高技术通讯 2022 年 第32 卷 第7 期:663-673

doi:10.3772/j.issn.1002-0470.2022.07.001

针对图神经网络加速器性能评估的标准测试集①

宋新开②[*,**,***] 支 天[*,***] 孔维浩[*,**,***] 杜子东③[*,***]

([*]中国科学院计算技术研究所计算机体系结构国家重点实验室 北京 100190)

([**]中国科学院大学 北京 100049)

([***]中科寒武纪科技股份有限公司 北京 100191)

摘 要 图神经网络(GNN)算法在图结构数据处理任务中取得了突破性的成功。然而,针对图神经网络硬件加速器设计的研究缺乏明确的设计目标和统一的评价标准。本文提出一种针对图神经网络硬件加速器性能评估的标准测试集(BenchGNN)。BenchGNN 包括宏测试集和微测试集2 部分。宏测试集包含了3 种主要任务类型的图神经网络算法和5 个典型应用领域的数据集。微测试集包含2 种微观操作类型和4 种不同量化特性的数据集。本文在现有运算设备中央处理器(CPU)、图形处理器(GPU)和图神经网络专用加速器上进行了 BenchGNN 的实验测试。实验结果表明,CPU 由于并行度不高而无法高效处理图神经网络算法。针对图神经网络算法的随机访存行为进行优化的专用加速器取得了优于通用并行处理器 GPU 的性能功耗表现。根据 BenchGNN 的评估结果,在图神经网络加速器设计过程中需要重点考虑运算并行度和随机访存优化这两种因素。

关键词 图神经网络(GNN);加速器;标准测试集

0 引 言

图神经网络(graph neural network,GNN)是近年来兴起的一种专门用来处理基于图结构数据的人工智能算法。该算法已经在各类图处理任务上实现了准确度的突破性进展,例如在电子商务[1]、分子生物学[2-3]、社交网络[4-5]、知识图谱[6]等领域[7-9]。图神经网络算法是卷积神经网络(convolutional neural network,CNN)和循环神经网络(recurrent neural network,RNN)等传统神经网络在图数据处理任务上的扩展。该算法将传统神经网络算法和图分析算法结合起来,弥补了传统神经网络算法不能处理图结构数据的问题。

随着图神经网络算法的迅速发展和应用,图神经网络性能优化问题开始受到研究人员的关注。近年来,已经有许多针对图神经网络算法设计专用硬件加速器的研究工作被发表[10-19]。他们提出了不同的设计以改善现有设备运行图神经网络算法时效率低的问题。然而,这些图神经网络硬件加速器研究工作在测试样例的选择上差异很大,缺乏明确的设计目标和评价手段。为了推动图神经网络硬件加速器研究的发展,学术界迫切需要一套针对硬件加速器研究的图神经网络标准测试集。

设计一套针对图神经网络硬件加速器评估的有效的标准测试集是一件有挑战的任务,本文从下列3 个方向梳理了该工作的挑战性和对应的解决思路。

首先,如何从大量的图神经网络算法中选择一

① 国家自然科学基金(61925208,61732007,61732002,61906179,62002338,U19B2019,U20A20227),北京市自然科学基金(JQ18013),中国科学院战略性先导科技专项(XDB32050200),北京智源人工智能研究院以及北京市科技新星计划(Z191100001119093)和中国科学院青年创新促进会和科学探索奖资助项目。

② 男,1993 年生,博士生;研究方向:计算机系统结构,人工智能算法;E-mail:songxinkai@ict.ac.cn。

③ 通信作者,E-mail:duzidong@ict.ac.cn。

(收稿日期:2021-05-05)

部分作为标准测试集是有挑战性的。为了控制执行图神经网络加速器评估的效率和成本,标准测试集无法全部包含已公开发表的图神经网络算法。对此,本文的解决思路是从图神经网络算法的主要任务类型和应用领域出发选择典型代表性算法和数据集。

其次,如何在选择尽可能少的数据集的情况下保证标准测试集中数据集选择的多样性是非常重要而且具有挑战性的。数据集选择的重要性体现在图神经网络加速器的性能优化设计与数据集的特性关系密切。例如,数据集的每个图的顶点数量直接影响到加速器片上缓存大小的设置和访存行为的优化。顶点的连接稀疏度不仅影响芯片存储结构的设计,而且对芯片的运算单元设计也有非常大的影响。数据集选择的挑战性体现在各种图神经网络算法可使用的数据集非常多。本文调研了与图神经网络算法相关的可公开获取的图数据集共 326 个,对它们的关键特性进行量化和分析,并选取最大化数据集多样性的方案。

最后,如何设计标准测试集使研究人员可以通过评估结果来揭示和分析硬件加速器的性能瓶颈也是一大挑战。一个有效的标准测试集需要能够根据评估结果来分析加速器的性能瓶颈。本文通过对标准测试集中的程序样例的运算步骤进行拆分梳理,对其中的关键操作类型进行分类测试,以揭示加速器性能优化的瓶颈,进而帮助研究人员改进设计。

本文提出的图神经网络标准测试集(Benchmark for graph neural network, BenchGNN)解决了上述三大挑战。BenchGNN 包括宏测试集和微测试集两部分,其中,宏测试集从图神经网络任务类型和应用领域的角度选取代表性算法和数据集,而微测试集则包括图神经网络算法中包含的两种基础操作类型和 4 个不同规模特性的图数据集。

本文的主要贡献如下。

(1)提出了一种针对图神经网络硬件加速器评估的标准测试集 BenchGNN。该测试集包含多种主要任务类型和应用领域,同时还包括用于分析硬件加速器的设计优劣的微测试集。

