Long range arena:高效transformer的基准

Yi, Mostafa, Samira Abnar, Yikang Shen[1]，Dara Bahri[1]，Philip Pham Jinfeng Rao，刘杨，Sebastian Ruder, Donald Metzler
[1]谷歌研究
[2]谷歌DeepMind
{yitay, dehghani} @google.com

摘要

变压器不能很好地扩展到较长的序列长度，主要是因为二次自关注复杂性。在最近几个月，广泛的高效，快速的变压器已经提出了解决这个问题，往往不是声称优于或可比的模型质量香草变压器模型。到目前为止，关于如何评估这类模型还没有一个完善的共识。此外，在广泛的任务和数据集上的基准测试不一致，使得很难评估许多模型之间的相对模型质量。本文提出了一个系统和统一的基准，即*远程竞技场*，专门用于评估长期情景下的模型质量。所提出的基准是一套由1K到16K token序列组成的任务，包括广泛的数据类型和模态，如文本、自然、合成图像和需要相似性、结构和视觉-空间推理的数学表达式。我们在我们新提出的基准套件上系统地评估了十种完善的远程变压器模型(改革者，Linformers，线性变压器，下沉角变压器，表演者，合成器，稀疏变压器和长变形器)。远程竞技场为更好地理解这类高效的变压器模型铺平了道路，促进了这一方向的更多研究，并提出了新的具有挑战性的任务。我们的基准代码将在https://github.com/google-research/long-range-arena上发布。

1 介绍

变压器(Vaswani等人，2017)在许多方面都是最先进的，从语言(Devlin等人，2018;Raffel等人，2019;Child et al.， 2019)到图像(Tan & Bansal, 2019;Lu et al.， 2019)到蛋白质序列(Rives et al.， 2019)。变压器的一个共同弱点是它们在自关注机制中的二次内存复杂性，这限制了它们在需要更长的序列长度的领域的潜在应用。迄今为止，已经提出了令人眼花缭乱的高效变压器模型(“xformers”)来解决这个问题(刘等人，2018;Kitaev等，2020;Wang等，2020;Tay等，2020b;Katharopoulos等人，2020)。这些模型中的许多表现出了与普通Transformer模型相当的性能，同时成功地降低了自注意力机制的记忆复杂性。这个研究领域的概述可以在(Tay等人，2020c)中找到。

比较其中许多论文的评估和实验设置，我们可以做出以下观察。首先，对于高效变压器基准测试的可接受测试平台，没有统一的共识。所采用的任务类型也有很大的多样性——每个单一模型都是在不同的任务集和数据集上进行评估的，这使得不同模型的比较以及对其相对优势和弱点的评估变得困难。其次，用于评估的基准往往是任意选择的，没有过多考虑该任务是否适合评估长程建模。第三，许多论文倾向于将归纳偏差的有效性与预训练的好处混为一谈(Ainslie et al.， 2020;

[*] First two authors contributed equally.

Zaheer等人，2020;Wang等人，2020)，这往往会混淆架构的真正价值。预训练本身是一项计算成本高昂的努力，将归纳偏差研究与预训练解耦将使xformer研究更容易获得。

在本文中，我们提出了一个新的基准，*远程竞技场*(LRA)，用于长上下文场景下序列模型的基准测试。我们设计了一个由合成探测任务和现实世界任务组成的基准测试套件，并为最近提出的十种高效变压器模型提供了相对比较，包括稀疏变压器(Child等人)，

2019)， Reformer (Kitaev等人，2020)，Linformer (Wang等人，2020)，Longformer (Beltagy等人，2019)，2020)、下沉角变压器(Tay等人，2020b)、表演者(Choromanski等人，2020b)、syn- sizer (Tay等人，2020a)、线性变压器(Katharopoulos等人，2020)和BigBird (Zaheer等人，2020)。这是对这类模型进行的最全面、最广泛的并排评估。

虽然该基准测试的重点是这些体系结构在长上下文场景中推理的能力，但我们也对了解这些xformer体系结构在暴露于不同类型的数据和条件时的能力和属性感兴趣。因此，我们的基准被有意设计为能力探索，即我们选择具有某些固有结构的数据集和任务。例如，这些架构能否对本质上具有层次性或包含某种形式的空间结构的长序列进行建模?一般来说，我们对这些xformer模型在不同情况下的相对性能特别感兴趣。我们希望更好地理解这些，将在未来激发对更高效架构的研究。虽然本文的重点是高效的Transformer模型，但我们的基准也是模型无关的，也可以作为长程序列建模的基准。

