面向Transformer模型边缘端部署的常用激活函数高精度轻量级量化推理方法

杨赟辉,程 虎,魏敬和*,刘国柱,桑贤侦
(中国电子科技集团公司第五十八研究所,江苏无锡 214072)

摘 要: 基于Transformer的大语言模型(Large Language Models, LLM)和视觉Transformer(Vision Transformers, ViTs)分别在自然语言处理、机器视觉任务上实现了最为先进的性能. 但是ViTs和LLM的常用激活函数GELU(Gaussian Error Linear Unit)、Swish在Transformer全量化推理中存在精度不足、计算效率低的问题,限制了它们在资源受限的边缘端设备上的部署和应用. 本文提出了一种基于分段二次多项式拟合的激活函数高精度近似计算方法(Segmented Quadratic Polynomial Fitting,SQPF)及其量化推理过程,以实现端侧非线性激活函数的高性能部署. SQPF采用最小二乘法和粒子群优化方法求解非线性激活函数拟合优化问题,给出最优的二次多项式拟合系数和区间划分. 得到的二次多项式拟合采用动态精度定点对称量化方法进行纯整数推理,推理过程仅包含移位操作和乘加运算. 本文使用SQPF计算了GELU和Swish的二次多项式拟合Si-GELU和Si-Swish,并评估了量化推理精度. 实验结果表明,在标准数据集ImageNet上,Si-GELU引起的ViTs(ViT、DeiT和Swin)模型分类任务准确率衰减仅为0.09%,是其他同类方法的27.3%;在主流的大语言模型评测数据集MMLU上,Si-Swish引起的子类别精度衰减不超过0.77%,大类别精度衰减不超过0.23%. 极小的精度损失表明SQPF计算得到的最优分段二次多项式拟合可以直接替换Transformer模型中全精度浮点激活函数,不必进行参数微调或者重训练.

关键词: Transformer;全量化推理;GELU函数;Swish函数;分段二次多项式拟合

基金项目: 江苏省自然科学基金(No.K20211041,No.BK20211040,No.BE2021003-1,No.BE2023005-1);国家自然科学基金(No.62174150)

中图分类号: TP301 **文献标识码:** A **文章编号:** 0372-2112(2024)10-3301-11

电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20240435

High-Precision Lightweight Quantization Inference Method for Prevalent Activation Functions in Transformer Models in Edge Device Deployment

YANG Yun-hui, CHENG Hu, WEI Jing-he*, LIU Guo-zhu, SANG Xian-zhen
(*China Electronics Technology Group Corporation No.*58 *Research Institute*, *Wuxi*, *Jiangsu* 214072, *China*)

Abstract: Transformer-based models, such as large language models (LLM) and vision Transformers (ViTs), had achieved state-of-the-art performance in tasks across natural language processing and machine vision. However, the prevalent activation functions such as GELU (Gaussian Error Linear Unit) and Swish in ViTs and LLMs encountered challenges with insufficient precision and low computational efficiency during fully quantized inference, which constrained their deployment and application in resource-limited edge devices. This paper introduced a high-precision segmented quadratic polynomial fitting method (SQPF) and its corresponding quantized inference process, to achieve high-performance deployment of nonlinear activation functions on the edge side.The SQPF adopted the least squares method and particle swarm optimization to fetch the optimal coefficient and interval divisions for the quadratic polynomial fitting of activation functions. The obtained quadratic polynomials-were subjected to dynamic fixed-point symmetric quantization, enabling pure integer inference that solely required shift operations and multiply-accumulate computations. This paper calculated the quadratic polynomials of GELU and Swish to Si-GELU and Si-Swish, and evaluated their inference accuracy. The experimental results demonstrated that on ImageNet, the Si-GELU in-

收稿日期:2024-05-10;修回日期:2024-07-11;责任编辑:覃怀银
*通讯作者:魏敬和

duced a minimal accuracy reduction of only 0.09% in the classification tasks for ViTs (ViT, DeiT, and Swin), which is 27.3% of other methods. On large language model benchmark dataset MMLU, Si-Swish caused a negligible precision degradation, with subcategory precision degradation not exceeding 0.77% and major category precision degradation not exceeding 0.23%. The minimal loss in precision indicated that the optimal quadratic polynomials derived from SQPF were a direct substitute for the full-precision floating-point activation functions in Transformer models, negating parameter fine-tuning or retraining.

Key words: Transformers; fully quantized inference; GELU; Swish; segmented quadratic polynomial fitting

Foundation Item(s): Natural Science Foundation of Jiangsu Province (No. K20211041, No. BK20211040, No. BE2021003-1, No.BE2023005-1); National Natural Science Foundation of China (No.62174150)

1 引言

基于Transformer的算法模型，例如DeiT[1]、BERT[2]、Swin[3]，LlaMa2[4]等，在机器视觉、自然语言处理等任务上实现了最为先进的效果[5]，但是Transformer推理需要占用大量存储和计算资源，在资源受限的边缘端设备[6,7]上难以实现高效部署和推理[8~11]. 量化是解决这一问题的有效手段[12~14]. 量化以较低精度表示权重和激活值，显著降低了神经网络的模型尺寸，有效减少了内存占用. 量化后的模型使用低精度整数乘加运算，相较于浮点运算，可以显著提升推理速度，降低延迟和硬件功耗[15,16]. 此外在芯片或硬件电路设计过程中，相同运算复杂度下，使用整数运算逻辑替代浮点运算逻辑，可以有效减少芯片面积和硬件逻辑资源[17]. 因此探索Transformer算法模型高精度全量化推理是实现Transformer算法模型边缘端高效部署和应用的关键. 当前Transformer算法模型的量化方法大多属于模拟量化和部分量化，模型推理过程中没有完全取消浮点运算，非线性操作Softmax、Layer Normalization以及激活函数GELU(Gaussian Error Linear Unit)、Swish等多是反量化后进行浮点运算[18~21]，以保证推理精度.

非线性激活函数的高精度纯整数计算是实现Transformer全量化推理的重要环节. 非线性激活函数对精度要求严格，低精度处理会造成Transformer算法模型推理结果产生显著的精度衰减[17]. 因此近年来关于Transformer全量化推理架构中都研究了非线性激活函数纯整数计算方法. Li等人[18]的I-ViT算法中采用基于Sigmoid函数的GELU函数近似计算公式[22]，使用移位代替乘法提出了ShiftGELU方法，通过调用自然指数运算和除法运算模块实现GELU高精度纯整数运算. 但是Sigmoid近似公式本身损失了部分精度，而且依赖于运算复杂度较大的自然指数运算和除法运算模块. Howard等人[23]在MobilenetV3算法中提出h-GELU方法，使用h-Sigmoid来对Sigmoid进行纯整数近似，避免了自然指数、除法运算，运算复杂度显著下降. Kim等人[17]在I-BERT算法中使用轻量级的二次多项式拟合高斯误差函数(Gauss Error Function, ERF)，提出了GELU函数高精度量化方法i-GELU. h-GELU和i-GELU虽然运算复杂度低，但是精度不足，直接使用会导致Transformer算法推理结果的准确率有较大衰减[18]. 对于INT8以下的量化，可以采用查找表方法实现对非线性激活函数GELU、Swish的边缘端部署，同时需要对模型参数进行微调或者重训练[18]，但是对于LlaMa2为代表大语言模型(Large Language Models, LLM)微调或者重训练成本相对昂贵.