(2)BenchGNN 解决了图神经网络加速器性能测评结果严重依赖于数据集选取的问题,通过对 326 个数据集进行量化分析进而选出代表性的数据集。

(3)在现有运算设备上对 BenchGNN 进行了实验测试。实验结果表明,BenchGNN 可以展示出不同设备在处理图神经网络运算的不同任务时各自的优劣所在。

1 相关工作

本节将介绍图神经网络算法、图神经网络硬件加速器和图神经网络标准测试集的背景知识和相关工作,并说明设计一款针对图神经网络硬件加速器的标准测试集的必要性。

1.1 图神经网络算法

图神经网络算法是一种处理图数据的神经网络算法,该算法以图数据为输入,根据不同的任务类型输出不同类型的数据结果。例如,处理顶点级任务的图神经网络算法输出每个顶点的分类或回归信息,处理边级任务的图神经网络算法预测每条边的存在和类别,处理图级任务的图神经网络算法输出整个图的分类或者回归结果。

图神经网络由多层组成,每层以图数据为输入,输出具有新的顶点特征向量或新的图拓朴结构的图数据。输入图数据先后经过这些层的处理,最终得到对图数据进行特征提取后的结果。根据任务需求的不同,再根据这个包含新特征的图数据样本预测最终输出结果,例如预测每个顶点的分类信息,预测每条边的分类信息或者预测整个图的类别信息。

图神经网络层的基本计算过程包括邻居顶点聚合和特征向量转换这两个主要步骤。如图 1 所示,以图中的 2 号顶点为例,先执行邻居顶点聚合运算,将其邻居顶点的特征向量聚合为一个中间结果向量。然后再进行特征向量转换,2 号顶点的中间结果向量经过一个内积层与权值矩阵相乘得到 2 号顶点的输出向量。对所有顶点都执行上述步骤进行特征向量转换,就是一个基础图神经网络层的运算过程。

根据一项开源项目的统计,2016 年 9 月至 2020

年 3 月已经有至少 1287 篇与图神经网络算法相关的论文发表。这导致在对图神经网络硬件加速器进行测试时,无法对全部图神经网络算法进行测试,只能选择其中具有代表性的算法进行测试。

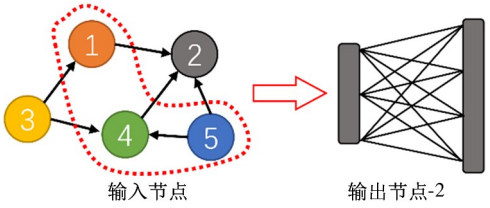

图 1　图神经网络的基本运算过程

1.2　图神经网络硬件加速器

自从 2019 年 HyGCN[12] 设计被发表之后,已经有共计 10 篇针对图神经网络算法设计硬件加速器的研究工作被发表,包括 AWB-GCN[10]、EnGN[11]、GRIP[15] 和 Cambricon-G[19] 等。

从事图神经网络硬件加速器研究的团队在测试算法的选择上展现出巨大的差异性。本文整理了这些图神经网络硬件加速器论文在性能评估时使用的测试集,图 2 所示是到 2020 年 3 月为止发表的 10 篇图神经网络加速器所选择的测试算法的统计。从算法选取的角度来看,在全部 14 个被用于评估加速器性能的图神经网络算法中,有 10 个算法都仅被一个加速器用于评估,仅有 GCN 算法被全部 10 个加速器共同选取。图 3 展示了现有加速器评估数据集的选取情况,在被用于评估的 30 个数据集中,有 21 个数据集是仅被一个加速器用于评估。用于评估图神经网络硬件加速器设计的测试集选取的巨大差异性无法在同行之间进行直观的对比,阻碍了图神经网络加速器研究的进一步发展。

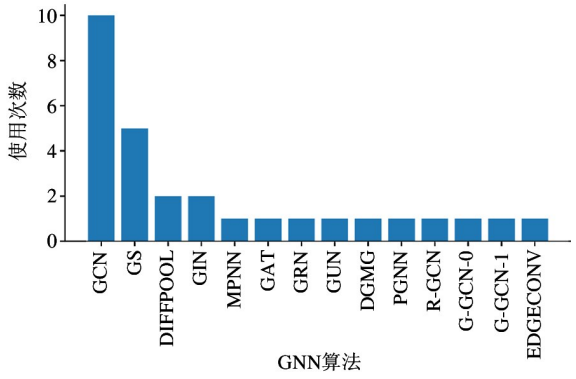

图 2　现有加速器选用的测试算法

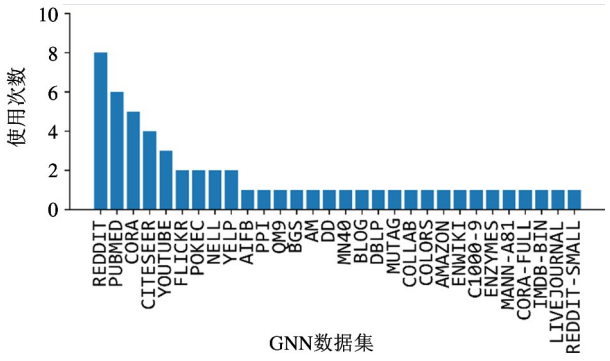

图 3　现有加速器选用的测试数据集

1.3　图神经网络标准测试集

目前,神经网络领域针对硬件性能优化的标准测试集的典型代表是 MLPerf[20],该测试集是一个针对神经网络各应用领域的权威测试集,在学术界和工业界被广泛应用。MLPerf 测试集的设计面向各种不同规模类型的硬件设备,包括移动端设备和高性能设备等。同时 MLPerf 还包含各种主流神经网络的类型,包括卷积神经网络、循环神经网络、Transformer 和深度强化学习等。但是,MLPerf 中还没有任何与图神经网络相关的测试内容。本文的研究内容可以弥补 MLPerf 在图神经网络相关方向测试内容的缺失。

另外一部分和图神经网络相关的标准测试集研究工作包括 open graph Benchmarking(OGB)[21] 和 Benchmarking GNN[22] 等。OGB 包括一些中等规模的真实的图数据集并且对这些数据集进行划分来实现对算法泛化能力的评估。Benchmarking 图神经网络由 8 个数据集组成,包括 4 个人工合成的数据集,2 个半人工合成的数据集和 2 个真实的数据集。其设计重点是提高标准测试集针对不同图神经网络算法性能和鲁棒性的区分度。