除了比较这些模型的质量，我们还对这些模型进行了广泛的效率和内存使用分析。我们相信这样一个并排的性能基准将对社区有价值，为这些方法的实际效率提供更深入的见解。总的来说，本文提出了一个统一的框架，以方便地对高效的Transformer模型和广义上讲的长程序列模型进行并排比较。我们开源的框架是用JAX/FLAX[1]编写的。

2 远程竞技场(LRA)

介绍远程竞技场(remote Arena, *LRA)基准测试*。我们在Python 3和Jax/Flax中实现了我们的基准(包括任务、评估器和模型)，并开源了我们的代码2，使其易于扩展和在我们的工作之上构建。

2.1 站

为了创建远程竞技场基准，我们建立了一组所需数据:

1. 通用性:所有高效的transformer模型都应该适用于我们的任务。例如，考虑到并非所有的xformer模型都能够执行自回归解码(Wang et al.， 2020)，我们包括了只需要编码的任务。

2. 简单性:任务的设置应该很简单。所有使比较困难的因素都应该被移除。这鼓励使用简单的模型，而不是繁琐的管道方法。例如，我们避免包括任何特定的数据增强，并认为预训练超出了此基准的范围。

3.具有挑战性:对于当前的模型来说，任务应该足够困难，以确保有改进的空间，以鼓励未来在这个方向上的研究。
4. 长输入:输入序列长度应该相当长，因为评估不同的模型如何捕获远程依赖关系是LRA的核心焦点。
5. 探索不同方面:任务集应该评估模型的不同能力，如它们建模关系和层次/空间结构的能力，泛化能力等。

[1] https://github.com/google/flax

[2] https://github.com/google-research/long-range-arena

6. 非资源密集型和可访问性:基准应该被故意设计为轻量级，以便没有工业级计算资源的研究人员可以访问。

2.2 任务

介绍LRA基准测试的任务。请注意，这些任务是专门为评估高效Transformer模型的不同方面而设计的。关于每个任务的进一步详细信息可以在附录中找到。

2.2.1 长LISTOPS

在这项任务中，我们对长上下文场景中建模层次结构数据的能力感兴趣。这个基准任务是(Nangia & Bowman, 2018)中提出的标准ListOps任务的更长的变体，该任务旨在调查神经模型的解析能力。
数据集由具有层次结构的序列和由分隔符(括号)括起来的MAX、MEAN、MEDIAN和SUM MOD运算符组成。下面是一个(短得多)序列的例子:

INPUT: [MAX 4 3 [MIN 2 3] 1 0 [MEDIAN 1 5 8 9, 2]] **OUTPUT:** 5

在我们的任务中，我们使用序列长度高达2K的ListOps版本来测试在处理长上下文时分层推理的能力。自然，在上面的例子中，模型需要访问所有标记并对输入的逻辑结构进行建模，以便做出预测。该任务是一个十向分类任务，具有相当大的挑战性。

2.2.2 字节级别的文本分类

这个使用真实世界数据的任务代表了高效transformer的一个常见用例，在处理长文档时经常需要transformer。文本分类尤其与许多现实世界的应用有关，如垃圾邮件、欺诈、机器人检测和商业文档分类等(Howard & Ruder, 2018)。
这项任务还对模型处理组合性的能力进行基准测试，因为它需要将字符组合成单词，以进入更高级别的短语。与ListOps相比，边界的定义不太好，需要从数据中学习，这本身就是一个具有挑战性的问题(Kawakami等人，2019)。

我们考虑了这个任务的字节/字符级别的设置，以便模拟更长的输入序列，这也使任务具有相当大的挑战性。[3]请注意，这种设置与字符级语言建模(char LM)有显著不同。在char LM中，阅读附近的上下文就足以确定下一个字符，例如，模型很可能在看到前缀` *appl* `*后预测* `*e* `。对于字节级别的文本分类，模型需要对组合的、未分割的数据进行推理，以便解决一个有意义的现实世界任务。我们使用IMDb reviews (Maas et al.， 2011)数据集，这是一个常用的数据集来对文档分类进行基准测试。我们为这项任务使用固定的最大长度4K，在必要时进行截断或填充。这是一个以准确度为度量标准的二分类任务。