本文针对现有Transformer全量化推理方案中典型非线性激活函数GELU、Swish纯整数计算方法存在精度不足、计算效率低的问题，提出了一种基于分段二次多项式拟合的激活函数高精度近似计算方法(Segmented Quadratic Polynomial Fitting, SQPF). SQPF采用最小二乘法和粒子群优化方法求解非线性激活函数(Activation, Act)的拟合优化问题，给出最优的二次多项式拟合系数和区间划分，得到分段非线性激活函数二次多项式拟合Si-Act；采用动态精度定点对称量化方法实现Si-Act的纯整数推理，推理过程仅包含移位操作和乘加运算.

2 非线性激活函数轻量级高精度多项式近似

2.1 分段二次多项式近似方法

采用多项式拟合非线性激活函数可以兼顾精度和边缘端数字电路的推理效率. 随着阶数的提升，多项式对非线性函数的逼近效果趋于优化，能够更加精确地反映目标函数特性，但代价是计算复杂度和内存开销显著提升，尤其在处理中间结果时需要大位宽避免溢出问题，保证数值计算的准确性. 文献[17]采用二次多项式拟合非线性激活函数以平衡计算精度要求和存算资源开销，本文在此基础上提出了一种基于分段二次多项式拟合的激活函数高精度近似计算方法SQPF.

2.1.1 多变量分段二次拟合优化问题降维

假设非线性激活函数Act的主要变化发生在区间$[C_1, C_2]$内. SQPF采用2段二次多项式l_1和l_2分别拟合区间$[C_1, \mathrm{thd})$和$[\mathrm{thd}, C_2]$内的非线性激活函数Act. 构建Act及其拟合函数Si-Act的损失loss关于二次多项式l_1和l_2拟合系数a_1、a_2、a_3和区间划分节点thd的多变量

优化问题如下:

$$\min_{a_1,a_2,a_3,\mathrm{thd}} \mathrm{loss}=\frac{1}{2}\left\|\mathrm{Act}-\mathrm{Si\text{-}Act}\right\|_2^2$$

s.t.

$$\mathrm{Act}\left(x_i\right)=\mathrm{Si\text{-}Act}\left(x_i\right), x_i\in\left\{C_1,\mathrm{thd},C_2\right\} \tag{1}$$

$$\mathrm{Si\text{-}Act}(x):=\begin{cases} l_1=a_{11}\left(x+a_{21}\right)^2+a_{31}, & x\in\left[C_1,\mathrm{thd}\right) \\ l_2=a_{12}\left(x+a_{22}\right)^2+a_{32}, & x\in\left[\mathrm{thd},C_2\right) \end{cases}$$

式(1)中,Act和Si-Act在区间节点$x\in\{C_1,\mathrm{thd},C_2\}$上满足连续性条件;$C_1$和$C_2$分别是拟合区间的下界和上界.求解过程如下:

(a)在区间$[x_i,\ x_{i+1})$内,构造一元二次多项式如下:

$$L_i(x)=a_i+b_i\left(x-x_i\right)+c_i\left(x-x_i\right)^2 \tag{2}$$

式(2)中,a_i、b_i、c_i分别为二次多项式$L_i(x)$的系数;$x_i\in\{C_1,\mathrm{thd},\ C_2\}$. 代入插值条件$L_i\left(x_i\right)=y_i$,得到

$$a_i=y_i \tag{3}$$

式(3)中,$y_i=\mathrm{Act}\left(x_i\right)$.

(b)在内部节点处各段二次多项式连续,满足:

$$L_i\left(x_{i+1}\right)=y_{i+1}, a_i+b_i\left(x_{i+1}-x_i\right)+c_i\left(x_{i+1}-x_i\right)^2=y_{i+1} \tag{4}$$

为后续表述方便,令$h_i=x_{i+1}-x_i$,根据式(4)得到b_i关于c_i的表达式:

$$b_i=\frac{y_{i+1}-y_i}{h_i}-c_ih_i \tag{5}$$

(c)在各段区间内进行采样,采用最小二乘法来求解c_i的最优估计. 二次多项式$L_i(x)$在区间$[x_i,x_{i+1})$内进行采样$\left\{x_m|x_m\in\left[x_i,x_{i+1}\right),m=0,1,\cdots,M-1\right\}$,构造残差平方和$E_i$的表达式为

$$E_i=\sum_{m=0}^{M-1}\left[L_i\left(x_m\right)-y_m\right]^2 \tag{6}$$

式(6)中,$y_m=\mathrm{Act}\left(x_m\right)$. 令$h_m=x_m-x_i$,代入式(6)得到

$$E_i=\sum_{m=0}^{M-1}\left[y_i+\frac{y_{i+1}-y_i}{h_i}h_m+c_i\left(h_m^2-h_ih_m\right)-y_m\right]^2 \tag{7}$$

求解$\partial E_i/\partial c_i=0$,得到$c_i$在区间$[x_i,x_{i+1})$内的最小二乘法估计$\tilde{c}_i$为

$$\tilde{c}_i=\frac{-\sum_{m=0}^{M-1}\left[y_i+\frac{y_{i+1}-y_i}{h_i}h_m-y_m\right]\left(h_m^2-h_ih_m\right)}{\sum_{m=0}^{M-1}\left(h_m^2-h_ih_m\right)^2} \tag{8}$$

代入式(5),进而求解出b_i. 将式(2)转化为二次函数的标准形式便于后续量化推理:

$$L_i(x)=a_{1i}\left(x+a_{2i}\right)^2+a_{3i} \tag{9}$$

以上推导表明,给定节点thd可以确定一组l_1和l_2曲线及其损失loss. 式(1)的多变量优化问题转化为拟合损失函数loss关于节点thd的单变量优化问题.

2.1.2　粒子群方法求解单变量拟合优化问题

采用粒子群优化算法(Particle Swarm Optimization,PSO)求解拟合损失函数loss关于节点thd的单变量优化问题,给出最优的区间划分节点thd的过程如下:

(a)PSO初始化为一群随机粒子,粒子群的位置表示为

$$D=\left\{d_0,d_1,\cdots,d_{N-1}\right\},\ d_n\in\left(C_1,C_2\right) \tag{10}$$

其中,d为单个粒子位置,代表一个区间划分节点,第n个粒子位置d_n划分采样区间$[C_1,d_n)$和$[d_n,C_2)$;N为粒子数量. 随机粒子的飞行速度在范围$(-R,R)$随机初始化为

$$V=\left\{v_0,v_1,\cdots,v_{N-1}\right\},v_n\in\left(-R,R\right) \tag{11}$$

(b)分别在区间$[C_1,\ d_n)$和$[d_n,\ C_2)$内确定拟合曲线l_1和l_2,根据式(1)计算损失loss,记录"个体历史最优值"(p_n)、"全局最优值"(g)以及对应粒子位置d_{pn}和d_g.

(c)粒子群更新速度,并对越界的速度进行约束. 第n个粒子更新速度表达式为

$$v_i^{t+1}=\omega v_i^t+\varepsilon_1r_1\left(d_{\mathrm{pn}}^t-d_n^t\right)+\varepsilon_2r_2\left(d_{\mathrm{g}}-d_n^t\right) \tag{12}$$

式(12)中,ω为惯性因子;ε_1和ε_2为学习因子,r_1和r_2为随机值.