当前提出的这些图神经网络标准测试集都是图数据集的集合,其设计目的是用于评估各种图神经网络算法的识别准确度。它们不适用于图神经网络硬件加速器性能评估的原因具体体现在以下两点。第一,不同图神经网络算法的运算模式对加速器设计影响很大。而现有图神经网络标准测试集只包含各种图数据集,没有对图神经网络算法进行挑选。第二,这些标准测试集在挑选数据集时没有从性能

和能耗优化的角度进行考虑。综上所述,现有的图神经网络标准测试集都无法满足图神经网络加速器评估的需求。

2 标准测试集 BenchGNN

本节将介绍本文所提出的图神经网络硬件加速器测评标准测试集的具体内容。BenchGNN 分为宏测试集和微测试集两部分。宏测试集以整个图神经网络算法为测试单位,包括各主要类型的图神经网络算法和多种主要应用领域的数据集,用来评估图神经网络加速器的整体性能和功耗表现。微测试集以微观操作类型为测试单位,包括两种操作类型和 4 种不同规模尺寸的数据集。微测试集用来分析图神经网络加速器在处理不同运算模式和规模尺寸时的优劣之处,进而为设计改进提供启发。

2.1 宏测试集

宏测试集是用来评估图神经网络加速器的宏观性能和功耗表现的测试样例集合,以整个图神经网络算法为测试单位。宏测试集中测试程序的选取考虑了算法类型和应用领域这两方面,包括 3 种主要算法类型,分别是顶点分类(node classification)任务、图分类(graph classification)任务和连接预测(link prediction)任务。应用领域包括社交网络领域、文献检索领域、生物学领域、知识图谱和语言学领域。宏测试集的具体内容如表 1 所示,包括模型的参数量、所需的计算量和需要达到的精度。

表 1 宏测试集列表

任务类型	应用领域	算法	数据集	精度	操作数/GOPS	模型体积/MB
顶点分类	社交网络	GCN	Reddit	95.6%	160	0.62
顶点分类	文献检索	GAT	Cora	84%	0.506	0.35
图分类	生物学	DiffPool	Enzymes	63.3%	0.68	0.13
连接预测	知识图谱	CompGCN	FB15k-237	0.355 MRR	22.0	1.22
连接预测	语言学	CompGCN	WN18RR	0.479 MRR	7.08	1.22

注:MRR(mean reciprocal rank)是连接预测任务的精度指标

宏测试集中选取的算法介绍如下。

图卷积网络(graph convolutional network,GCN)[4]是最具有代表性的图神经网络算法。该算法是为了解决图数据的半监督顶点分类问题而提出的。GCN 中的图卷积层可以把图中每个顶点的特征向量转换为新的特征向量,其结果可以通过 Softmax 运算得到顶点类别预测结果。式(1)和式(2)是图卷积层运算的 2 个步骤。首先,将图中每个顶点的所有邻居顶点的特征向量聚合为一个向量;然后,该聚合向量再乘以权值矩阵,得到每个顶点的新的特征向量作为图卷积层的输出。GCN 算法的上述 2 个步骤在各种图神经网络算法中具有普适性和代表性。

$$Y_i = Reduce(X_i),\quad j \in Neighbor_i \tag{1}$$

$$Z_i = Y_i \times W \tag{2}$$

图注意力网络(graph attention network,GAT)[23]将注意力机制引入到图神经网络算法中,提出了图注意力层。在图注意力层中,首先根据每个顶点的特征向量计算出该顶点的两个自注意力分数值,分别代表本顶点作为一条边的源顶点和目的顶点时的注意力值;然后根据每条边的两端顶点的注意力值计算出该边的注意力值;最后在之后的聚合过程中使用上述计算得到的每条边的注意力值作为权重执行邻居顶点聚合运算。

可微池化算法(differentiable pooling,DiffPool)[2]是图分类算法的典型代表。该算法引入了图池化层操作,可以对图拓扑结构数据进行下采样,减少图中顶点的数量,增大顶点的感受野,提炼图的高层次信息。图池化层可以对图拓扑数据进行粗化,经过粗化后的图中的顶点数量减少,相应的顶点特征向量包含更多的全局信息,最终可以将这些顶点特征向量进行全局聚合,得到一个向量来表示整个图的特征信息。DiffPool 是图池化神经网络的典型代表,该

算法使用矩阵乘法的方式更新顶点的聚类分组信息,实现了可微分的池化操作。

多关系组合图卷积网络(composition-based multi-relational graph convolutional networks, CompGCN)[6]是连接预测算法的典型代表,在知识图谱的实体关系补全任务中取得优异表现。该算法解决了知识图谱中连接关系类型多样性导致的参数数量爆炸问题,提出了组合连接关系编码的图神经网络聚合方式。同时,CompGCN 还通过数据增广的方式将连接关系划分为正向、反向和自旋 3 种类型,分别学习 3 种权值矩阵,并对它们的运算结果进行加权求和。

最后,为了明确具体测试标准,以下罗列了宏测试集中的 4 种图神经网络算法的具体超参数。GCN 算法包括 2 个 GCN 层,其中间层的特征向量长度为 256。GAT 算法同样包括 2 个 GAT 层,其中间特征向量长度为 8,2 个 GAT 层的注意力通道分别为 8 和 1。DiffPool 算法包括 1 个输出特征向量长度为 64 的 GCN 层,1 个聚合类型数量为 12 的 DiffPool 层,该层对应的特征向量长度为 64,以及 1 个全局池化层和最终的图分类层。CompGCN 算法采用 TransE 作为连接预测的计分函数,网络结构包含 2 个 GCN 层,其中间层的特征向量长度为 200。

