(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 113222133 A
(43)申请公布日 2021.08.06

(21)申请号 202110563720.9

(22)申请日 2021.05.24

(71)申请人 南京航空航天大学
地址 211106 江苏省南京市江宁区将军大道29号

(72)发明人 葛芬 崔晨晨 张伟枫 岳鑫 李梓瑜 周芳 吴宁

(74)专利代理机构 南京经纬专利商标代理有限公司 32200
代理人 沈海霞

(51)Int.Cl.
G06N 3/063(2006.01)
G06N 3/08(2006.01)
G06N 3/04(2006.01)
G06F 15/78(2006.01)
G06F 17/16(2006.01)

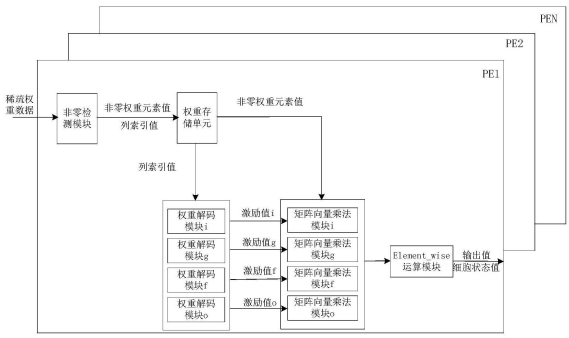
权利要求书1页 说明书5页 附图4页

(54)发明名称

一种基于FPGA的压缩LSTM加速器及加速方法

(57)摘要

本发明公开了一种基于FPGA的压缩LSTM加速器及加速方法,FPGA加速器内部包括多个计算单元、存储单元和控制单元;利用非零检测模块检测隐藏节点稀疏权重矩阵的非零权重元素值及对应的列索引值;利用权重解码模块根据列索引值找到对应的激励值;并将多个非零权重元素值及多个激励值送至矩阵向量乘法模块进行运算得到四个门的结果向量;Element_wise运算模块根据四个门的结果向量计算当前时刻的细胞状态值和输出值。在一个计算周期内仅仅将每个门的非零权重元素值和对应的激励值进行乘运算,从而保证在一个计算周期内每个矩阵向量乘法模块不会闲置,同时缩短了单个计算周期时间,从而提高了加速器的计算性能及吞吐量,同时节约了FPGA的片内缓存。



CN 113222133 A

1.一种基于FPGA的压缩LSTM加速器，其特征在于，所述FPGA加速器内部包括多个计算单元、存储单元和控制单元；

所述计算单元包括非零检测模块、权重存储单元、四个权重解码模块、四个矩阵向量乘法模块及Element_wise运算模块，所述非零检测模块用于检测隐藏节点稀疏权重矩阵的非零权重元素值及对应的列索引值；所述权重存储单元用于存储非零权重元素值及列索引值；所述权重解码模块根据列索引值找到输入激励对应的激励值；所述矩阵向量乘法模块将多个非零权重元素值及多个激励值进行乘累加运算得到单个门的结果向量，所述矩阵向量乘法模块并行运算得到四个门的结果向量；所述Element_wise运算模块根据四个门的结果向量计算当前时刻的细胞状态值和输出值；

所述存储单元用于缓存LSTM网络计算所需的权重数据、输入激励值、输出值以及细胞状态值；所述控制单元用于控制LSTM网络计算的状态转换和数据流传输过程。

2.根据权利要求1所述的一种基于FPGA的压缩LSTM加速器，其特征在于，所述Element_wise运算模块，采用时分复用的策略，将运算过程分成三个不同的状态周期，实际消耗的资源只有一个sigmoid激活函数模块，一个tanh激活函数模块，一个加法器和一个乘法器，最终得到当前时刻的细胞状态值以及输出值，具体包括以下计算步骤：

步骤S1.第一个周期将输入门对应的结果向量进行sigmoid函数激活计算得到输入门i，同时将记忆门对应的结果向量进行tanh函数激活计算得到记忆门g，然后将输入门i和记忆门g相乘；

步骤S2.第二个周期将遗忘门对应的结果向量进行sigmoid函数激活计算得到遗忘门f，将遗忘门f与上一时刻的细胞状态值C_{t-1}相乘后，再加上第一周期的输出$i\times g$，更新当前时刻新的细胞状态值C_t；

步骤S3.第三个周期将输出门对应的结果向量进行sigmoid函数激活计算得到输出门o，同时将新的细胞状态值C_t进行tanh函数激活计算，然后将两者进行乘法运算得到当前时刻的输出值h_t。

3.根据权利要求1所述的基于FPGA的压缩LSTM加速器的加速方法，其特征在于，包括以下计算步骤：采用块平衡剪枝算法对LSTM网络的权重矩阵进行剪枝，从而每个隐藏节点稀疏权重矩阵的每一行具有相同的剪枝率，将稀疏权重矩阵按照行序依次查找非零权重元素，将第一行中每个非零权重元素值及对应的列索引值记录在不同的地址空间中，后面每行非零权重元素值及对应的列索引值依次写入前面划分的地址空间中，从同一个地址空间内依次读取四个门对应的非零权重元素值及列索引值，根据列索引值找到输入激励对应的激励值，将四个门所有的非零权重元素值及对应的激励值并行乘累积运算得到四个门的结果向量，最后利用四个门的结果向量得到当前时刻的细胞状态值和输出数据。