(d)粒子群更新位置,并对越界的位置进行约束:

$$d_n^{t+1}=d_n^t+v_n^{t+1} \tag{13}$$

(e)跳转到步骤(b)进行迭代,直到达到最大迭代次数,输出最优的区间划分节点thd,根据式(3)、式(5)、式(8)计算出l_1和l_2曲线的二次多项式系数.

2.2　GELU函数的分段二次多项式近似

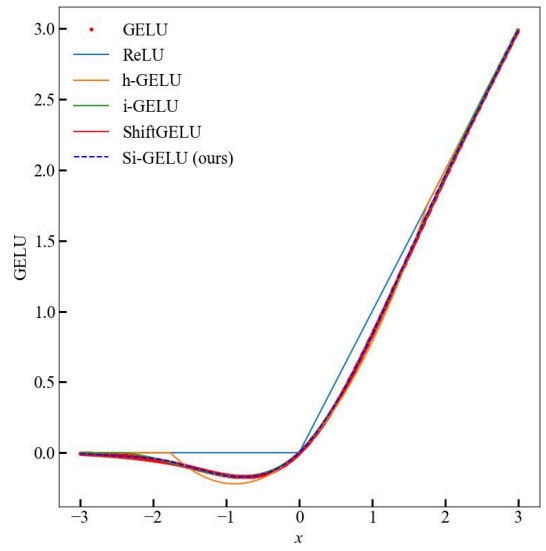
GELU函数是视觉Transformer算法模型(Vision Transformers,ViTs)中的非线性激活函数. 相比ReLU,GELU函数在0附近区间导数是连续的,减少了训练过程中出现的梯度消失问题,加速模型训练的收敛速度,提高模型在自然语言处理、计算机视觉等任务中的性能[22]. 如图1所示,GELU函数(图1红色点线)和ReLU函数(图1蓝色线)在正负绝对值较大区间上行为相似,但是在0附近区间上的差别较大. GELU(x)函数的理论计算公式[22]如下:

$$\mathrm{GELU}(x)=\frac{1}{2}x\left[1+\mathrm{ERF}\left(\frac{x}{\sqrt{2}}\right)\right] \tag{14}$$

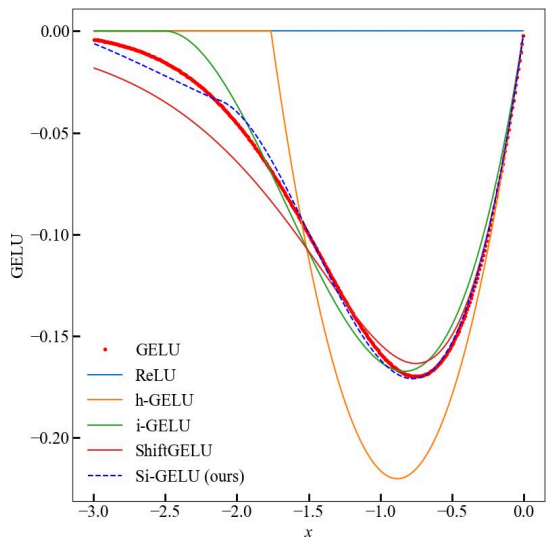
$$\mathrm{ERF}(x)=\frac{2}{\sqrt{\pi}}\int_0^x\mathrm{e}^{-t^2}\mathrm{d}t \tag{15}$$

式(15)中,ERF(·)为高斯误差函数. GELU函数的非线性部分由ERF函数构成,并且由于积分项的存在,直接计算ERF函数并不高效,目前学界普遍采用近似方法进行量化推理[17,18,21,22]. 由于近似值与准确值存在较大的差别,直接应用近似方法会导致Transformer模型的推理结果不理想,需要进行参数微调或者重训练[18].

SQPF采用分段二次多项式拟合的方式提升对ERF



(a) 全区间



(b) 局部区间

图1 本文方法与其他GELU函数拟合效果的比较

函数的近似精度，构建GELU函数高精度拟合Si-GELU计算公式如下：

$$\text{Si-GELU}(x) \coloneqq \frac{1}{2}x\left[1+\tilde{\text{ERF}}\left(\frac{|x|}{\sqrt{2}}\right)\right] \tag{16}$$
$$\tilde{\text{ERF}}(x)=\text{sign}(x)L(|x|)$$

式(16)中，$L(x)$为二次多项式；sign为符号．由于直接在整个实数域内对ERF函数进行近似会导致较差的近似效果[17]，观察到ERF函数的主要变化发生在区间$[-3.0,3.0]$，且ERF函数为奇函数．分段二次多项式$L(x)$对ERF函数在区间$[0.0,3.0]$内进行近似，超过这一个区域函数值可以认为是1.0．各分段的二次多项式拟合函数在各个区间节点上满足插值条件，参考式(1)构建拟合损失函数loss关于二次多项式拟合系数a_1、a_2、a_3和区间划分节点thd的多变量优化问题如下：

$$\min_{a_1,a_2,a_3,\text{thd}} \text{loss}=\frac{1}{2}\left\|\text{GELU}\left(\sqrt{2}\,x\right)-\text{Si-GELU}\left(\sqrt{2}\,x\right)\right\|_2^2$$
s.t.
$$L(x)=\begin{cases} l_1=a_{11}\left(x+a_{21}\right)^2+a_{31},\ x\in[0,\text{thd}) \\ l_2=a_{12}\left(x+a_{22}\right)^2+a_{32},\ x\in[\text{thd},3.0] \end{cases}$$
$$L_1(0)=\text{ERF}(0),\ L_2(\text{thd})=\text{ERF}(\text{thd})$$
$$L_1(\text{thd})=\text{ERF}(\text{thd}),\ L_2(3.0)=\text{ERF}(3.0) \tag{17}$$

式(17)中，l_1和l_2分别是两个子区间内对ERF函数的二次多项式拟合．在区间节点$x\in\{0,\ \text{thd},\ 3.0\}$上$l_1$和$l_2$满足插值条件．采用2.1节步骤求解式(17)，得到各段最优的二次多项式拟合系数a_1、a_2、a_3和最优的区间划分节点thd，实现对ERF函数的高精度拟合，$L(x)$对ERF函数的拟合效果如图2所示．根据式(16)，得到Si-GELU对GELU函数的拟合效果如图1蓝色虚线所示．

2.3 Swish函数的分段二次多项式近似

Swish激活函数具有平滑非饱和性质，能够提高模型的泛化能力和表达能力，在PaLM[24]、LlaMa[4]为代表的Transformer大模型中被用作全连接层的激活函数，增强模型的表达能力和性能．Swish函数表达式如下：

$$\text{Swish}_\beta(x)=x\sigma(\beta x) \tag{18}$$
$$\sigma(x)=\frac{1}{1+\text{e}^{-x}} \tag{19}$$

其中，β是常数；$\sigma(x)$为Sigmoid函数，构成Swish函数的非线性部分．由于Sigmoid函数是一个奇函数，参考GELU，SQPF的二次多项式$L(x)$仅Sigmoid函数拟合区间$x>0$的部分．