宏测试集所选取的数据集都是图神经网络算法研究领域的常用测试数据集。其中,顶点分类任务的常用数据集 Cora[4]是表示科学文献之间的互相引用关系的图数据。以 2708 篇文献为顶点,10 556 条引用关系为边,任务目标是对每篇文献进行 7 选 1 分类。Reddit[4]也是顶点分类任务的常用数据集,包含 232 965 个表示社交发贴的顶点和 114 615 892 条边,每条边表示 2 个发贴被同一网络用户留言的相关关系,任务目标是对每个网络发帖进行分类。图分类任务的常用数据集 Enzymes[2]是一个包含 600 个蛋白质三级结构的数据集,用于根据每个蛋白质的氨基酸组成结构预测蛋白质属性。连接预测任务的常用数据集 FB15k-237[6]和 WN18RR[6]分别来自知识图谱领域和语言学领域,顶点表示实体概念,边表示这些实体之间的相互关系。这些图数据都是由多个“实体-关系-实体”三元组组成,连接预测任务需要预测两个实体顶点之间的边是否存在以及预测边的类型。

2.2 微测试集

本文除了提出上述宏测试集对图神经网络加速器的性能功耗进行总体评估之外,还提出一系列微测试集对加速器的微观性能功耗表现进行测试。具体来说,微测试包含图神经网络运算中需要的 2 种操作类型和 4 种不同规模尺寸的图数据集。通过对这些不同细分类型的微观运算场景进行分类测试,微测试集的测试结果可以用来分析图神经网络加速器的性能功耗优化的不足之处,进而启发设计人员进行针对性的改进。

微测试集的 2 种操作类型分别是随机向量规约操作和矩阵乘法操作。这 2 种操作类型是通过对宏测试算法的运算过程进行拆解所得到的。表 2 列举了宏测试集中 4 种算法所包含的主要运算模式及其操作类型。

表 2 图神经网络算法操作类型分析

运算类型	GCN	GAT	DiffPool	CompGCN	主要操作类型
顶点聚合运算	有	有	有	有	随机向量规约
特征向量转换运算	有	有	有	有	矩阵乘法
注意力运算	无	有	无	无	矩阵与向量乘法
可微池化运算	无	无	有	无	矩阵乘法

顶点聚合运算是图神经网络算法的基础运算类型之一。图聚合运算是指在图上的顶点特征信息按照顶点之间的边的连接关系进行信息传递的过程。图聚合运算最常见的做法是每个顶点将邻居顶点的信息聚合到本顶点,具体的聚合方法包括求和、求均值或求最大值等,如式(1)所示。该过程的核心操作类型就是随机向量规约操作,即取随机位置的向量组合执行规约运算。因此,本文选择随机向量规

约操作作为微测试集中的一种操作类型。

图特征转换运算是指对图中的特征向量进行转换的过程,其操作对象可能包括每个顶点、每条边或者整个图的特征向量。图特征转换运算的具体操作类型为矩阵乘法操作,即每个特征向量与图神经网络中的一个权值矩阵相乘,得到对应对象的输出特征向量。该运算不仅可以用于将输入特征信号转换为隐空间的特征信号,也可以用于在不同层的隐空间之间进行转换或者从隐空间转换为具有语义信息的输出空间的特征信号,例如转换为代表输出的类别预测信息的特征向量。除此之外,表 2 中的注意力运算和可微池化运算的核心操作类型也都是矩阵乘法操作。矩阵乘法操作具有运算量大、访存连续性强、数据复用规则清晰的特点,这与前述随机向量规约操作有明显区别。因此本文选择矩阵乘法操作为微测试集中的第二种操作类型。

除了操作类型之外,数据集的选择对图神经网络加速器性能优化设计的影响也很大。例如,图数据中每个图的顶点数量和每个顶点的特征向量长度共同决定了该图的数据体积。在顶点聚合运算过程中,由于每个顶点可能被多个其他顶点连接,所以每个顶点可能需要多次被访问。在这种情况下,对于每个图的顶点数据体积较小的数据集,可以将顶点特征向量全部缓存在片上存储中,从而避免重复进行片外访存带来的性能损失。但是,对于顶点数据体积远超芯片片上存储空间的图数据集,如何做好片上存储层次和访存复用就成为加速器优化设计的关键所在。综上所述,数据集对加速器优化设计的影响很大,所以必须专门挑选微测试集所用的图数据集的规模尺寸特性以保证微测试集评估的多样性。

图数据集的规模尺寸特性主要体现在 3 个方面,分别是顶点数量、边数量和顶点特征向量长度。但是由于图神经网络运算过程中只有第一层的顶点特征向量长度与原数据集一致,其后的所有图神经网络层中顶点特征向量长度均为模型所指定的长度,因此本文没有选取顶点特征向量长度作为数据集的筛选指标。同时,本文使用连接稠密度来替代边数量作为数据集的筛选指标。

本文统计了 326 个真实的图数据集的顶点数量和图的连接稠密度。然后,根据这两个量化特性对数据集使用 K-Means 算法进行聚类,类别数设置为 4。最后,选取距离每个聚类中心最近的数据集作为微测试集中使用的数据集。聚类中心和最后选取的数据集如图 4 所示。这 4 个图数据集分别是 Enzymes(ENZ)、computer science(CS)、AM 和 FRI。Enzymes 是蛋白质结构数据库,其中包含 600 个蛋白质三级结构的数据集。CS 是论文共同作者关系图数据,来自计算机领域顶级会议的接收论文的共同作者数据。FRI 是社交网络中的好友关系图数据,来自社交网站 Friendster。

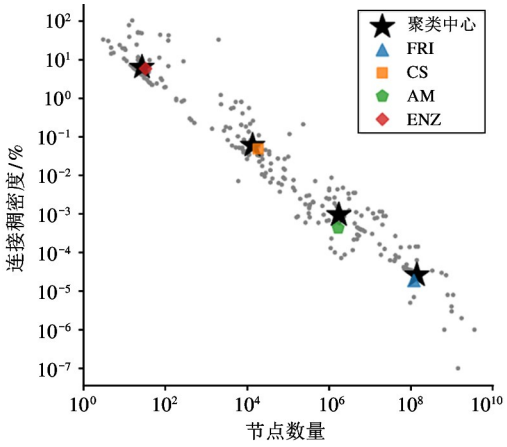

图 4　数据集特性的聚类分析图

这 4 个数据集的规模尺寸如表 3 所列,包括图数量、顶点数量、边数量和连接稠密度。除此之外,本文根据统计经验设置了 16、64 和 256 这 3 种常用的特征向量长度,见表 4。可见,在不同的顶点特征向量长度设置下,图数据集的顶点特征向量的总体积从 2.04 kB 到 112 GB 均有分布。微测试集包含图数据集的各种规模尺寸,以分析不同微观运算场景下的硬件加速器设计表现。

