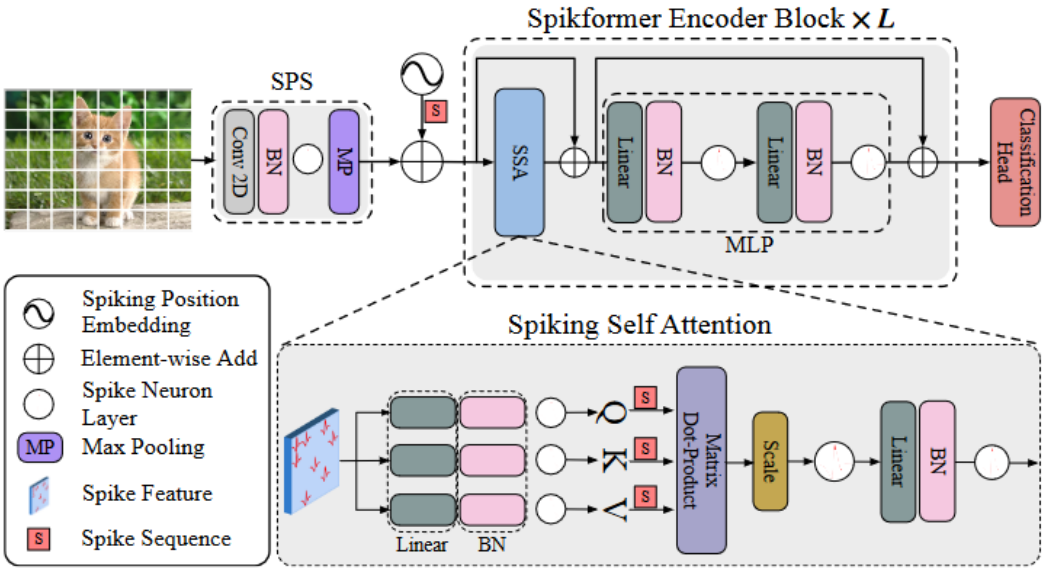


图 2. SpilingFormer 整体结构

3.2 脉冲输入分块 (SPS)

给定二维图像序列 $I \in \mathbb{R}^{T\times C\times H\times W}$，SPS 模块将其线性投影到 D 维的脉冲形式的特征向量上，并将其分割成 N 个脉冲形式的 patchs 序列 X。SPS 可以包含多个块。类似于 ViT，在每个 SPS 块中应用卷积层，将归纳偏差引入 Spikformer。公式 4表示其数据流变化，其中 SN 表示脉冲神经元层。

$$x = MP(SN(BN(Conv2D(I)))), \quad I \in \mathbb{R}^{T\times C\times H\times W}, x \in \mathbb{R}^{T\times N\times D} \tag{4}$$

3.3 脉冲位置编码 (SRPE)

浮点数形式的位置编码无法在 SNN 中使用。论文作者选择采用条件位置嵌入生成器 [2] 生成脉冲形式的相对位置编码（SRPE），并将 SRPE 添加到 patches 序列 X 中得到 X_0。位置编码生成器包含一个核大小为 3 的二维卷积层（Conv2d）、批归一化（BN）和脉冲神经元层（SN）。然后我们将 X_0 传递到 EncoderBlock 中。数据流变化如公式 5所示。

$$SRPE = SN(BN(Conv2d(x))), \quad SRPE \in \mathbb{R}^{T\times N\times D} \tag{5}$$

3.4 脉冲自注意力机制 (SSA)

编码器是整个架构的最主要组成部分，其中包含了脉冲自注意力 (SSA) 机制和多层感知机模块。回顾普通的标准自注意力 (VSA)，给定输入特征序列 $X \in \mathbb{R}^{T\times N\times D}$，ViT 中的注意力机制具有三个浮点数类型的关键组件，即查询 (Query)，键 (Key) 和值 (Value)，其矩阵分别用 Q_F, K_F, V_F 表示。这三个矩阵可以是通过三个参数可学习 $W_Q, W_K, W_V \in \mathbb{R}^{D\times D}$ 矩阵来计算：