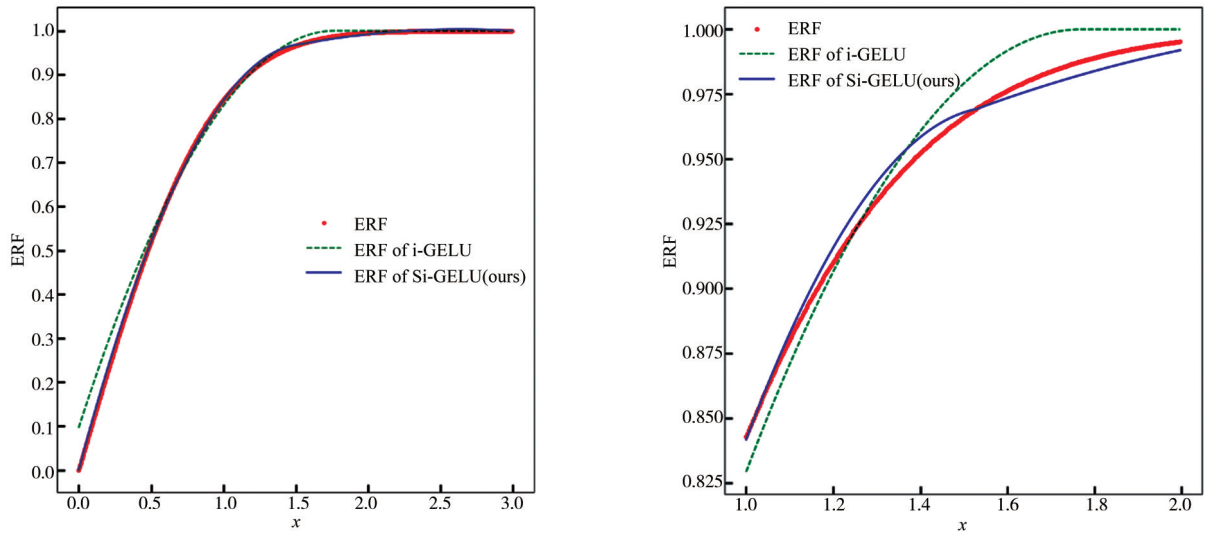
$$\tilde{\sigma}(x)=\left[0.5+\text{sign}(x)\left(L(|x|)-0.5\right)\right]$$
$$L(x)=a_1\left(x+a_2\right)^2+a_3\ \ x\in[0,\text{Const}) \tag{20}$$

Sigmoid函数的主要变化发生在区间$[-8.0,8.0]$，对于超过这一个区域函数值可以认为是1.0．根据$\tilde{\sigma}(x)$计算Swish函数的高精度拟合Si-Swish如下：

$$\text{Si-Swish}(x)\coloneqq x\tilde{\sigma}(x) \tag{21}$$

参考式(1)可以构建优化问题如下：

$$\min_{a_1,a_2,a_3,\text{thd}} \text{loss}=\frac{1}{2}\left\|\text{Swish}(x)-\text{Si-Swish}(x)\right\|_2^2$$
s.t.
$$L(x)=\begin{cases} l_1=a_{11}\left(x+a_{21}\right)^2+a_{31},\ x\in[0,\text{thd}) \\ l_2=a_{12}\left(x+a_{22}\right)^2+a_{32},\ x\in[\text{thd},8.0] \end{cases} \tag{22}$$
$$L_1(0)=\sigma(0),\ L_2(\text{thd})=\sigma(\text{thd})$$
$$L_1(\text{thd})=\sigma(\text{thd}),\ L_2(8.0)=\sigma(8.0)$$



(a) 全区间拟合效果对比　　(b) 局部区间拟合效果对比

图2　Si-GELU和i-GELU[17]对ERF函数的拟合效果对比

求解得到最优的二次多项式拟合系数a_1、a_2、a_3和最优的区间划分节点thd. 不同β的Si-Swish对Swish函数拟合效果如图3所示.

3　分段二次多项式近似的量化推理

模型量化将浮点参数转化为低精度整数,以一种硬件友好的方式压缩模型尺寸[25~28]. 文献[29,30]等都提出了面向Transformer结构的量化方法,但是非线性操作Softmax、Layer Normalization、GELU、Swish会在推理过程反量化回浮点以保证推理精度[18],无法充分利用低精度运算单元,限制了模型的推理加速. 完全消除反量化、浮点运算等操作,采用低精度纯整数运算是边缘端高效部署Transformer模型的关键. 本文采用动态精度定点对称量化方法对2类二次多项式拟合Si-Act(Si-GELU和Si-Swish)进行纯整数推理,量化推理过程仅包含移位操作和乘加运算,对边缘端数字电路部署和应用友好.

3.1　Si-GELU的动态精度量化推理

代入定点量化公式$\boldsymbol{X}=\boldsymbol{Q}2^{-k}$到式(16)得到

$$\boldsymbol{Q}_{\text{ou}}2^{-k_{\text{ou}}}=\frac{1}{2}\boldsymbol{Q}_{\text{in}}2^{-k_{\text{in}}}\left[1+\text{sign}(\boldsymbol{X})L\left(\left|\boldsymbol{Q}_{\text{in}}\right|2^{-k_{\text{in}}-0.5}\right)\right] \tag{23}$$

式中,$\boldsymbol{X}$是浮点向量,sign(·)为符号;$\boldsymbol{Q}$为量化后的定点整数,k为量化位宽,角标in和ou分别代表输入和输出. 其中,

$$L\left(\left|\boldsymbol{Q}_{\text{in}}\right|2^{-k_{\text{in}}-0.5}\right)=2^{-2k_{\text{in}}-1}\left[a_1\left(\left|\boldsymbol{Q}_{\text{in}}\right|+a_2 2^{k_{\text{in}}+0.5}\right)^2+a_3 2^{2k_{\text{in}}+1}\right] \tag{24}$$

对式(24)化简后得到

$$L\left(\left|\boldsymbol{Q}_{\text{in}}\right|2^{-k_{\text{in}}-0.5}\right)=2^{-2k_{\text{in}}-1}\boldsymbol{Q}_{\text{ERF}} \tag{25}$$

$$\boldsymbol{Q}_{\text{ERF}}=a_1\left(\left|\boldsymbol{Q}_{\text{in}}\right|+a_2 2^{k_{\text{in}}+0.5}\right)^2+a_3 2^{2k_{\text{in}}+1} \tag{26}$$

根据输入、输出量化的关系,

$$\boldsymbol{Q}_{\text{ou}}2^{-k_{\text{ou}}}=\boldsymbol{Q}_{\text{in}}\left[2^{2k_{\text{in}}+1}+\text{sign}(\boldsymbol{X})\boldsymbol{Q}_{\text{ERF}}\right]2^{-3k_{\text{in}}-2} \tag{27}$$

得到量化后的输出结果及其量化位宽为

$$\boldsymbol{Q}_{\text{ou}}=\boldsymbol{Q}_{\text{in}}\left(2^{2k_{\text{in}}+1}+\text{sign}(\boldsymbol{X})\boldsymbol{Q}_{\text{ERF}}\right) \tag{28}$$

$$k_{\text{ou}}=3k_{\text{in}}+2 \tag{29}$$

3.2　Si-Swish的动态精度量化推理

代入定点量化公式$\boldsymbol{X}=\boldsymbol{Q}2^{-k}$到式(21)得到

$$\boldsymbol{Q}_{\text{ou}}2^{-k_{\text{ou}}}=\boldsymbol{Q}_{\text{in}}2^{-k_{\text{in}}}\left\{0.5+\text{sign}(\boldsymbol{X})\left[L\left(\left|\boldsymbol{Q}_{\text{in}}\right|2^{-k_{\text{in}}}\right)-0.5\right]\right\} \tag{30}$$