表 3　微测试集数据集的规模特性

数据集	图数量	顶点数量	边数量	连接稠密度/%
ENZ	600	19 580	37 282	5.836 29
CS	1	18 333	163 788	0.048 73
AM	1	1 666 764	11 976 642	0.000 43
FRI	1	117 751 379	2 586 147 869	0.000 019

表 4　微测试集数据集的顶点特征向量总体积

特征向量长度	16	64	256
ENZ	2.04 kB	8.16 kB	32.6 kB
CS	1.12 MB	4.48 MB	17.9 MB
AM	102 MB	407 MB	1.59 GB
FRI	7.02 GB	28.1 GB	112 GB

3　实验测试

为了展示 BenchGNN 的实际效果,本文在典型硬件设备上对 BenchGNN 进行了实验测试,包括中央处理器(central processing unit,CPU)、图形处理器(graphics processing unit,GPU)和图神经网络加速器。本文实验所用的 CPU 为 Intel(R) Xeon(R) CPU E5-2690 v4,GPU 为 NVIDIA Tesla P100-16 GB,选用的图神经网络专用加速器为 Cambricon-G[19]。这 3 种硬件设备的关键特性列举在表 5 中。其中,Cambricon-G 的功耗为其论文中所列数据,该数据为芯片静态功耗,且不包含片外存储的功耗。

表 5　实验设备的关键特性

	E5-2690 v4	P100	Cambricon-G
峰值算力	582.4 GFlops	9.3 TFlops	4 TFlops
访存带宽	76.8 GB/s	720 GB/s	128 GB/s
内存	128 GB DDR4	16 GB HBM2	16 GB HBM
片上存储	38.9 MB	22.8 MB	12 MB
板卡功率	135 W	250 W	3.62 W

为了保证测试实验能够准确地反映设备的最佳性能功耗表现,针对 CPU 和 GPU 的实验过程使用的是当前最先进的图神经网络软件框架 DGL(deep graph library)。其中,宏测试集算法 CompGCN 不支持在 DGL 框架中实现,因此使用的是论文对应的开源代码。对于专用加速器 Cambricon-G,本文首先根据公开论文编写软件模拟器,然后针对每个算法的运算过程使用脚本生成指令,最后在模拟器上运行指令对其性能和功耗进行评估测试。

本文使用上述 3 种硬件设备分别运行了宏测试集的 5 个测试程序,对其性能和功耗结果进行了评估和分析。图 5 是宏测试集性能测试结果,图中展示了 3 种运算设备分别运行宏测试集的推理时间,单位是毫秒(ms)。为了显示清晰,本文在图中使用缩写 Cam-G 代表图神经网络专用加速器 Cambricon-G。可见,CPU 的性能表现远差于具有较高并行运算能力的 GPU 和 Cambricon-G,主要原因是图神经网络算法运算过程中的主要数据类型为顶点特征向量,其相关操作均为向量运算,较弱的并行运算性能使得 CPU 在处理图神经网络算法时性能很差。

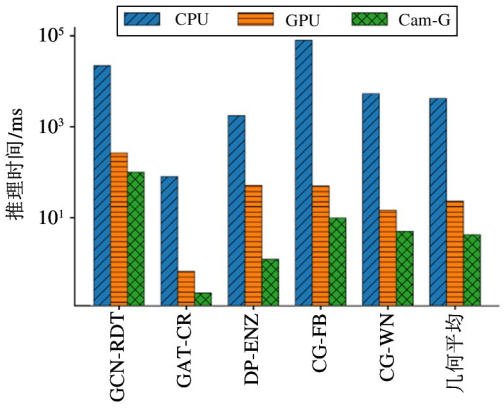

图 5　宏测试集性能测试结果

从图 5 来看,GPU 和 Cambricon-G 的性能表现较为接近。为了能更直观地对比 GPU 和 Cambricon-G 在处理图神经网络算法时的相对性能表现,本文以 CPU 的性能表现为基准,进一步计算和分析了其他两种设备相对于 CPU 的加速比,如图 6 所示。平均来看,GPU 相对于 CPU 实现了 181.1 倍的加速比,而 Cambricon-G 相对于 CPU 实现了 996.5 倍的加速比。其中,在 DiffPool-Enzymes 测试程序上,Cambricon-G 的性能达到 GPU 的 42.6 倍。而在宏测试集的其他 4 种测试程序上,2 种硬件设备的性能差距稳定在 2.6 ~ 5.2 倍。为了探究造成这一特殊情况的原因,本文进一步测试了 GPU 在运行宏测试集程序时的利用率。如图 7 所示,本文使用 NVIDIA 官方提供的 GPU 状态实时监测工具 nvidia-smi 抓取了实验过程中 GPU 能达到的利用率的最大值。可见,在执行 DiffPool-Enzymes 测试程序时,GPU 的最大利用率仅为 11%,远低于 GPU 在运行其他测试程序时的利用率。造成这一现象的原因是该测试程序是图分类任务,Enzymes 数据集是由 600

个规模很小的图结构组成,平均每个图仅包含 33 个顶点,并且每个图数据的顶点数和拓扑结构各不相同,因此 GPU 无法高效地进行批处理,频繁地启动核函数处理每个小图数据造成 GPU 利用率低,最终导致性能表现较差。而在其他测试程序中,GAT-Cora 测试程序的 GPU 利用率为 42%,低于其他 3 种测试程序 GCN-Reddit、CompGCN-FB15k237 和 CompGCN-WN18RR,原因在于其 Cora 数据集的规模较小,仅 2708 个顶点,同时 GAT 算法隐藏层通道数也较少,中间特征向量长度仅为8。两者共同导致 GAT-Cora 测试程序并行度不高,GPU 的运算单元无法被充分利用。