2.2.3 字节级文档检索

我们进一步评估模型编码和存储对匹配和检索有用的压缩表示的能力。学习两个向量之间的相似性得分是机器学习中的一个常见问题，并且对广泛的应用很有用(Guo et al.， 2016)。

因此，这个任务主要是在“双塔设置”中对两个文档之间的相似度分数进行建模，在这种设置中，压缩表示被连接并传递到线性分类器中。请注意，我们故意防止模型使用交叉注意力。因此，这项任务是为了测试如何使用交叉注意力

[3] On the IMDb word-level task, models without pre-training achieve accuracies in the high 80s while the same models score in the mid 60s on the character-level task (Tay et al., 2020b).

好的模型能够将长序列*压缩*为适合基于相似性匹配的表示。

我们使用ACL Anthology网络(AAN;Radev et al.， 2013)数据集，用于识别两篇论文是否有引文链接，这是长形式文档匹配中常用的设置(Jiang et al.， 2019;Yang et al.， 2020)。

与文本分类设置类似，我们使用字节/字符级别的设置，这对模型在较长的上下文上组合和聚合信息提出了挑战。我们为每个文档使用了4K的序列长度，这使得这项任务的总文本长度为8K。这是一个以准确率为指标的二分类任务。

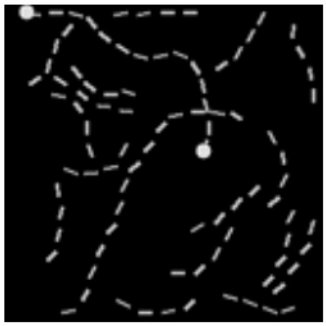
2.2.4 像素序列上的图像分类

这个任务是一个图像分类任务，其中输入是像素序列。换句话说，一个N ×N的图像被平展为一个长度为N^2像素的序列。
与之前的任务要求捕获数据中的层次结构类似，此任务要求模型学习输入像素之间的2D空间关系，同时以1D符号序列的形式呈现。

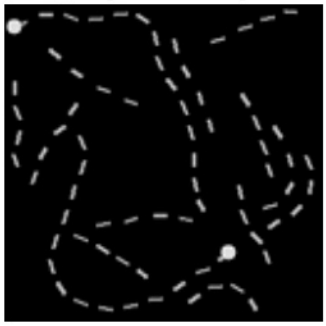
我们专注于评估旨在处理离散符号序列的Transformer模型，因此不允许嵌入像素级输入的CNN stem等额外模块。为了简化设置，我们将输入图像映射到单个灰度通道，其中每个像素用8位像素强度表示(词汇量大小为256)。在LRA中，我们使用CIFAR-10数据集(Krizhevsky, 2009)进行图像分类任务。

2.2.5 pathfinder(远程空间依赖)

探路者挑战(Linsley等人，2018;Kim* et al.， 2020)最初是为了学习长程空间依赖性而引入的。这是一种由认知心理学激发的合成视觉任务(Houtkamp & Roelfsema, 2010)。
该任务需要一个模型来做出一个二元决策，即表示为圆圈的两个点是否由由破折号组成的路径连接。我们在图1中展示了两个连通点的正例和两个不连通点的反例。数据集还包含干扰路径，这使得这个设置具有挑战性。我们通过将图像视为像素序列来对这项任务进行建模。在这个任务中，图像具有维度(32 ×32)，这些维度组成了1024个序列长度。



(a) A positive example.



(b) A negative example.

图1:探路者任务的示例。

2.2.6 pathfinder-x(具有极端长度的长程空间依赖关系)

最后，我们考虑一个极端版本的探路者(探路者- x)，其中的例子由16K像素组成(即128像素的图像×128)。

这里的关键目标是观察模型是否无法解决16K极限版本，即使它可以成功学习1024个令牌的标准版本。这是一个有趣的试金石测试，看看当序列长度长得多时，相同的算法挑战是否承担了不同程度的难度。我们将其作为Path-X纳入我们的基准。

2.3 需要LRA任务的注意广度

LRA基准的主要目标之一是评估不同的高效变压器模型捕获远程依赖关系的能力。设计任务和设置时就考虑到了这一目标。为了对注意力机制编码输入所需考虑的空间范围进行定量估计，我们定义了*所需的注意力广度*。
给定一个经过训练的基于注意力的模型和一个token序列作为输入，注意力模块所需的注意力广度被计算为查询token和参与token之间的平均距离，按注意力权重进行缩放。在这里，我们在我们最好的vanilla Transformer模型中为每个任务计算所有注意力模块*所需*的平均注意力*广度*，在验证集的1K随机样本上平均。