$$Q_F = XW_Q, K_F = XW_K, V_F = XW_V \tag{6}$$

其中 F 表示数据由浮点数形式表示。普通自注意力的输出可以表示为：

$$VSA(Q_F, K_F, V_F) = Softmax(\frac{Q_F K_F^T}{\sqrt{d}})V_F \tag{7}$$

其中 $d = D/H$ 是注意力机制中一个注意力头的特征尺寸，H 表示注意力头的数量。但是，标准的自注意力的计算方式并不能直接适用于 SNN 中。原因有两点：(1) 浮点数矩阵乘法运算和 Softmax 函数运算包含了指数计算和除法运算，不符合 SNN 的计算规则。(2)VSA 的序列长度的空间和时间复杂度不满足 SNN 对于高效计算的要求。于是，论文作者定义了一种适用于 SNN 的脉冲自注意力计算方式。其结构与计算方式如图 2底部部分和图 3(b) 所示。

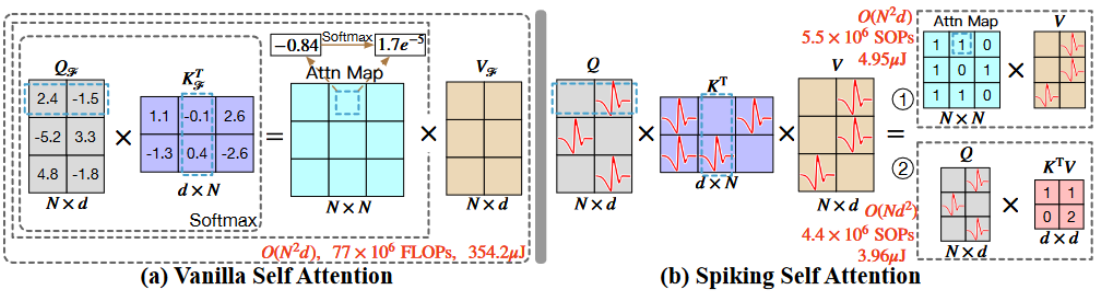


图 3. 标准自注意力 (VSA) 与脉冲自注意力 (SSA)

在脉冲自注意力机制中，首先通过可学习矩阵计算查询矩阵 Q，键矩阵 K 和值矩阵 V，然后各自通过不同的脉冲神经元层，将其转化为脉冲序列：

$$Q = SN_Q(BN(XW_Q)), K = SN_K(BN(XW_K)), V = SN_V(BN(XW_V)) \tag{8}$$

其中 $Q, K, V \in \mathbb{R}^{T\times N\times D}$，T 代表脉冲序列化的时间步长。在注意力计算矩阵的过程中，应该使用纯脉冲形式的矩阵，即矩阵元素仅包含 0 或 1。同时，论文加入了一个缩放因子 s 用来控制矩阵相乘结果的最大值，这个缩放因子 s 并不会影响 SSA 的性质。如图 2所示，脉冲自注意力机制定义如下：

$$SSA'(Q, K, V) = SN(QK^TV * S) \tag{9}$$

$$SSA(Q, K, V) = SN(BN(Linear(SSA'(Q, K, V)))) \tag{10}$$

此处定义的单头 SSA 可以很容易拓展到多头 SSA，且在每个时间步上独立进行计算。如公式 9所示，SSA 取消了 softmax 来归一化注意力矩阵，取而代之的是直接对矩阵 Q, K, V 进行相乘。论文作者认为由于脉冲序列的特性，在 SSA 中 softmax 运算是不必要的，它甚至会阻碍 SNN 中自注意力机制的实现。脉冲神经元层产生的脉冲序列输出天然具有非负特性，从而可以直接产生非负注意力图。SSA 仅聚合了相关的特征，而忽略了不相关的信息。因此 SSA 不再需要 softmax 来确保注意力图的非负性。此外，与浮点型数据流的 ANN 相比，SNN 中的注意力数据流全部为脉冲形式，包含的信息要较为有限，所以浮点数形式的数据流实际上对于 SNN 来说是多余的。

3.5 多层感知机模块 (MLP)

单个编码器中，多层感知机由两组线性模块组成，每组线性模块都包含了一个线性层，一个批归一化层和一个脉冲神经元层。同时在 MLP 前后建立残差链接。

3.6 编码器 (EncoderBlock)

编码器部分由脉冲注意力模块和多层感知机模块组成，并分别应用残差连接方式进行连接。其数据输入输出表达式如下所示：

$$X_0 = x + SRPE, \quad X_0 \in \mathbb{R}^{T\times N\times D} \tag{11}$$

$$X'_l = SSA(X_{l-1}) + X_{l-1}, \quad X'_l \in \mathbb{R}^{T\times N\times D}, l = 1...L \tag{12}$$

$$X_l = MLP(X'_l) + X'_l, \quad X_l \in \mathbb{R}^{T\times N\times D}, l = 1...L \tag{13}$$