一种基于FPGA的压缩LSTM加速器及加速方法

技术领域

[0001] 本发明涉及神经网络计算机硬件加速领域，尤其是涉及一种基于FPGA的压缩LSTM加速器及加速方法。

背景技术

[0002] 当前，LSTM网络在机器翻译、多语言处理、笔迹生成和图像标题生成等多种应用中取得了重大的成功，然而LSTM网络的计算以及存储复杂度随着网络模型规模的扩大也变得越来越高，选择合适的加速器平台就变得尤为重要。FPGA可以设计适应神经网络算法的硬件结构，开发人员可以根据自己的需求通过可编程的连接将FPGA内部的逻辑单元连接起来，来实现相应的功能。同时FPGA又可以在神经网络算法的硬件加速设计时根据算法特性来设计硬件架构。并且在综合计算和功耗这两个方面，FPGA相比于GPU具有更加出色的能耗比。因此，可编程性、可重构性、高并行性以及低功耗等优点使得FPGA很适合作为LSTM网络硬件加速的平台。

[0003] LSTM网络作为循环神经网络的一种变体，是一类以序列数据作为输入的递归神经网络，可以有效处理与时序相关的现实任务。主要通过引入门控机制来控制网络中信息累计的速度，相比于一般的循环神经网络，LSTM保存信息的周期更长。但是由于网络规模越来越大，导致神经网络的参数规模量也越来越巨大，运行LSTM网络时需要消耗大量的存储资源和运算资源，这严重制约了它在嵌入式平台或者在一些较小的移动设备上进行部署。由于大规模的LSTM网络模型存在大量的冗余参数，可以通过络剪枝算法对网络模型进行剪枝，合理的去除网络中存在的零值权重参数或者数值接近零的权重参数，再通过对剩余的稀疏权重参数进行重训练微调，使得网络模型的准确率基本保持不变，神经网络剪枝算法可以将LSTM网络模型大小有效的压缩，减少模型的存储量和计算量。

[0004] 目前，基于FPGA的LSTM加速器一般采用并行运算及并行读书数据来提升加速器的加速性能，如果将上述稀疏权重矩阵直接参与计算，由于存在多个零元素导致一个计算周期内较多运算单元闲置，从而导致整体效率不高。

发明内容

[0005] 本发明的目的是：提供一种基于FPGA的压缩LSTM加速器及加速方法，在一个计算周期内仅仅将每个门的非零权重元素值和对应的激励值进行乘运算，从而保证在一个计算周期内每个矩阵向量乘法模块不会闲置，同时缩短了单个计算周期时间，从而提高了加速器的计算性能及吞吐量，同时节约了FPGA的片内缓存。

[0006] 本发明的技术方案是：

[0007] 一种基于FPGA的压缩LSTM加速器，所述FPGA加速器内部包括多个计算单元、存储单元和控制单元；

[0008] 所述计算单元包括非零检测模块、权重存储单元、四个权重解码模块、四个矩阵向量乘法模块及Element_wise运算模块，所述非零检测模块用于检测隐藏节点稀疏权重矩阵

的非零权重元素值及对应的列索引值;所述权重存储单元用于存储非零权重元素值及列索引值;所述权重解码模块根据列索引值找到输入激励对应的激励值;所述矩阵向量乘法模块将多个非零权重元素值及多个激励值进行乘累加运算得到单个门的结果向量,所述矩阵向量乘法模块并行运算得到四个门的结果向量;所述Element_wise运算模块根据四个门的结果向量计算当前时刻的细胞状态值和输出值;

[0009] 所述存储单元用于缓存LSTM网络计算所需的权重数据、输入激励值、输出值以及细胞状态值;所述控制单元用于控制LSTM网络计算的状态转换和数据流传输过程。

[0010] 进一步地,所述Element_wise运算模块,采用时分复用的策略,将运算过程分成三个不同的状态周期,实际消耗的资源只有一个sigmoid激活函数模块,一个tanh激活函数模块,一个加法器和一个乘法器,最终得到当前时刻的细胞状态值以及输出值,具体包括以下计算步骤:

[0011] 步骤S1.第一个周期将输入门对应的结果向量进行sigmoid函数激活计算得到输入门i,同时将记忆门对应的结果向量进行tanh函数激活计算得到记忆门g,然后将输入门i和记忆门g相乘;

[0012] 步骤S2.第二个周期将遗忘门对应的结果向量进行sigmoid函数激活计算得到遗忘门f,将遗忘门f与上一时刻的细胞状态值C_{t-1}相乘后,再加上第一周期的输出i×g,更新当前时刻新的细胞状态值C_t;