对分段二次多项式L进行展开,

$$L\left(\left|\boldsymbol{Q}_{\text{in}}\right|2^{-k_{\text{in}}}\right)=a_1\left(\left|\boldsymbol{Q}_{\text{in}}\right|2^{-k_{\text{in}}}+a_2\right)^2+a_3=2^{-2k_{\text{in}}}\boldsymbol{Q}_{\sigma} \tag{31}$$

$$\boldsymbol{Q}_{\sigma}=a_1\left(\left|\boldsymbol{Q}_{\text{in}}\right|+a_2 2^{k_{\text{in}}}\right)^2+a_3 2^{2k_{\text{in}}} \tag{32}$$

简化后得到

$$\boldsymbol{Q}_{\text{ou}}=\boldsymbol{Q}_{\text{in}}\left\{\left[1-\text{sign}(\boldsymbol{X})\right]2^{2k_{\text{in}}-1}+\text{sign}(\boldsymbol{X})\boldsymbol{Q}_{\sigma}\right\} \tag{33}$$

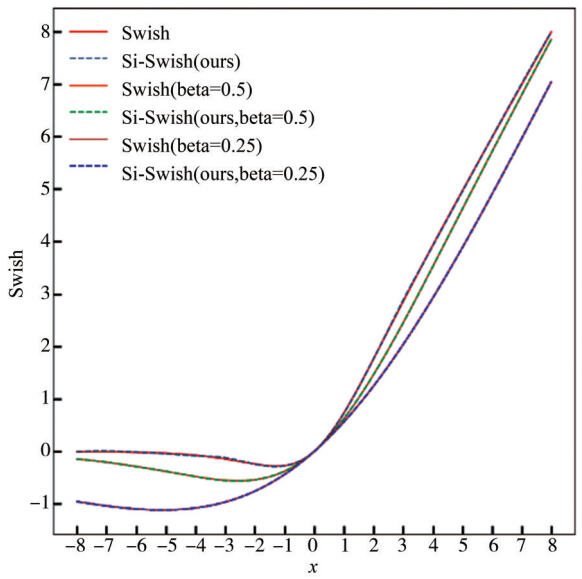
$$k_{\text{ou}}=3k_{\text{in}} \tag{34}$$

4　实验结果与分析

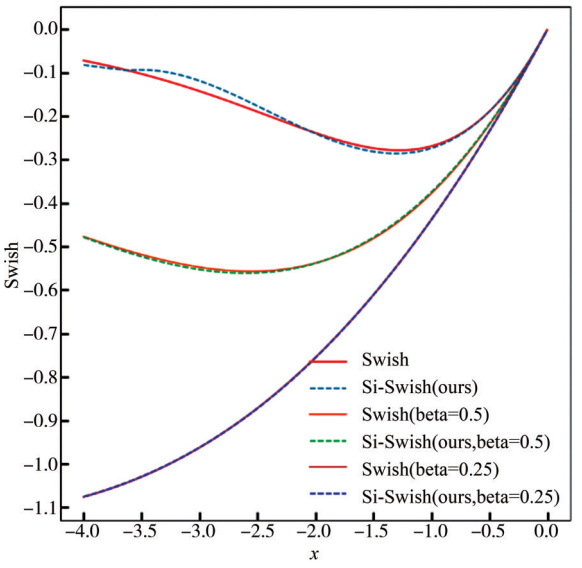
4.1　实验环境

本章以Si-GELU和Si-Swish为例,评估SQPF方法得到的Si-Act分别在大规模分类任务和大模型多任务语言理解评估上的推理精度以展示SQPF方法的先进性. 具体实现方式如下:

(1)使用Si-GELU、ShiftGELU和i-GELU分别替换ViTs模型中原有的GELU全精度浮点运算单元,并在拟合函数与激活层输入、输出之间分别增加动态精度定



(*a*) 全区间拟合效果对比



(*b*) 局部区间拟合效果对比

图3 Si-Swish对不同β的Swish函数的拟合效果对比

点量化、反量化模块，如图4所示进行端到端测试. ViTs模型采用主流的ViT[31]、DeiT[1]和Swin[3]及其对应的公开模型权重进行测试；测试数据集合采用ImageNet (ILSVRC-2012)[32].

(2)使用Si-Swish替换LlaMa2模型中全精度浮点的Swish函数，并在Si-Swish与激活层输入、输出之间分别增加动态精度定点量化、反量化模块，按图4所示流程进行端到端测试. 权重使用公开的LlaMa2-7B权重，测试数据集合采用主流LLM评测数据集MMLU(Massive Multitask Language Understanding)[33].

算法1 Si-GELU量化推理算法

输入:量化输入$\boldsymbol{Q}_{\text{in}}$;量化位宽$k_{\text{in}}$

输出:量化输出$\boldsymbol{Q}_{\text{ou}}$;量化位宽$k_{\text{ou}}$

(*a*)记录符号sign($\boldsymbol{Q}_{\text{in}}$),计算输入绝对值$|\boldsymbol{Q}_{\text{in}}|$.

(*b*)计算$\boldsymbol{Q}_{\text{ERF}}$,

$C_{a2} \leftarrow \left(\sqrt{2}\,a_2\right) << \left(k_{\text{in}}\right)$;/*$\sqrt{2}\,a_2$是预计算并存储*/

$C_{a3} \leftarrow \left(a_3\right) << \left(2k_{\text{in}}+1\right)$;

$\boldsymbol{Q}_{\text{ERF}} \leftarrow a_1\left(\left|\boldsymbol{Q}_{\text{in}}\right| + C_{a2}\right)^2 + C_{a3}$.

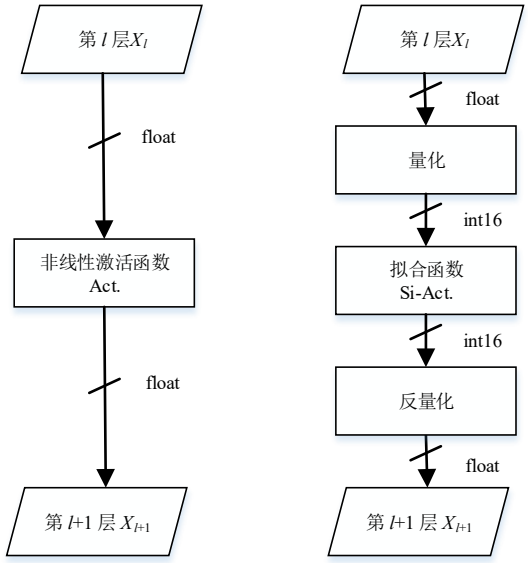
(*c*)计算$\boldsymbol{Q}_{\text{ou}}$和$k_{\text{ou}}$,

$C_{c1} \leftarrow 1 << \left(2k_{\text{in}}+1\right)$;

$\boldsymbol{Q}_{\text{ou}} \leftarrow \boldsymbol{Q}_{\text{in}}\left[C_{c1} + \text{sign}\left(\boldsymbol{Q}_{\text{in}}\right)\boldsymbol{Q}_{\text{ERF}}\right]$;

$k_{\text{ou}} \leftarrow 3k_{\text{in}}+2$.