3.7 分类头 (CH)

经编码器编码处理后的特征向量在输入分类头后，先经过全局平均池化层，然后通过线性层进行分类结构的输出。

$$Y = CH(GAP(X_L)) \tag{14}$$

4 复现细节

4.1 与已有开源代码对比

原论文作者已在 GitHub 开源其实验所使用的代码, 源码地址为 https://github.com/ZK-Zhou/spikformer 。其中分别包括了适用于 CIFAR10/100, CIFAR10DVS 和 ImageNet 图像分类数据集的代码。考虑到自身硬件资源条件, 选择对 CIFAR10/100 数据集图像分类工作进行复现。

在复现过程中, 根据原论文提供的细节和源码, 除随机参数外, 其他设置和超参数上与原文严格对齐。对于 CIFAR10/100 数据集, 选择使用 Spikformer-4-384 结构, 即 EncoderBlock 模块重复 4 次, 特征向量维度为 384。同样分类头保持不变, 使用大小为特征向量维度-类别总数的线性层, 即 Linear(384,10/100)。脉冲序列时间步为 4, 使用学习率 5e-4, 训练周期 400 轮。损失函数使用交叉熵函数, 优化器选择 SGD。实验得到了与原论文相一致的结果, 详细结果见本报告第 5 部分。

原论文源码是在 Timm 开源库提供的 Transformer 框架上进行修改而来, 同时其脉冲神经元使用论文作者课题组维护的开源库 SpikingJelly 实现。站在巨人的肩膀上, 本次课程实践中对于原论文的改进工作同样基于原论文作者开源代码和上述开源框架修改实现。

4.2 实验环境

实验平台操作系统为 Linux5.4.0 (Ubuntu9.3.0), 训练使用硬件设备为单块 TitanX, 环境部分配置如下:

apex == 0.9.10
cupy == 12.2.0
nvidia-cuda == 11.4
nvidia-cudnn == 8.5.0.96
python == 3.10.12
spiklingjelly == 0.0.0.0.14
timm == 0.6.13
torch == 2.0.1
torchvision == 0.15.2

4.3 改进工作一

原论文所提出的 Spikformer 框架的详细结构在第三部分详细介绍了,其整体结构如图 2所示。可以观察到, 由于 EncoderBlock 模块中引入了 ResNet 残差连接, 所以实际上会导致编码器中, 数据流不再是纯脉冲形式的。这会导致在计算过程中, 脉冲数据流的纯逻辑运算 (这里指一个浮点数乘 0 或者乘 1) 变为浮点数乘法运算。这与我们设计 SNN 的初衷相违背。如图 4所示。

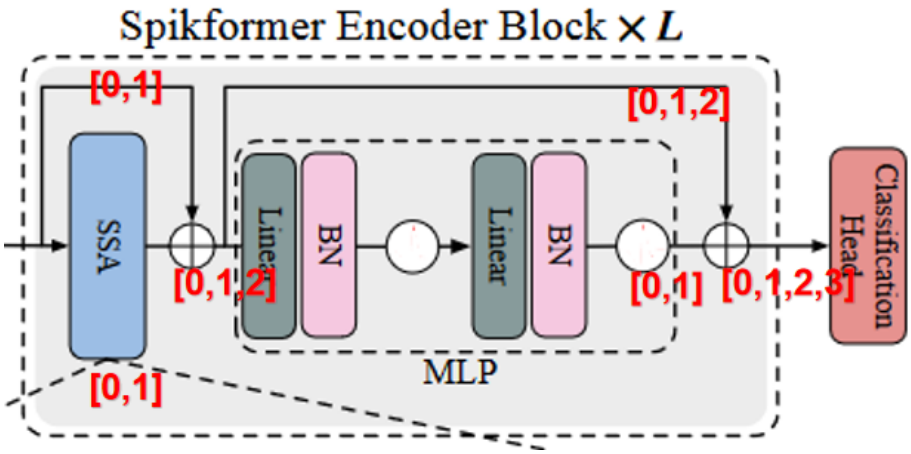


图 4. 残差连接导致编码器产生非脉冲数据

针对这个情况，本课程实践调整了残差连接的位置，对神经网络各层的位置和结构做出修改。修改逻辑如图 5所示。使用修改后的网络结构，可以确保特征向量数据在计算过程中，始终以纯脉冲数据流的形式存在。