[0013] 步骤S3.第三个周期将输出门对应的结果向量进行sigmoid函数激活计算得到输出门o,同时将新的细胞状态值C_t进行tanh函数激活计算,然后将两者进行乘法运算得到当前时刻的输出值h_t。

[0014] 同时本发明还提供一种基于FPGA的压缩LSTM加速器的加速方法,包括以下计算步骤:采用块平衡剪枝算法对LSTM网络的权重矩阵进行剪枝,从而每个隐藏节点稀疏权重矩阵的每一行具有相同的剪枝率,将稀疏权重矩阵按照行序依次查找非零权重元素,将第一行中每个非零权重元素值及对应的列索引值记录在不同的地址空间中,后面每行非零权重元素值及对应的列索引值依次写入前面划分的地址空间中,从同一个地址空间内依次读取四个门对应的非零权重元素值及列索引值,根据列索引值找到输入激励对应的激励值,将四个门所有的非零权重元素值及对应的激励值并行乘累积运算得到四个门的结果向量,最后利用四个门的结果向量得到当前时刻的细胞状态值和输出值。

[0015] 本发明的有益效果在于:在每个计算单元内,采用非零检测模块用于检测隐藏节点稀疏权重矩阵的非零权重元素值及对应的列索引值,并通过编码将非零权重元素值及对应的列索引值按照列方式存储在权重存储单元,保证每次从同一个地址取出的8个数据两两分别送至权重解码模块及矩阵向量乘法模块,实现了四个矩阵向量乘法模块并行的乘累加运算;由于采用块平衡剪枝算法保证了四个门的权重稀疏矩阵每行具有相同的非零元素数,因此划分的地址空间数和每行的非零元素数相同,从而保证在一个计算周期内每个矩阵向量乘法模块不会闲置,同时缩短了单个计算周期时间,从而提高了加速器的计算性能及吞吐量,同时节约了FPGA的片内缓存。

附图说明

[0016] 图1是基于FPGA的LSTM加速器的设计方法流程图;

[0017] 图2是控制单元的运算状态图；

[0018] 图3是Element_wise模块示意图；

[0019] 图4是一实施例中权重矩阵剪枝算法流程图；

[0020] 图5是一实施例中稀疏权重矩阵经过非零检测单元存入权重储存模块的数据示意图；

[0021] 图6是计算单元的运算架构图；

[0022] 图7是一具体实施例中权重储存单元、权重解码模块及矩阵向量乘法模块之间数据分配示意图。

具体实施方式

[0023] 下面将结合本发明实施例中的附图，对本发明实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例仅仅是本发明一部分实施例，而不是全部的实施例，本发明中的临时、第一均是为了说明算法训练中的不同阶段，没有限定意义。基于本发明中的实施例，本领域技术人员在没有作出创造性劳动前提下所获得的所有其他实施例，都属于本发明保护的范围。

[0024] 如图1所示，一种基于FPGA的压缩LSTM加速器，所述FPGA加速器内部包括多个计算单元(PE单元)、存储单元和控制单元；

[0025] 所述计算单元包括非零检测模块、权重存储单元、四个权重解码模块、四个矩阵向量乘法模块及Element_wise运算模块，所述非零检测模块用于检测隐藏节点稀疏权重矩阵的非零权重元素值及对应的列索引值；所述权重存储单元用于存储非零权重元素值及列索引值；所述权重解码模块根据列索引值找到输入激励对应的激励值；所述矩阵向量乘法模块将多个非零权重元素值及多个激励值进行乘累加运算得到单个门的结果向量，所述矩阵向量乘法模块并行运算得到四个门的结果向量；所述Element_wise运算模块根据四个门的结果向量计算当前时刻的细胞状态值和输出值；

[0026] 所述存储单元分为输入缓存、权重缓存及输出缓存，其中，权重缓存用于缓存LSTM网络计算所需的权重数据，输入缓存用于缓存输入激励值，输出缓存用户缓存输出值以及细胞状态值；所述控制单元用于控制LSTM网络计算的状态转换和数据流传输过程。

[0027] 如图2所示，所述控制单元用于控制LSTM网络计算的状态转换和数据流传输过程，其中，数据流传输具体如下：控制单元主要控制FPGA片外DRAM和片上BRAM的读写信号，所述稀疏权重矩阵是从片外DRAM读入，控制单元还控制整个前向推理运算过程中输入数据、权重数据以及计算中间数据的分配，同时还控制了加速器整体的计算逻辑；状态转换具体如下：加速器控制单元由三个状态组成，其中S0是空闲状态，等待LSTM计算开始；S1状态是将LSTM网络中的稀疏权重数据从权重缓存依次写入到N个并行计算单元的权重存储单元中；S2状态是LSTM前向推理算法的计算，其中包括矩阵向量乘法运算及Element-wise运算。当开始使能信号start有效时，加速系统从S0状态进入S1状态，开始从权重缓存中读取稀疏权重数据然后逐个写入计算单元的权重存储单元中；当数据写入完成时，write_done信号有效，系统开始进入S2状态，LSTM计算使能信号calculate有效开始第一个时刻的运算，当所有时刻运算均完成后，LSTM_done信号有效完成一层LSTM的加速计算，同时跳转回S0空闲状态开始下一层的LSTM网络计算。