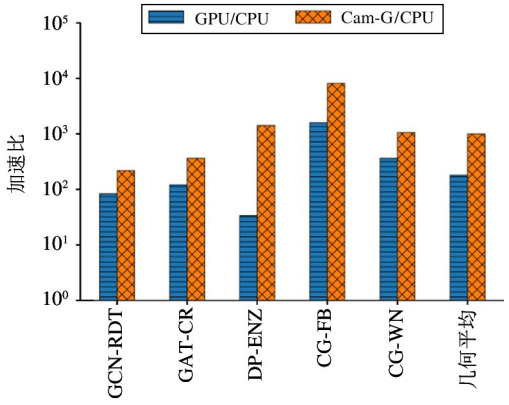

图 6　宏测试集加速比测试结果

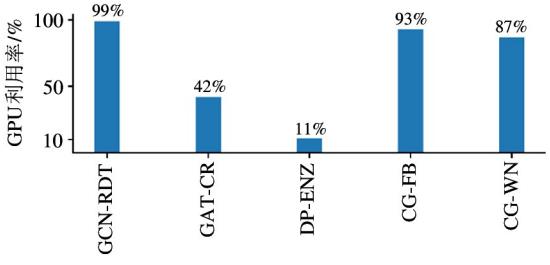

图 7　宏测试集 GPU 利用率

图 8 展示了 CPU、GPU 和 Cambricon-G 的性能功耗比,单位是 GFlops/W。总体来看,CPU 和 GPU 的性能功耗比平均仅为 0.014 GFlops/W 和 8.62 GFlops/W,而 Cambricon-G 的平均性能功耗比达到 56.6 GFlops/W,原因在于 Cambricon-G 设计了专门针对图神经网络算法的片上存储层次和访存优化方案。通过对图拓扑进行预处理,Cambricon-G 的片上缓存结构可以高效地进行顶点特征向量在缓存中的替换,使其缓存命中率大幅提高。因此,Cambricon-G 运行图神经网络算法时大幅降低了片外访存总量,提高了总体性能,同时也降低了访存功耗,因此具有较高的性能功耗比表现。

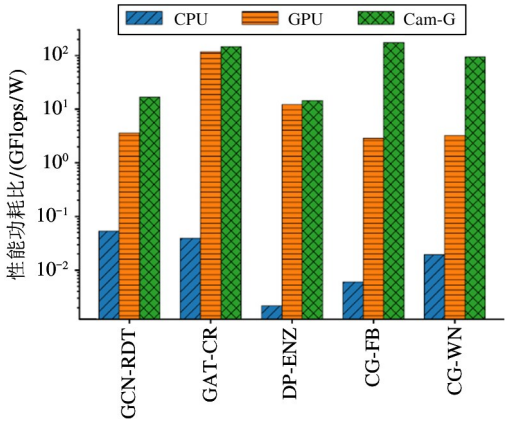

图 8　宏测试集性能功耗比测试结果

同时,本文使用上述 3 种运算设备对 BenchG-NN 的微测试集进行了实验测试。在微测试集中,由于 FRI 数据集顶点特征向量的体积达到 112 GB,远远超过当代 GPU 和各类图神经网络加速器的存储容量,现有 GPU 和加速器都无法支持与 FRI 数据集相关的测试,因此本文的后续实验和分析都不包含 FRI 数据集。事实上,由于数据集规模太大,包括学术界所提出的图神经网络加速器在内的大部分的现有运算设备都无法端到端地支持与 FRI 规模尺寸相近的数据集。然而,从大量图数据集规模特性的聚类结果(如图 4)来看,有相当数量的数据集具有比 FRI 更大的规模特性。这种超大规模图数据的部署和加速优化问题是当前图神经网络加速运算的空白领域,有待研究人员针对这类超大规模图处理任务设计专门的硬件加速器,或者设计专门处理超大规模图神经网络任务的分布式运算系统。

图 9 为微测试集矩阵乘法操作的性能测试结果。在处理较大规模的图数据集 CS 和 AM 时,Cambricon-G 的性能表现弱于 GPU,其原因是矩阵乘法操作具有运算量大、访存连续性强和数据复用规则清晰等特点,适合 GPU 这种规整的并行处理器。因此 GPU 的性能表现优于 Cambricon-G。而在处理 Enzymes 数据集时,由于每个图数据规模较小且顶点数量不同,导致 GPU 无法高效地进行批处

理,因此性能弱于 Cambricon-G。本文使用 nvprof 工具对微测试集运算过程中的关键硬件指标进行监测。图 10 和图 11 分别展示了 GPU 在运行微测试集的矩阵乘法操作时的运算单元利用率和实际片外访存带宽。可以发现,对于 CS 和 AM 这 2 个数据集,GPU 可以保持不低于 25% 的运算单元利用率和 49 GB/s 以上的实际访存带宽。对比之下,以 Enzymes 为代表的小图数据集则只能实现不到 4% 的运算单元利用率和不到 4 GB/s 的实际访存带宽。

图 12 为微测试集随机向量规约操作的性能测试结果。随机向量规约操作需要根据图拓扑连接关系进行大量的随机访存操作,因此访存连续度较低。而 GPU 使用高带宽的 HBM2 片外存储适合对向量或矩阵进行连续访存。而 Cambricon-G 针对图神经网络的这种随机访存模式进行了优化设计,因而其性能表现优于 GPU。根据如图 13 和图 14 所示的运算单元利用率和实际访存带宽监测结果,GPU 在执行随机向量规约操作时在全部数据集上只能实现最多 1.02% 的运算单元利用率和不超过 40 GB/s 的实际访存带宽,远低于 GPU 在执行矩阵乘法操作时的相应指标。进一步分析可以发现,GPU 在小图数据