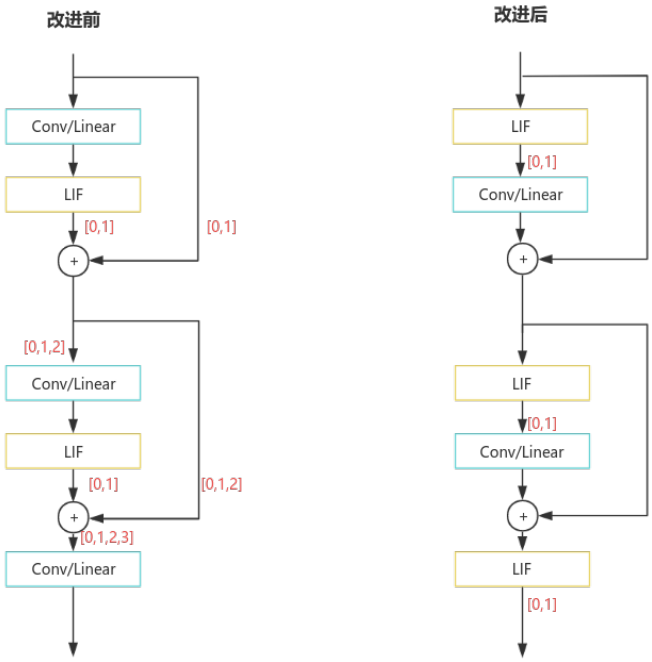


图 5. 网络结构修改逻辑示意

根据此修改逻辑，本课程实践对原论文模型中的的 SPS 模块和 EncoderBlock 模块做出修改，修改前后的如图 6和 7所示。

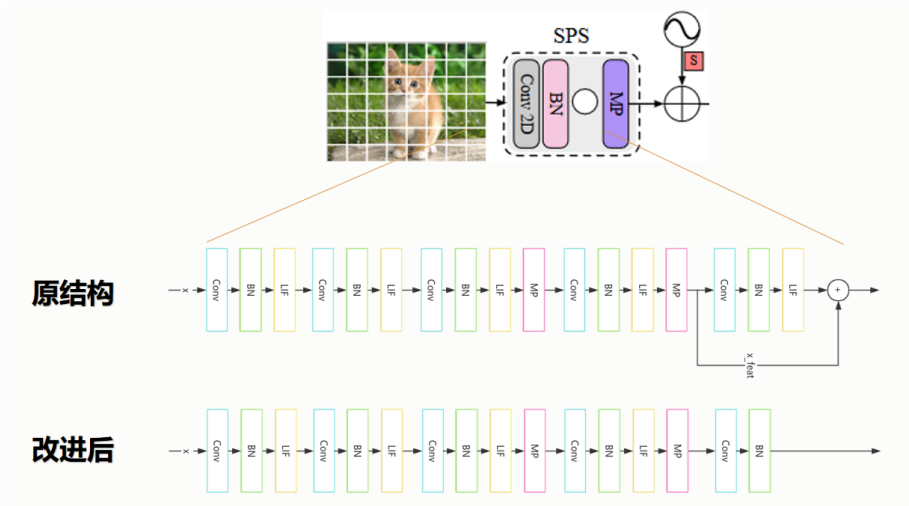


图 6. SPS 模块修改示意图

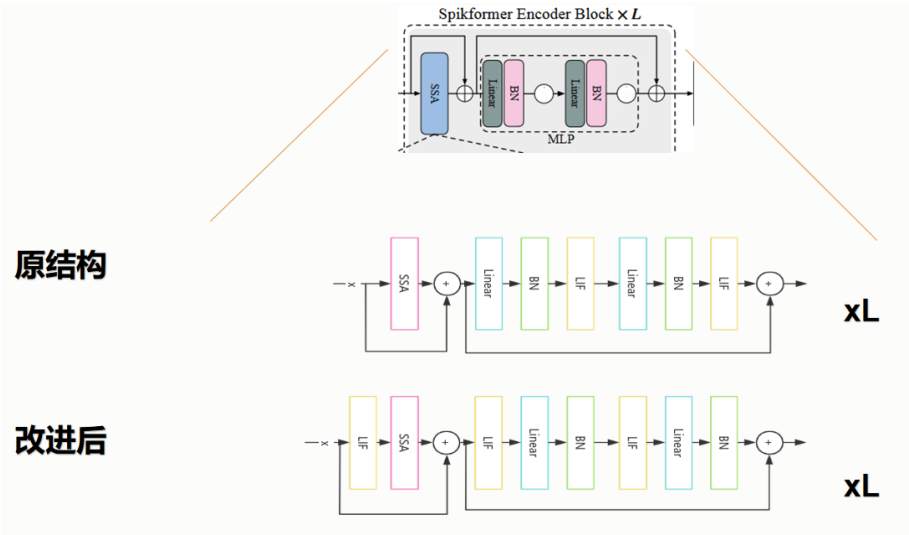


图 7. EncoderBlock 修改示意图

4.4 改进工作二

由于在前文中提出了纯脉冲化的神经网络结构，实验结果表明分类准确度出现了一定程度的降低。于是考虑在 EncoderBlock 中使用卷积层替代线形层，以提取图像更深层次的特征。实验结果在本报告第 5 部分详细阐述。结果表明，在使用了卷积层之后，最终结果的准确度出现了一定程度的提高，但是虽然整体网络的参数量没有增加，但是计算复杂度出现了较大的增加。在实践过程中的直接体现就是相同数据集上的训练耗时明显变多。这与 SNN 简化计算的初衷是相违背的，而复杂计算带来了少量涨点实际上没有太大的意义。