[0028] 如图3所示，所述Element_wise运算模块，采用时分复用的策略，将运算过程分成三个不同的状态周期，实际消耗的资源只有一个sigmoid激活函数模块，一个tanh激活函数模块，一个加法器和一个乘法器，最终得到当前时刻的细胞状态值以及输出值，具体包括以下计算步骤：

[0029] 步骤S1.第一个周期将输入门对应的结果向量进行sigmoid函数激活计算得到输入门i，同时将记忆门对应的结果向量进行tanh函数激活计算得到记忆门g，然后将输入门i和记忆门g相乘；

[0030] 步骤S2.第二个周期将遗忘门对应的结果向量进行sigmoid函数激活计算得到遗忘门f，将遗忘门f与上一时刻的细胞状态值C_{t-1}相乘后，再加上第一周期的输出i×g，更新当前时刻新的细胞状态值C_t；

[0031] 步骤S3.第三个周期将输出门对应的结果向量进行sigmoid函数激活计算得到输出门o，同时将新的细胞状态值C_t进行tanh函数激活计算，然后将两者进行乘法运算得到当前时刻的输出值h_t。

[0032] 块平衡剪枝算法包括以下步骤：

[0033] 步骤1)采用深度学习框架Tensorflow来搭建LSTM网络模型，并利用数据集对模型进行参数训练得到初始模型；

[0034] 步骤2)对初始模型进行剪枝，通过设定剪枝的次数，多次重复剪枝操作保证模型准确率最佳；

[0035] 步骤3)设定剪枝的块大小，此后模型在剪枝和重训练时只对矩阵行中的每一个块进行细粒度的剪枝操作，并且会在训练算法过程中诱导权重矩阵按照每一个块中具体的阈值来独立修剪每个权重块；

[0036] 步骤4)重训练时，只对剪枝后剩余的稀疏权重参数进行调整，对已经剪枝的权重参数不作处理。

[0037] 重复剪枝和重训练的过程，直到LSTM网络模型达到预期的剪枝率并且模型准确率达到最佳。

[0038] LSTM网络剪枝主要包括三个部分，第一部分是调用Tensorflow的函数来进行模型的搭建，并下载常用的数据集进行训练得到初始模型。第二部分包括设定剪枝的次数，因为剪枝需要多次进行重训练微调来保证模型的准确率，因此一开始设定初始剪枝率时可以不用太大，然后逐步提升剪枝率达到最终的剪枝率；设定剪枝的块大小，并且按照每一个块中具体的阈值独立裁剪每个权重块，从而使得每个权重块都有相同的稀疏率；第三部分对剪枝后的权重矩阵进行重训练，并将该LSTM网络的准确率训练至最佳，然后重复剪枝和重训练的过程，使得LSTM网络模型达到预期的剪枝率和准确率。

[0039] 本发明还提供一种基于FPGA的压缩LSTM加速器的加速方法，如图4所示，包括以下计算步骤：

[0040] 步骤1.采用块平衡剪枝算法对LSTM网络的权重矩阵进行剪枝，从而每个隐藏节点稀疏权重矩阵的每一行具有相同的剪枝率，具体可以参照图5的剪枝过程；

[0041] 步骤2.利用控制单元从片外DRAM中读取稀疏权重矩阵并写入权重缓存中，将输入激励写入输入缓存中，接着将权重缓存中的稀疏权重矩阵依次送入各个计算单元中，每个计算单元的非零检测模块按照行序依次判定权重元素是否非零，将第一行中每个非零权重

元素值及对应的列索引值记录在权重存储单元不同的地址空间中，后面每行非零权重元素值及对应的列索引值依次写入前面划分的地址空间中；具体判定及存储过程中数据形式可参考图6，此时剪枝率为80％，即在权重存储单元中划分2个不同的地址空间来存储非零权重元素值及对应的列索引值；

[0042] 步骤3.从同一个地址空间内依次读取四个门对应的非零权重元素值及列索引值并两两分割依次送至各个门对应的权重解码模块及矩阵向量乘法模块，权重解码模块根据列索引值找到输入激励对应的激励值，四个矩阵向量乘法模块根据对应的所有非零权重元素值及对应的激励值并行乘累积运算得到四个门的结果向量，最后利用Element_wise运算模块将四个门的结分别经过对应的激活函数得到当前时刻的细胞状态值和输出值。

[0043] 上述步骤2及步骤3对应的计算单元均采用并行运算，如果计算单元的个数少于隐藏节点的个数，采用复用策略；为了进一步对上述加速过程进行说明，如图7所示给出了一个具体实施例，其中PE-weight是每个计算单元的权重存储单元，权重解码模块为位选择器，矩阵向量乘法模块采用乘法器及加法器实现。