(*d*)输出$\boldsymbol{Q}_{\text{ou}}$和$k_{\text{ou}}$.



(*a*) Act端到端测试流程 (*b*) Si-Act端到端测试流程

图4 非线性激活函数Act及其拟合函数Si-Act端到端测试流程图

所有测试在PyTorch深度学习框架下执行，Si-GELU、ShiftGELU、i-GELU、Si-Swish的量化精度均采用16 bit，二次多项式拟合系数a_1、a_2、a_3和区间划分节点thd均采用整型数据格式存储. 实验环境配置如下：Python3.8，PyTorch1.12.1，GTX 2070，Inter Core i9-9900KF 3.60 GHz CPU以及16 GB内存.

4.2 拟合效果评估

粒子群优化算法中粒子数量设定为30个，最大迭代次数设为200次，速度变化范围设为(−0.2, 0.2)，惯性因子ω设为1.0，学习因子c_1和c_2均设置为2.0.

4.2.1 Si-GELU对GELU函数的拟合效果评估

粒子群算法得到Si-GELU中ERF函数最优的区间

划分节点thd为1.533,计算对应的l_1和l_2曲线如下:

$$\begin{cases} l_1=-0.3927(x-1.572)^2+0.9699, & x\in(0,1.533) \\ l_2=-0.02749(x-2.644)^2+1.003, & x\in[1.533,3.0] \end{cases} \tag{35}$$

$L(x)$对ERF函数的拟合效果如图2蓝色实线所示,拟合效果优于i-GELU方法得到的二次多项式. 图1比较了现有其他方法对GELU函数的拟合效果,定性分析可得Si-GELU拟合效果最优,与GELU原函数曲线实现紧密贴合;定量计算Si-GELU与GELU函数在区间$[-3\sqrt{2},3\sqrt{2}]$内的误差结果如表1所示,Si-GELU的欧几里得距离(L2)与切比雪夫距离(L∞)最小,相比现有方法下降60%.

表1　各GELU函数量化推理方法拟合效果比较

Method	L2 dist	L∞ dist
ShiftGELU	0.000 40	0.020 33
h-GELU	0.001 12	0.068 70
i-GELU	0.000 30	0.018 15
Si-GELU (ours)	**0.000 12**	**0.006 63**

各GELU函数量化推理方法在区间$[-3\sqrt{2},3\sqrt{2}]$内的误差变化曲线图5所示.

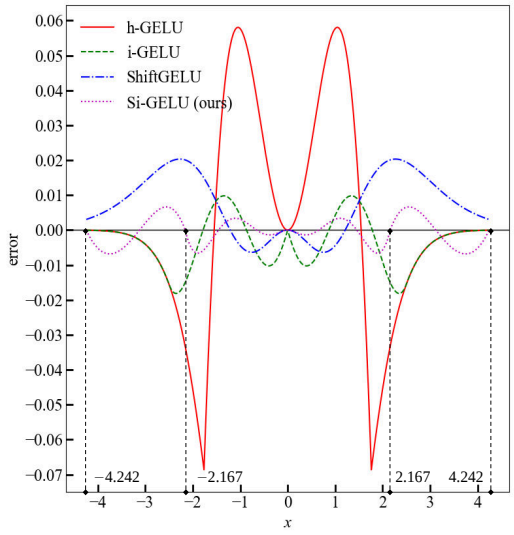


图5　各GELU函数量化推理方法误差变化曲线

Si-GELU在区间内误差波动最小. 由于Si-GELU在区间节点处满足插值条件,故在{0, 2.167, 4.242}处误差为0. ERF函数及其$L(x)$的导数如下:

$$\mathrm{ERF}'(x)=\frac{2}{\sqrt{\pi}}\mathrm{e}^{-x^2} \tag{36}$$

$$L'(x)=\begin{cases} -0.7854x+1.235, & x\in(0,1.533) \\ -0.05498x+0.1454, & x\in[1.533,3.0] \end{cases} \tag{37}$$

式(36)和式(37)在区间(0, 1.533)内存在3个交点,在区间[1.533, 3.0]内存在2个交点,在交点处误差达到极大值. 图5所示Si-GELU的误差曲线在区间(0, 2.167)内存在3个极值点,在区间[2.167, 4.242]内存在2个极值点,并且误差最大值L∞在其中一个极值点处取到.

4.2.2　Si-Swish对Swish函数的拟合效果评估

粒子群算法得到Si-Swish最优的区间划分节点thd为3.612,计算β=1对应的l_1和l_2曲线如下:

$$\begin{cases} l_1=-0.03652(x-3.602)^2+0.9737, & x\in(0,3.612) \\ l_2=-0.002239(x-7.126)^2+1.001, & x\in[3.612,8.0] \end{cases} \tag{38}$$

图3展示了不同β的Si-Swish对Swish函数在区间[−8, 8]内的拟合效果,拟合曲线与原曲线基本重合. 定量计算不同β的Si-Swish与Swish函数在区间[−8, 8]内的L2和L∞距离如表2所示. 图6给出了不同β的Si-Swish在区间[−8, 8]内的误差变化曲线. 由于Si-Swish(β=1)在区间节点处满足插值条件,故在{0, 3.612, 8}处误差为0. 与Si-GELU类似Si-Swish(β=1)的误差曲线也具有5个极值点,误差最大值L∞在其中一个极值点处取到.

表2　不同β的Si-Swish拟合效果评估

β	L2 dist	L∞ dist
1	0.000 25	0.024
0.5	0.000 063	0.005 3
0.25	0.000 009 2	0.000 84

4.3　对Transformer模型推理性能影响评估

4.3.1　Si-GELU对ViTs推理性能影响评估

如图4所示使用Si-GELU、ShiftGELU和i-GELU分别代替浮点GELU函数完成ViTs模型的前向推理,并在ImageNet上评估分类准确率. 表3详细展示了Si-GELU、ShiftGELU和i-GELU对ViT、DeiT和Swin等模型分类准确率的影响. 在相同量化精度下,Si-GELU引起的ViTs模型分类准确率衰减最小,和浮点结果相比准确率衰减基本维持在0.09%以下.

为进一步评估计算开销和推理性能,本条目采用Xilinx Vivado HLS 2019.2实现Si-GELU、ShiftGELU和i-GELU推理方法的IP设计和封装,主芯片采用Xilinx ZynqUltraScale+ MPSoCs系列的芯片,型号为XCZU7EV-2FFVC1156I,工作时钟频率设为100 MHz.