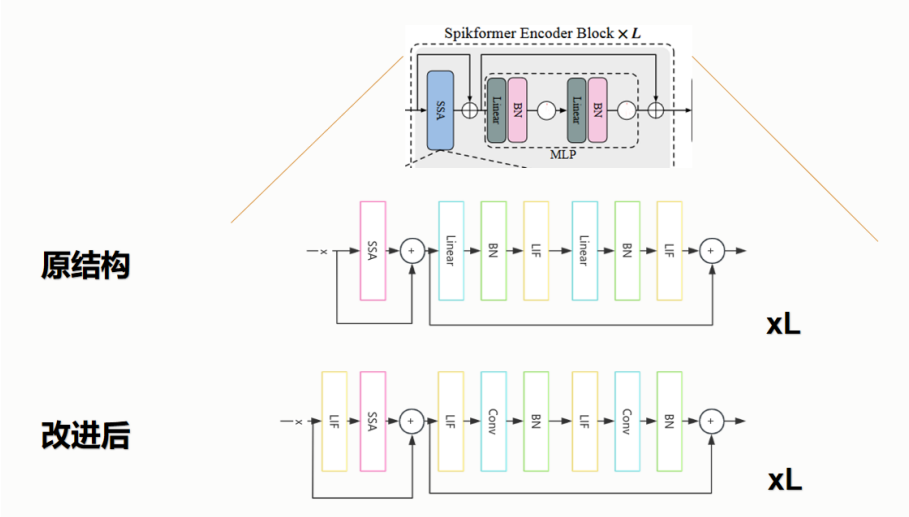


图 8. EncoderBlock 中使用卷积层替换

4.5 改进工作三

有工作 [4] 提出，对于脉冲神经网络，全局最大池化 (GMP) 往往比全局平均池化 (GAP) 能取得更好的效果。直觉上，考虑到脉冲神经元积累电荷，达到阈值则释放脉冲的工作特性，最大池化层在生物可解释型的层面上与脉冲神经网络也有更好的契合度。在原论文中，模型主体结构中使用最大池化层来提取特征，但是在分类头中原文作者使用了全局平均池化层-线形层的结构。于是考虑尝试将分类头中平均池化替换为最大池化，替换工作分别在不使用纯脉冲结构和使用纯脉冲修改两种情况下进行实验。不过遗憾的是，在不改变模型整体框架的前提下，暂时还没能实验得到有意义的结果。在分类头中对特征向量使用池化层或许会损失大量信息，只能认为这是一个毫无意义的失败的修改。

5 实验结果分析

5.1 原论文复现实验结果

对原论文复现的实验结果，如表 1所示：

表 1. CIFAR10/100 数据集图像分类准确率 (%)