[0044] 由于采用块平衡剪枝算法保证了四个门的权重稀疏矩阵每行具有相同的非零元素数，因此划分的地址空间数和每行的非零元素数相同，从而保证在一个计算周期内每个矩阵向量乘法模块不会闲置，同时缩短了单个计算周期时间，从而提高了加速器的计算性能及吞吐量，同时节约了FPGA的片内缓存。

[0045] 本发明FPGA的加速方法同时节约了FPGA的片内缓存，本发明的块剪枝结合非零检测单元的压缩方法和其他针对神经网络剪枝后的稀疏矩阵进行压缩编码的方式(坐标压缩COO，压缩稀疏行CSR，压缩稀疏列CSC和游程编码RLC)进行对比。由表1可知，虽然前四种方法也可以取得不错的网络压缩率，但是这些编码方式大多是针对采用细粒度剪枝后的稀疏权重矩阵，剪枝后的稀疏权重矩阵仍然是不规则的，不利于硬件的并行运算，并且在硬件实现时需要设置专门的硬件电路来实现矩阵运算，从而来提高硬件计算效率。本文在LSTM网络的剪枝算法上采用了块平衡稀疏的剪枝方式，这种剪枝方式得到的权重矩阵更加规则，取得了良好的压缩效果并适用于硬件加速器的设计实现。

[0046]

编码方式	压缩前存储量	压缩后存储量	压缩率
COO	20KB	9.1KB	55％
CSR	20KB	6.2KB	70％
ELL	20KB	10.8KB	45％
RLC	20KB	5.3KB	73％
本发明	20KB	5.1KB	75％

[0047] 表1

[0048] 以上实施例仅表达了本发明优选实施方式，其描述较为具体和详细但并不能因此而理解为对本申请专利范围的限制。对于本领域的技术人员来说，本发明可以有各种更改和变化。凡在本发明的精神和原则之内，所作的任何修改、等同替换、改进等，均应包含在本发明的保护范围之内。