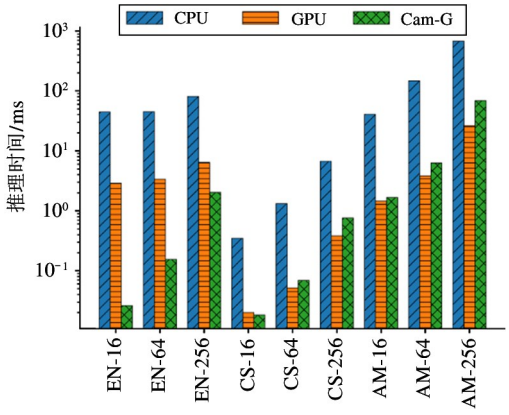

图 9　微测试集矩阵乘法操作的性能测试结果

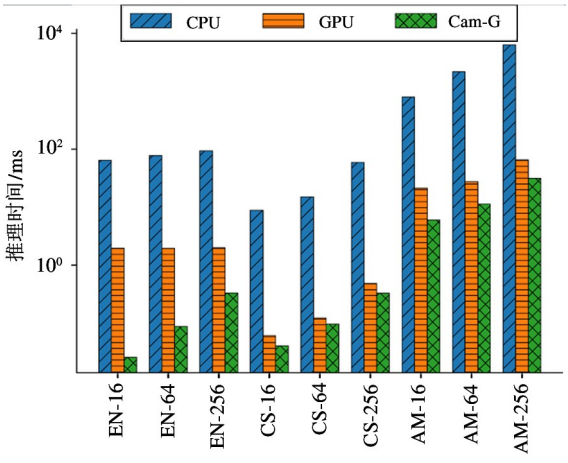

图 12　微测试集随机向量规约操作的性能测试结果

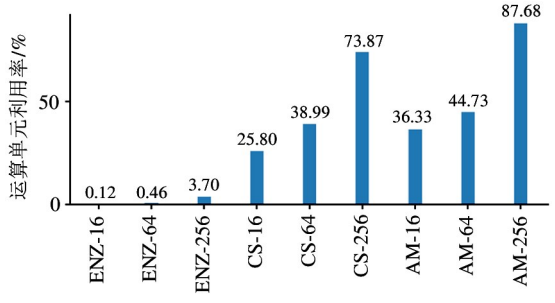

图 10　矩阵乘法操作的 GPU 运算单元率

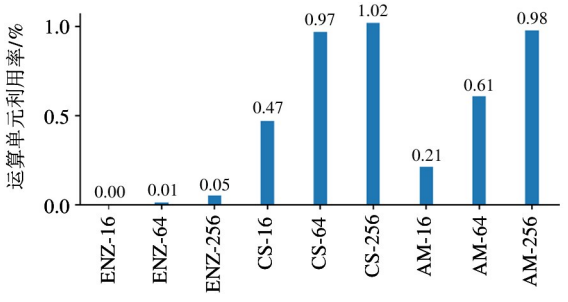

图 13　随机向量规约操作的 GPU 运算单元率

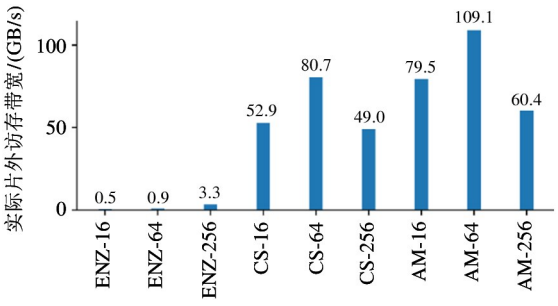

图 11　矩阵乘法操作的 GPU 实际片外访存带宽

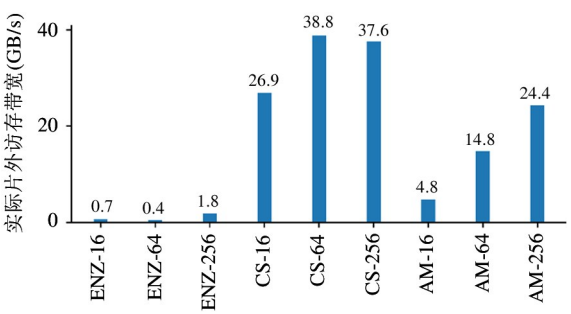

图 14　随机向量规约操作的 GPU 实际片外访存带宽

集 Enzymes 上的性能表现也远不如 Cambricon-G。原因是在 Enzymes 数据集上执行随机向量规约时，GPU 只能实现不超过 0.05% 的运算单元利用率和不到 2 GB/s 的实际访存带宽。

根据上述实验结果可以得出以下结论，高效处理图神经网络算法需要硬件设备具有较高的并行度，而以 GPU 为代表的通用并行处理器由于无法高效处理图神经网络算法的随机访存问题，减弱了其性能功耗表现。因此，针对图神经网络算法设计专用的硬件加速器成为必不可缺的技术路线和重要研究方向。本文所提出的 BenchGNN 在多种任务类型、应用领域、微观操作类型和数据集规模特性等多种场景对图神经网络运算设备进行评估，可以作为学术界针对图神经网络专用硬件加速器研究的设计目标和评价标准。

4 结 论

针对现有图神经网络硬件加速器研究缺乏统一的标准测试集的问题，本文提出一种针对图神经网络硬件加速器性能评估的标准测试集 BenchGNN。BenchGNN 包括用于整体性能评估的宏测试集和用于性能表现优劣势分析的微测试集。BenchGNN 的宏测试集包含图神经网络算法的 3 种任务类型和 5 种应用领域，微测试集包含 2 种主要操作类型和不同量化特性的图数据集。本文还在现有设备 CPU、GPU 和图神经网络专用加速器上对 BenchGNN 进行了实验测试，实验结果表明 BenchGNN 可以展示出不同设备在处理图神经网络运算时的性能和功耗表现。同时，结合微测试集的实验结果，BenchGNN 可以对后续设计新的图神经网络加速器提出有价值的优化建议。

参考文献

[1] YING R, HE R, CHEN K, et al. Graph convolutional neural networks for web-scale recommender systems[C]// Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, London, UK, 2018: 974-983

[2] YING Z, YOU J, MORRIS C, et al. Hierarchical graph representation learning with differentiable pooling[C]// Proceedings of the 32nd International Conference on Neural Information Processing Systems, Montreal, Canada, 2018: 4805-4815