模型结构	参数量 (M)	时间步数	CIFAR10	CIFAR100
Spikformer-4-384	9.32	4	95.2	78.1

复现结果与原论文结果基本一致，在 CIFAR10 数据集上达到了 95.2% 的准确率，在 CIFAR100 数据集上到达了 78.1% 的准确率。同时，对时间步，编码器个数和特征向量维度三个方面进行消融实验，结果如表 2和表 3所示。可以看到，当 SNN 的时间步增加时，准确率呈现明显的上升趋势。而当时间步仅为 1 时，Spikformer 依然表现出较好的能力，这说明这个模型结构在时间步数上具有较强的鲁棒性。同时，编码器数量以及特性向量维度大小也

会对模型整体性能产生影响。在不出现过拟合的程度下，编码器越多，特性向量维度越大，分类结果就越好。

表 2. 时间步消融实验结果

模型结构	时间步数	CIFAR10	CIFAR100
Spikformer-4-384	1	93.5	74.4
Spikformer-4-384	2	93.6	76.3
Spikformer-4-384	4	95.2	78.1

表 3. 模型结构消融实验结果

模型结构	参数量 (M)	CIFAR10	CIFAR100
Spikformer-4-256	4.15	93.9	75.9
Spikformer-2-384	5.76	94.8	77.0
Spikformer-4-384	9.32	95.2	78.1

5.2 改进后实验结果

本课程报告中提出了三种改进思路，具体方法在第 4 部分有详细阐述。改进后的的实验结果如表 4所示：

表 4. 改进前后对比表

原论文	复现	纯脉冲化	编码器卷积	最大池化	CIFAR10 准确率 (%)
√					95.5
	√				95.2
		√			92.3
		√	√		95.6
				√	-
		√		√	-

在对原文的模型结构进行纯脉冲化修改之后，准确率略微降低，但是依然有 92.3% 的准确率。推断在将数据流纯脉冲话之后，损失的部分信息导致准确率下降。在对模型结构进行纯脉冲化的修改的基础上，将编码器中多层感知机模块中的线性层修改为卷积层，增加模型的特征提取能力，取到了涨点的效果。但是由于卷积层导致的运算增加，模型训练时间也明显增加。这与设计高效 SNN 的初衷相违背。实验结果表明，在不修改原论文模型结构的情况

下，使用纯脉冲结构会导致模型能力减弱，即体现为分类准确率的下降。最大池化层在模型主体结构中能起到良好的作用，但是将分类头中的平均池化修改为最大池化并不能得到有价值的结果。整体而言，对比结构改进带来的运算简化，较小的分类准确率降低时可以接受的。进一步探索更加合适的模型结构也许可以得到两全其美的结果。

6 总结与展望

本次课程作业选择对 Spikformer 的工作进行复现。在阅读论文及其源码之后，实验设置于原论文对齐，得到了和原论文一致的结果。同时在对其原论文提出的结构进行部分修改之后，得到了略微更加准确的结果。

脉冲神经网络组成的 CNN 框架以及 ViT 框架都展现出了良好的图像特征提取能力，在较低的能耗下，表现出较高的图像分类能力。以其为主干网络，SNN 同样可以用于计算机视觉领域更进一步的目标检测、图像分割甚至生成任务中，能够以极低的能耗完成训练及推理，对于视觉模型的应用具有很强的现实意义。同时 SNN 结构可以处理神经形态数据集，并通过编码方式进一步展示出其矩阵稀疏性的优势。这个领域值得继续深入研究。原论文同时也在 CIFAR10DVS 神经分类数据集上得到了 SOTA 的结果，但是由于其选择了一种取巧的方式，将对神经形态数据集转化为了普通图像数据集进行处理。且为原论文为了突出其所提出的工作的仅需要极少量时间步就能实现高效 SNN 的优点，没有选择使用脉冲编码方式对原始输入数据进行处理。

对于 SNN 未来有三个主要的方向还值得探索：（1）SNN 目前在图像分类和目标检测任务上取得了优秀的成果，但是还可以进一步在其他任务上进行应用。例如图像分割，CLIP 甚至大模型等。同时 SNN 由于其脉冲序列所带来的低能耗特性，对于边缘设备上部署具有很强的现实意义。（2）SNN 由于其时间特性和稀疏性，非常适合用来处理基于事件相机 (Event Camera) 的数据。事件相机领域目前也正在快速发展中，这两个领域其数据特征天然匹配，其所产生的高效低耗的实践值得深入探索。（3）脉冲神经元的积累-放电的运行模式，天然契合以忆阻芯片为代表的存算一体芯片。在未来类脑计算芯片能承载较大参数量之后，基于 SNN 的深度学习算法可以快速在类脑芯片完成部署。由此也许能够打破传统计算机冯诺依曼结构带来的数据和能耗瓶颈。

参考文献

[1] Lasse F Wolff Anthony, Benjamin Kanding, and Raghavendra Selvan. Carbontracker: Tracking and predicting the carbon footprint of training deep learning models. *arXiv preprint arXiv:2007.03051*, 2020.