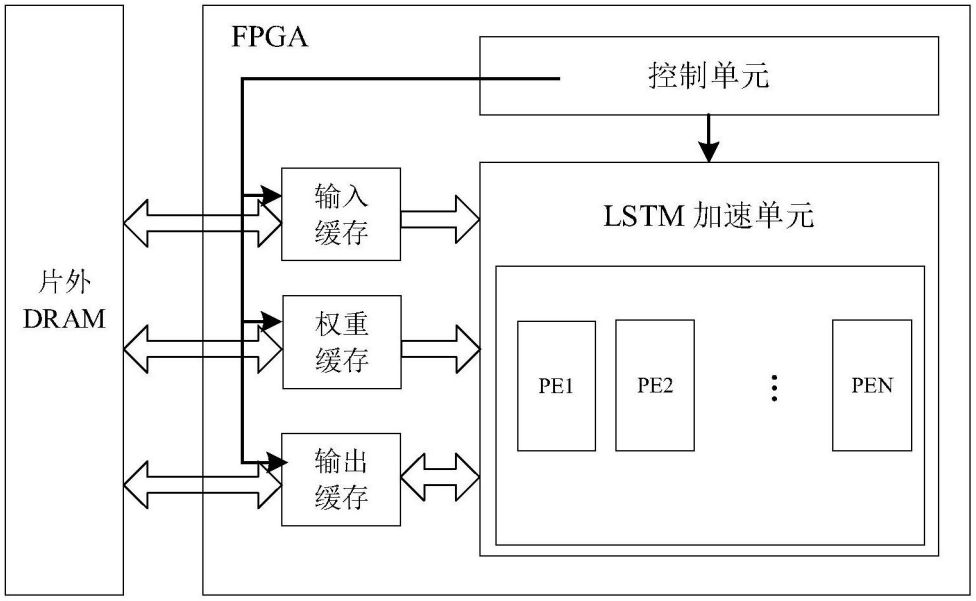

FPGA
控制单元
输入
缓存
LSTM 加速单元
片外
DRAM
权重
缓存
PE1
PE2
PEN
输出
缓存

图1

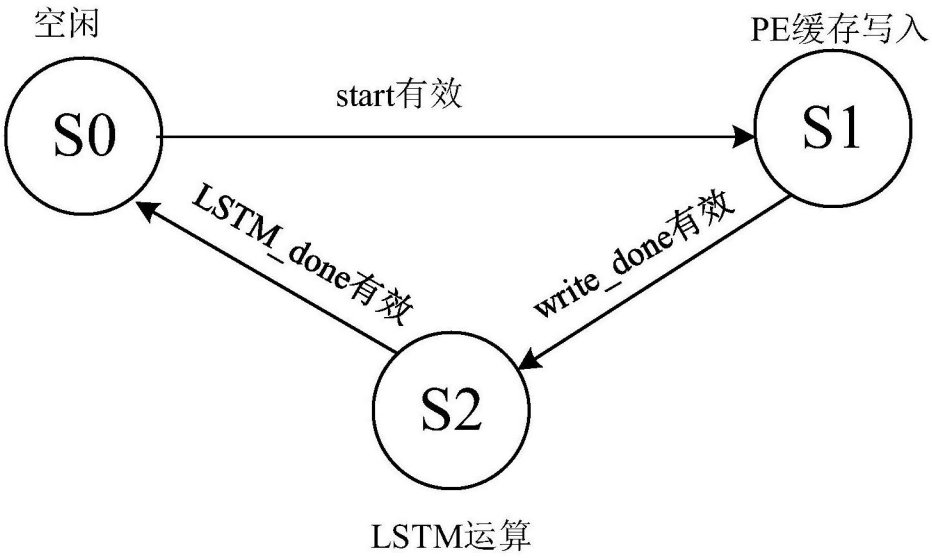

空闲
PE缓存写入
S0
start有效
S1
LSTM_done有效
write_done有效
S2
LSTM运算

图2

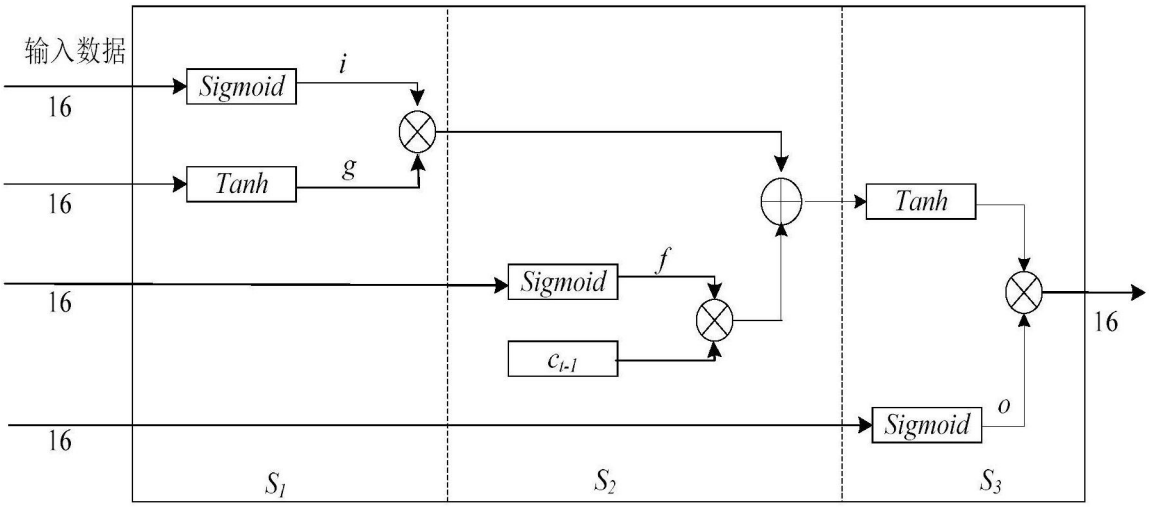

输入数据
16
16
16
16
Sigmoid
Tanh
Sigmoid
Sigmoid
Tanh
i
g
f
o
c_{t-1}
16
S_1
S_2
S_3