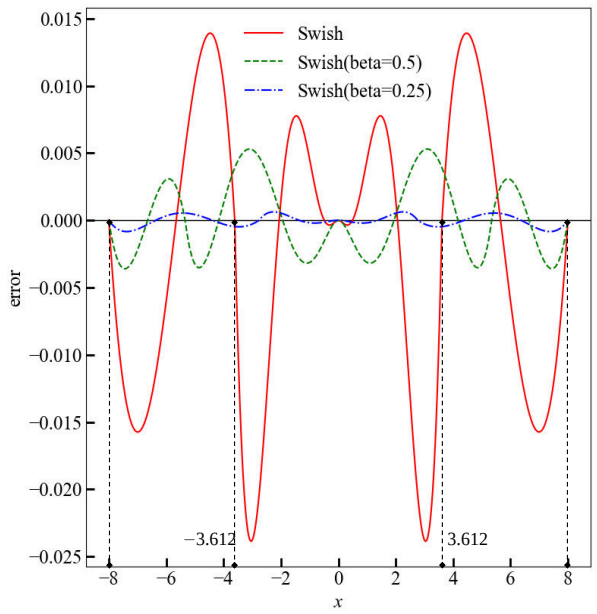


图6 不同β的Si-Swish误差变化曲线

表4给出了Si-GELU、ShiftGELU和i-GELU在Zynq FPGA上的推理延迟、总推理时间、资源占用以及功耗情况. ShiftGELU方法为避免计算自然指数EXP发生溢出,需要遍历所有输入数据以提取输入最大值,处理延迟与处理数据规模DS成正比. Si-GELU和i-GELU采用二次多项式近似方式,计算过程简单高效,推理延迟、资源占用以及功耗均低于ShiftGELU. 结合表3,Si-GELU和i-GELU计算开销和推理速度接近,但是Si-GELU的准确率衰减程度仅为i-GELU方法的27.3%.

4.3.2 Si-Swish对LlaMa2推理性能影响评估

如图4所示使用Si-Swish代替浮点Swish函数完成LlaMa2模型的前向推理,并在MMLU上评估推理性能. 表5详细展示了Si-Swish应用于大型语言模型LlaMa2时对评测数据集MMLU上不同科目推理性能的影响. Si-Swish在8个子类别中引起推理性能衰减,最大性能衰减程度不超过0.77%;在另外7个子类别中,Si-Swish提升了模型的推理性能,最高性能提升为1.38%. 在4个大类别中,Si-Swish引起2个大类别的推理性能衰减,衰减程度为0.23%;其余大类别的推理性能保持不变或提升,提升程度最高0.37%.

表3 各GELU函数量化推理方法对标准Transformer模型的分类准确率影响对比 单位:%

Model	Baseline	ShiftGELU		i-GELU		Si-GELU (ours)	
	Top-1 Acc.	Top-1 Acc.	Diff.	Top-1 Acc.	Diff.	Top-1 Acc.	Diff.
ViT-B	84.53	82.88	−1.65	84.20	−0.33	84.44	**−0.09**
ViT-L	85.84	84.87	−0.97	85.65	−0.19	85.80	**−0.04**
DeiT-T	72.14	70.99	−1.15	72.01	−0.13	72.11	**−0.03**
DeiT-S	79.83	79.29	−0.54	79.75	−0.08	79.83	**0.00**
DeiT-B	81.79	80.56	−1.23	81.57	−0.22	81.79	**0.00**
Swin-T	81.37	81.21	−0.16	81.30	−0.07	81.37	**0.00**
Swin-S	83.21	83.08	−0.13	82.98	−0.23	83.19	**−0.02**
Swin-B	83.60	83.56	−0.04	83.40	−0.20	83.57	**−0.03**

表4 各GELU函数量化推理方法在FPGA端计算开销情况

Methods	Latency/clk	Total time/clk	Logical resources			Power/W
			DSP	FF	LUT	
ShiftGELU	DS+46	2DS+46	8	2032	1723	28.54
i-GELU	15	DS+15	5	476	406	7.46
Si-GELU (ours)	15	DS+15	5	476	407	7.42

注:表中DS为处理数据的数量.

Si-GELU和Si-Swish在标准数据集上的测试结果表明,SQPF得到的Si-Act对Transformer模型推理精度的影响极小,可以直接原位替换全精度浮点模型中的非线性激活函数,不必进行参数微调或者重训练[18]. 此外,动态精度定点对称量化方式使得Si-Act量化推理过程仅包含移位操作和乘加运算,在FPGA、ASIC等数字电路上部署友好.

表5　Si-Swish对LlaMa2-7B在MMLU数据集上推理性能的影响　　单位:%

Subjects		Swish Acc.	Si-Swish Acc.	Diff.
subcategories	math	26.41	27.26	0.85
	health	42.20	42.68	0.48
	physics	33.59	33.91	0.32
	business	54.00	54.23	0.23
	biology	45.81	45.37	–0.44
	chemistry	32.01	32.01	0.00
	computer science	41.02	41.02	0.00
	economics	37.60	37.47	–0.13
	engineering	40.00	41.38	1.38
	philosophy	35.14	35.74	0.60
	other	48.07	47.30	–0.77
	history	53.98	53.87	–0.11
	geography	44.95	44.44	–0.51
	politics	51.70	51.39	–0.31
	psychology	49.27	49.18	–0.09
	culture	60.24	59.64	–0.60
	law	36.19	36.25	0.06
categories	STEM	34.06	34.43	0.37
	humanities	39.26	39.51	0.25
	social sciences	47.87	47.64	–0.23
	other (business, health, misc.)	45.90	45.89	–0.01

5　结论

本文针对现有Transformer全量化推理方案中激活函数纯整数计算方法存在精度不足、计算效率低的问题,提出了一种基于分段二次多项式拟合的激活函数高精度近似计算方法SQPF及其量化推理过程. 通过评估最优分段二次多项式拟合Si-GELU和Si-Swish在ViTs和LlaMa2模型上的推理精度损失,表明SQPF及其量化推理方法能够有效降低激活函数纯整数运算带来的Transformer模型推理精度衰减,显著提升Transformer全量化推理性能,为Transformer模型在FPGA、ASIC等边缘端数字电路上的激活函数部署问题提供了高效的解决方案.

参考文献

[1] TOUVRON H, CORD M, DOUZE M, et al. Training data-efficient image Transformers & distillation through attention[C]//International Conference on Machine Learning. New York: PMLR, 2021: 10347-10357.

[2] DEVLIN J, CHANG M W, LEE K, et al. BERT: Pre-training of deep bidirectional Transformers for language understanding[C]//Proceedings of NAACL-HLT. Minneapolis: Association for Computational Linguistics, 2019: 4171-4186.

[3] LIU Z, LIN Y T, CAO Y, et al. Swin Transformer: Hierarchical vision Transformer using shifted windows[C]//2021 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway: IEEE Press, 2021: 10012-10022.

[4] TOUVRON H, MARTIN L, STONE K R, et al. Llama 2: Open foundation and fine-tuned chat models[EB/OL]. (2023-07-19)[2024-05-10]. https://arxiv.org/abs/2307.09288.