[3] 汪琳琳, 施俊, 韩振奇, 等. 结合卷积神经网络与图卷积网络的乳腺癌病理图像分类研究[J]. 北京生物医学工程, 2021, 40(2): 130-138

[4] HAMILTON W L, YING R, LESKOVEC J. Inductive representation learning on large graphs[C]// Proceedings of the 31st International Conference on Neural Information Processing Systems, New York, USA, 2017: 1025-1035

[5] KIPF T, WELLING M. Semi-supervised classification with graph convolutional networks[C]// Proceedings of the 5th International Conference on Learning Representations, Toulon, France, 2017: 1-14

[6] VASHISHTH S, SANYAL S, NITIN V, et al. Composition-based multi-relational graph convolutional networks[C]// Proceedings of the 8th International Conference on Learning Representations, Addis Ababa, Ethiopia, 2019:1-16

[7] 许佳辉, 王敬昌, 陈岭, 等. 基于图神经网络的地表水水质预测模型[J]. 浙江大学学报(工学版),2021, 55(4):601-607

[8] 曹万平, 周刚, 陈黎, 等. 基于会话的图卷积递归神经网络推荐模型[J]. 四川大学学报(自然科学版), 2021,58(2):66-72

[9] 车向北, 康文倩 ,邓彬 ,等. 一种基于图神经网络的 SDN 路由性能预测模型[J]. 电子学报, 2021,49(3): 484-491

[10] GENG T, LI A, SHI R, et al. AWB-GCN: a graph convolutional network accelerator with runtime workload rebalancing[C]// The 53rd Annual IEEE/ACM International Symposium on Microarchitecture, Athens, Greece, 2020: 922-936

[11] LIANG S W, WANG Y, LIU C, et al. EnGN: a high-throughput and energy-efficient accelerator for large graph neural networks[J]. *IEEE Transactions on Computers*, 2021,70(9):1511-1525

[12] YAN M, DENG L, HU X, et al. HyGCN: a GCN accelerator with hybrid architecture[C]//2020 IEEE International Symposium on High Performance Computer Architecture, Washington, USA, 2020: 15-29

[13] TIAN C, MA L, YANG Z, et al. PCGCN: partition-centric processing for accelerating graph convolutional network[C]//2020 IEEE International Parallel and Distributed Processing Symposium, New Orleans, USA, 2020:

936-945

[14] AUTEN A, TOMEI M, KUMAR R. Hardware acceleration of graph neural networks[C]//The 57th ACM/IEEE Design Automation Conference, San Francisco, USA, 2020: 1-6

[15] KININGHAM K, LEVIS P, RE C. GRIP: a graph neural network accelerator architecture[J]. *IEEE Transactions on Computers*, doi:10.1109/TC. 2022.3197083

[16] ZHANG B, ZENG H, PRASANNA V. Hardware acceleration of large scale GCN inference[C]//2020 IEEE 31st International Conference on Application-specific Systems, Architectures and Processors, Manchester, UK, 2020: 61-68

[17] ZENG H, PRASANNA V. Graphact: accelerating GCN training on CPU-FPGA heterogeneous platforms[C]// Proceedings of the 2020 ACM/SIGDA International Symposium on Field-Programmable Gate Arrays, Seaside, USA, 2020: 255-265

[18] HUANG G, DAI G, YU W, et al. GE-SpMM: general-purpose sparse matrix-matrix multiplication on GPUs for graph neural networks[C]// Proceedings of the International Conference for High Performance Computing, Networking, Storage, and Analysis, Atlanta, USA, 2020: 1-12

[19] SONG X K, ZHI T, FAN Z, et al. Cambricon-G: a polyvalent energy-efficient accelerator for dynamic graph neural networks[J]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2021(99): 1-15

[20] MATTSON P, REDDI V, CHENG C, el al. MLPerf: an industry standard benchmark suite for machine learning performance[J] *IEEE Micro*, 2020, 40(2):8-16

[21] HU W H, FEY M, ZITNIK M, et al. Open graph benchmark: datasets for machine learning on graphs[C]//The 34th Conference on Neural Information Processing Systems, Vancouver, Canada, 2020: 1-34

[22] DWIVEDI V, JOSHI C, LAURENT T, et al. Benchmarking graph neural networks[EB/OL]. https://arXiv.org/pdf/2003.00982.pdf: arXiv, (2020-03-02), [2021-03-05]

[23] VELICKOVIC P, CUCURULL G, CASANOVA A, et al. Graph attention networks[C]//Proceedings of the 5th International Conference on Learning Representations, Vancouver, Canada, 2018: 1-12

Benchmarking graph neural network accelerators

SONG Xinkai[*,**,***], ZHI Tian[*,***], KONG Weihao[*,**,***], DU Zidong[*,***]

([*] State Key Laboratory of Computer Architecture, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190)

([**] University of Chinese Academy of Sciences, Beijing 100049)

([***] Cambricon Technologies, Beijing 100191)

Abstract

Graph neural network(GNN) has achieved breakthroughs in processing graph-structured data. However, researches on GNN accelerator design lack a clear design objective and unified evaluation methods. The Benchmark for graph neural network (BenchGNN) is proposed for evaluating the performance of GNN accelerators. BenchGNN consists of macro-benchmark and micro-benchmark. Macro-benchmark consists of algorithms of three task types of GNN and datasets from five application fields of GNN. Micro-benchmark consists of two basic micro-operation of GNN and four graph datasets of different scale characteristics. An experimental evaluation of BenchGNN is conducted on modern centrol processing unit (CPU), graphic processing unit (GPU), and a GNN accelerator. The experimental results show that the CPU cannot process GNN efficiently due to the lack of parallel processing units. The specifically designed accelerator achieves better performance and lower energy consumption than GPU, due to the fact that the design of the accelerator optimizes the random memory access of GNN workloads. The results inspire researchers of GNN accelerators that the design of GNN accelerators should take into account both the high parallelism of processors and the ability of performing random memory access.

Key words: graph neural network(GNN), accelerator, Benchmark