[2] Xiangxiang Chu, Zhi Tian, Yuqing Wang, Bo Zhang, Haibing Ren, Xiaolin Wei, Huaxia Xia, and Chunhua Shen. Twins: Revisiting the design of spatial attention in vision transformers. *Advances in Neural Information Processing Systems*, 34:9355–9366, 2021.

[3] P Dhar. The carbon impact of artificial intelligence. *Nat Mach Intell 2*, 32(11):423–425, 2020.

[4] Wei Fang, Zhaofei Yu, Yanqing Chen, Timothée Masquelier, Tiejun Huang, and Yonghong Tian. Incorporating learnable membrane time constant to enhance learning of spiking neural networks. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2641–2651, 2020.

[5] Bing Han, Gopalakrishnan Srinivasan, and Kaushik Roy. Rmp-snn: Residual membrane potential neuron for enabling deeper high-accuracy and low-latency spiking neural network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 13558–13567, 2020.

[6] Ali Hassani, Steven Walton, Nikhil Shah, Abulikemu Abuduweili, Jiachen Li, and Humphrey Shi. Escaping the big data paradigm with compact transformers. *arXiv preprint arXiv:2104.05704*, 2021.

[7] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are RNNs: Fast autoregressive transformers with linear attention. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 5156–5165. PMLR, 13–18 Jul 2020.

[8] Chankyu Lee, Syed Shakib Sarwar, Priyadarshini Panda, Gopalakrishnan Srinivasan, and Kaushik Roy. Enabling spike-based backpropagation for training deep neural network architectures. *Frontiers in neuroscience*, page 119, 2020.

[9] William B Levy and Victoria G. Calvert. Computation in the human cerebral cortex uses less than 0.2 watts yet this great expense is optimal when considering communication costs. *bioRxiv*, 2020.

[10] Qingyan Meng, Mingqing Xiao, Shen Yan, Yisen Wang, Zhouchen Lin, and Zhi-Quan Luo. Training high-performance low-latency spiking neural networks by differentiation on spike representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12444–12453, 2022.

[11] Zhen Qin, Weixuan Sun, Hui Deng, Dongxu Li, Yunshen Wei, Baohong Lv, Junjie Yan, Lingpeng Kong, and Yiran Zhong. cosformer: Rethinking softmax in attention. In *International Conference on Learning Representations*, 2022.

[12] Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. Dynamicvit: Efficient vision transformers with dynamic token sparsification. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 13937–13949. Curran Associates, Inc., 2021.

[13] Jeong-geun Song. Ufo-vit: High performance linear vision transformer without softmax. *arXiv preprint arXiv:2109.14382*, 2021.

[14] Neil C Thompson, Kristjan Greenewald, Keeheon Lee, and Gabriel F Manso. The computational limits of deep learning. *arXiv preprint arXiv:2007.05558*, 2020.

[15] Yuchen Wang, Malu Zhang, Yi Chen, and Hong Qu. Signed neuron with memory: Towards simple, accurate and high-efficient ann-snn conversion. In *International Joint Conference on Artificial Intelligence*, 2022.

[16] Yujie Wu, Lei Deng, Guoqi Li, Jun Zhu, and Luping Shi. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in Neuroscience*, 12, 2018.

[17] Tete Xiao, Mannat Singh, Eric Mintun, Trevor Darrell, Piotr Dollar, and Ross Girshick. Early convolutions help transformers see better. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 30392–30400. Curran Associates, Inc., 2021.

[18] Jianwei Yang, Chunyuan Li, Pengchuan Zhang, Xiyang Dai, Bin Xiao, Lu Yuan, and Jianfeng Gao. Focal attention for long-range interactions in vision transformers. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, volume 34, pages 30008–30022. Curran Associates, Inc., 2021.