[5] 刘兵, 李穗, 刘明明, 等. 基于全局与序列混合变分Transformer的多样化图像描述生成方法[J]. 电子学报, 2024, 52(4): 1305-1314.
LIU B, LI S, LIU M M, et al. Diverse image captioning based on hybrid global and sequential variational Transformer[J]. Acta Electronica Sinica, 2024, 52(4): 1305-1314. (in Chinese)

[6] 赵玥, 肖梦燕, 罗军, 等. 人工智能芯片及测评体系分析[J]. 电子与封装, 2023, 23(5): 31-37.
ZHAO Y, XIAO M Y, LUO J, et al. Analysisof artificial intelligence chip and evaluation system[J]. Electronics & Packaging, 2023, 23(5): 31-37. (in Chinese)

[7] 田文超, 谢昊伦, 陈源明, 等. 人工智能芯片先进封装技术[J]. 电子与封装, 2024, 24(1): 21-33.
TIAN W C, XIE H L, CHEN Y M, et al. Advanced packaging technology for artificial intelligence chips[J]. Electronics & Packaging, 2024, 24(1): 21-33. (in Chinese)

[8] LI Z K, MA L P, CHEN M J, et al. Patch similarity aware data-free quantization for vision Transformers[M]//Lecture Notes in Computer Science. Cham: Springer Nature Switzerland, 2022: 154-170.

[9] HOU Z J, KUNG S Y. Multi-dimensional vision Transformer compression via dependency guided Gaussian process search[C]//2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW). Piscataway: IEEE Press, 2022: 3669-3678.

[10] HAO Z, GUO J, JIA D, et al. Learning efficient vision Transformers via fine-grained manifold distillation[J]. Advances in Neural Information Processing Systems, 2022, 35: 9164-9175.

[11] TANG Y H, HAN K, WANG Y H, et al. Patch slimming for efficient vision Transformers[C]//P2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway: IEEE Press, 2022: 12165-12174.

[12] DONG Z, YAO Z W, GHOLAMI A, et al. HAWQ: Hessian aware quantization of neural networks with mixed-precision[C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway: IEEE Press, 2019: 293-302.

[13] JACOB B, KLIGYS S, CHEN B, et al. Quantization and training of neural networks for efficient integer-arithmetic-only inference[C]//2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE Press, 2018: 2704-2713.

[14] WU B, WANG Y, ZHANG P, et al. Mixed precision quantization of ConvNetsvia differentiable neural architecture search[EB/OL]. (2018-11-30) [2024-05-10]. https://arxiv.org/abs/1812.00090.

[15] WU H, JUDD P, ZHANG X J, et al. Integer quantization for deep learning inference: Principles and empirical evaluation[EB/OL]. (2020-04-20) [2024-05-10]. https://arxiv.org/abs/2004.09602.

[16] YAO Z, DONG Z, ZHENG Z, et al. HAWQ-V3: Dyadic neural network quantization[C]//International Conference on Machine Learning. New York: PMLR, 2021: 11875-11886.

[17] KIM S, GHOLAMI A, YAO Z, et al. I-BERT: Integer-only BERT quantization[C]//International Conference on Machine Learning. New York: PMLR, 2021: 5506-5518.

[18] LI Z K, GU Q Y. I-ViT: Integer-only quantization for efficient vision Transformer inference[C]//2023 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway: IEEE Press, 2023: 17065-17075.

[19] BHANDARE A, SRIPATHI V, KARKADA D, et al. Efficient 8-bit quantization of Transformer neural machine language translation model[EB/OL]. (2019-06-07) [2024-05-10]. https://arxiv.org/abs/1906.00532.

[20] SHEN S, DONG Z, YE J Y, et al. Q-BERT: Hessian based ultra low precision quantization of BERT[J]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(5): 8815-8821.

[21] ZAFRIR O, BOUDOUKH G, IZSAK P, et al. Q8BERT: Quantized 8bit BERT[C]//2019 Fifth Workshop on Energy Efficient Machine Learning and Cognitive Computing-NeurIPS Edition (EMC2-NIPS). Piscataway: IEEE Press, 2019: 36-39.

[22] HENDRYCKS D, GIMPEL K. Gaussian error linear units (GELUs)[EB/OL]. (2023-06-06)[2024-05-10]. https://arxiv.org/abs/1606.08415.

[23] HOWARD A, SANDLER M, CHEN B, et al. Searching for MobileNetV3[C]//2019 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway: IEEE Press, 2019: 1314-1324.

[24] CHOWDHERY A, NARANG S, DEVLIN J, et al. Palm: Scaling language modeling with pathways[J]. Journal of Machine Learning Research, 2023, 24(240): 1-113.

[25] KRISHNAMOORTHI R. Quantizing deep convolutional networks for efficient inference: A whitepaper[EB/OL]. (2019-06-21)[2024-05-10]. https://arxiv.org/abs/1806.08342.

[26] GHOLAMI A, KIM S, DONG Z, et al. A survey of quantization methods for efficient neural network inference [M]//Low-Power Computer Vision. Boca Raton: Chapman and Hall/CRC, 2022: 291-326.

[27] LI Z K, MA L P, LONG X L, et al. Dual-discriminator adversarial framework for data-free quantization[J]. Neurocomputing, 2022, 511: 67-77.

[28] ZHOU A, YAO A, GUO Y, et al. Incremental network quantization: Towards lossless CNNs with low-precision weights[EB/OL]. (2017-08-25)[2024-05-10]. https://arxiv.org/abs/1702.03044.

[29] LIN Y, ZHANG T, SUN P, et al. FQ-ViT: Post-training quantization for fully quantized vision Transformer[EB/OL]. (2023-02-17)[2024-05-10]. https://arxiv.org/abs/2111.13824.

[30] LI Z K, XIAO J R, YANG L W, et al. RepQ-ViT: Scale

reparameterization for post-training quantization of vision Transformers[C]//2023 IEEE/CVF International Conference on Computer Vision (ICCV). Piscataway: IEEE Press, 2023: 17227-17236.

[31] DOSOVITSKIY A, BEYER L, KOLESNIKOV A, et al. An image is worth 16x16 words: Transformers for image recognition at scale[EB/OL]. (2021-06-03) [2024-05-10]. https://arxiv.org/abs/2010.11929.

[32] KRIZHEVSKY A, SUTSKEVER I, HINTON G E. ImageNet classification with deep convolutional neural networks[J]. Communications of the ACM, 2017, 60(6): 84-90.

[33] HENDRYCKS D, BURNS C, BASART S, et al. Measuring massive multitask language understanding [EB/OL]. (2021-01-12)[2024-05-10]. https://arxiv.org/abs/2009.03300.

作者简介

杨赟辉　男,1994年11月出生于浙江省台州市. 现为中国电子科技集团公司第五十八研究所工程师,从事机器视觉、智能计算、高性能计算方面的研究工作.
E-mail: yangyunhui1315@qq.com

程　虎　男,1989年8月出生于江苏省徐州市. 现为中国电子科技集团公司第五十八研究所工程师,从事机器视觉、智能计算、高性能计算方面的研究工作.
E-mail: chhu1989@163.com

魏敬和　男,1970年1月出生于安徽省合肥市. 现为中国电子科技集团公司第五十八研究所研究员、博士生导师. 获国防科技进步二等奖等省部级奖励4项. 已授权发明专利50余项,国内外发表论文60余篇,出版专著1部.
E-mail: pume1975_cnjs@sina.com

刘国柱　男,1980年11月出生于江苏省盐城市. 现为中国电子科技集团公司第五十八研究所研究员、硕士生导师. 获省部级科技奖项4次. 已授权发明专利20余项,国内外发表论文30余篇.
E-mail: gzliucetc@163.com

桑贤侦　男,1992年10月出生于河南省商丘市. 现为中国电子科技集团公司第五十八研究所工程师,从事机器视觉、智能计算方面的研究工作.
E-mail: sangxianzhen@163.com