图3

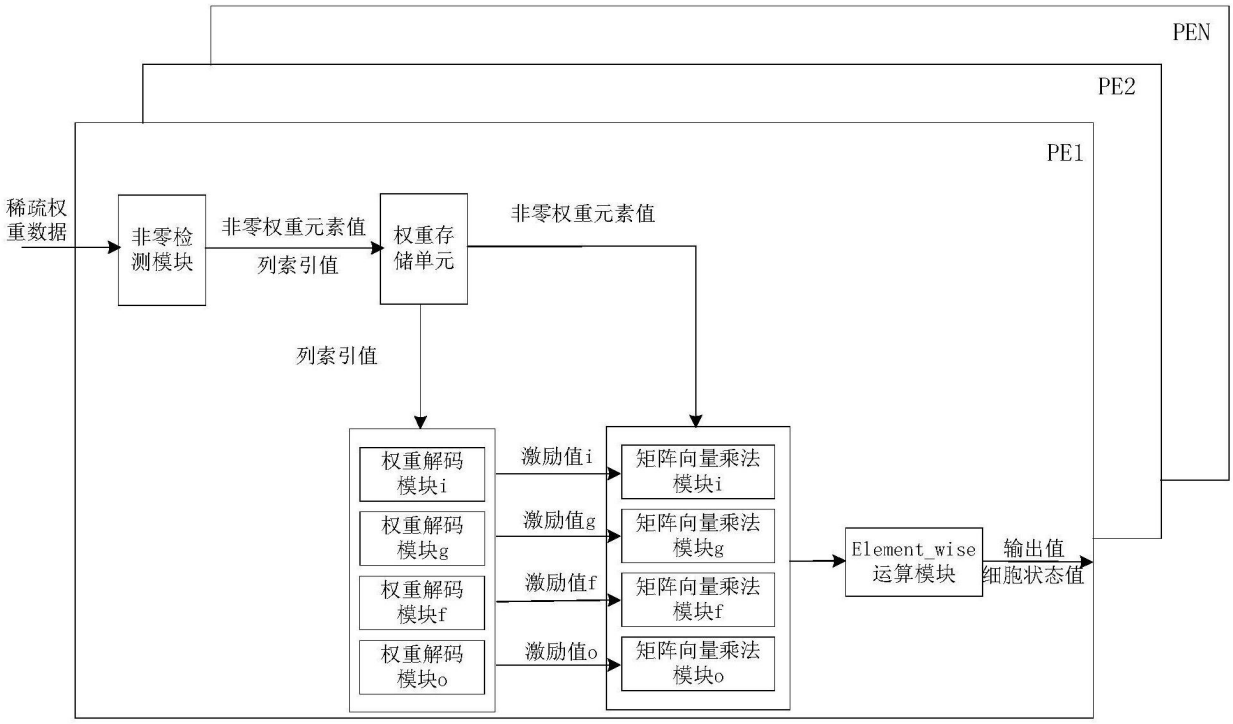

PEN
PE2
PE1
稀疏权重数据
非零检测模块
非零权重元素值
列索引值
权重存储单元
非零权重元素值
列索引值
权重解码模块i
权重解码模块g
权重解码模块f
权重解码模块o
激励值i
激励值g
激励值f
激励值o
矩阵向量乘法模块i
矩阵向量乘法模块g
矩阵向量乘法模块f
矩阵向量乘法模块o
Element_wise运算模块
输出值
细胞状态值

图4

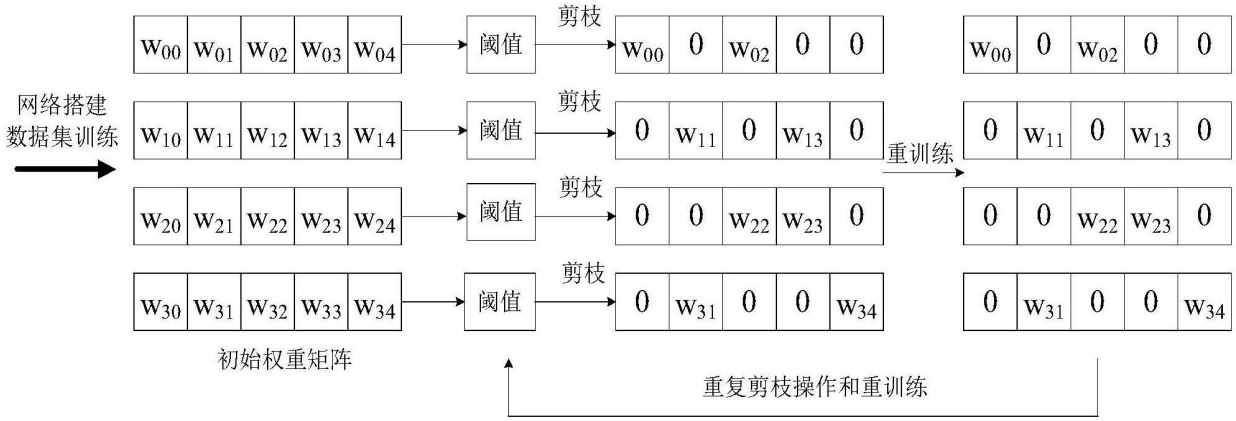

网络搭建
数据集训练
W00 W01 W02 W03 W04
W10 W11 W12 W13 W14
W20 W21 W22 W23 W24
W30 W31 W32 W33 W34
初始权重矩阵
阈值
剪枝
W00 0 W02 0 0
0 W11 0 W13 0
0 0 W22 W23 0
0 W31 0 0 W34
重训练
重复剪枝操作和重训练

图5

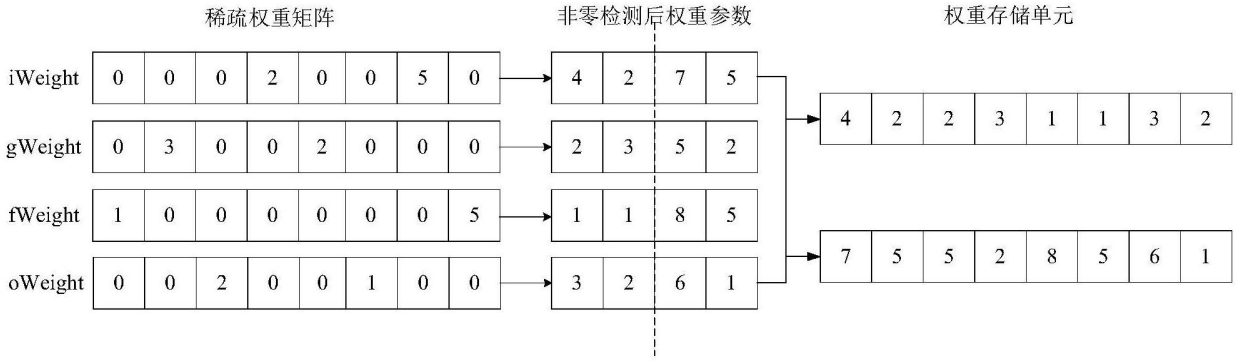

稀疏权重矩阵
非零检测后权重参数
权重存储单元
iWeight 0 0 0 2 0 0 5 0
gWeight 0 3 0 0 2 0 0 0
fWeight 1 0 0 0 0 0 0 5
oWeight 0 0 2 0 0 1 0 0
4 2 7 5
2 3 5 2
1 1 8 5
3 2 6 1
4 2 2 3 1 1 3 2
7 5 5 2 8 5 6 1

图6

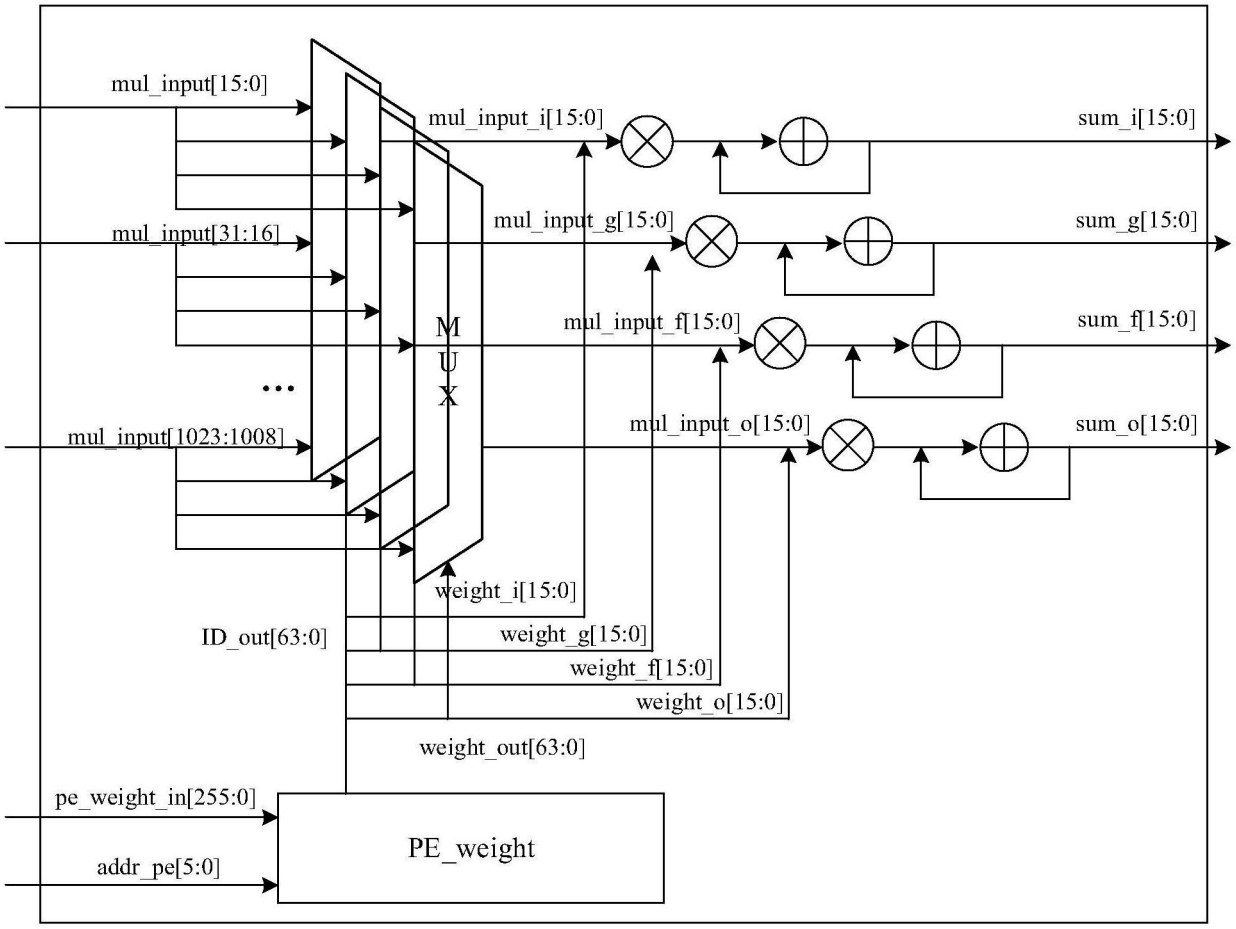

mul_input[15:0]
mul_input[31:16]
mul_input[1023:1008]
...
M
U
X
mul_input_i[15:0]
mul_input_g[15:0]
mul_input_f[15:0]
mul_input_o[15:0]
sum_i[15:0]
sum_g[15:0]
sum_f[15:0]
sum_o[15:0]
weight_i[15:0]
ID_out[63:0]
weight_g[15:0]
weight_f[15:0]
weight_o[15:0]
weight_out[63:0]
pe_weight_in[255:0]
addr_pe[5:0]
PE_weight

图7