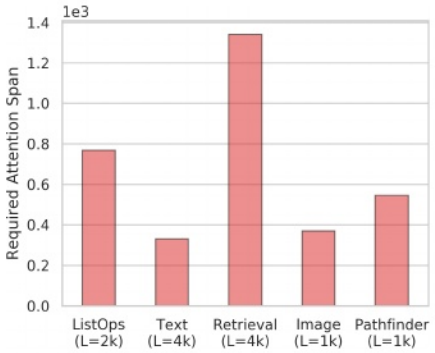


图2:不同任务上所需的注意力广度。

图2总结了LRA中每个任务所需的注意广度。对于LRA中的所有任务，所需的注意广度都相当高。这表明，Transformer模型需要超越仅结合局部信息，而在许多其他任务和数据集中，注意力机制大多需要结合邻近位置的信息。
考虑到LRA的目的，我们发现对于基于transformer的模型来说，*所需的注意力广度*可以很好地代表任务的难度

3 实验结果

3.1 模型

本节描述我们在LRA基准上评估的模型。我们的评估基于最近提出的10个有效的Transformer模型。除了标准的vanilla变压器(Vaswani等人，2017)和简单的局部注意力基线外，我们还比较了稀疏变压器(Child等人，2019)、Longformers (Beltagy等人，2020)、Linformers (Wang等人，2020)、Reformers (Kitaev等人，2020)、Sinkhorn变压器(Tay等人，2020b)、合成器(Tay等人，2020a)、BigBird (Zaheer等人，2020)、线性变压器(Katharopoulos等人，2020)和Per-formers (Choromanski等人，2020a)。
相信这10个模型代表了最近高效Transformer模型的不同横截面。

3.2 基准背后的哲学

我们注意到，要对所有模型进行完全公平的评估是不平凡的，几乎是不可能的。巨大的搜索空间激励我们遵循一组固定的超参数(数量

[4] Note that this metric mainly provides an indication of the required attention span for a task and the relative differences between tasks based on a reasonably strong model; a better model might only need to attend to shorter ranges (Daniluk et al., 2017; Rae & Razavi, 2020).

Model	ListOps	Text	Retrieval	Image	Pathfinder	Path-X	Avg
Transformer	36.37	64.27	57.46	42.44	71.40	FAIL	54.39
Local Attention	15.82	52.98	53.39	41.46	66.63	FAIL	46.06
Sparse Trans.	17.07	63.58	**59.59**	**44.24**	71.71	FAIL	51.24
Longformer	35.63	62.85	56.89	42.22	69.71	FAIL	53.46
Linformer	35.70	53.94	52.27	38.56	76.34	FAIL	51.36
Reformer	**37.27**	56.10	53.40	38.07	68.50	FAIL	50.67
Sinkhorn Trans.	33.67	61.20	53.83	41.23	67.45	FAIL	51.39
Synthesizer	36.99	61.68	54.67	41.61	69.45	FAIL	52.88
BigBird	36.05	64.02	59.29	40.83	74.87	FAIL	**55.01**
Linear Trans.	16.13	**65.90**	53.09	42.34	75.30	FAIL	50.55
Performer	18.01	65.40	53.82	42.77	**77.05**	FAIL	51.41
Task Avg (Std)	29 (9.7)	61 (4.6)	55 (2.6)	41 (1.8)	72 (3.7)	FAIL	52 (2.4)

表1:在远程Arena基准上的实验结果。最好的模型用黑体字表示，第二好的模型用下划线表示。与Pathfinder任务相反，所有模型在Path-X任务上都没有学到任何东西，用FAIL表示。这表明增加序列长度可能会给模型训练造成严重困难。我们把Path-X留在这个基准上，以供未来的挑战者使用，但不包括在平均分数中，因为它对相对性能没有影响。

层，头，嵌入维度等)为所有模型。如果我们积极调整所有模型的超参数，模型的最佳性能和相对顺序可能会发生变化。因此，本文提供的结果并不意味着是xformer最好的最终权威文档。相反，我们为未来的研究提供了一个起点，并努力做到尽可能的公平。为了做到这一点，我们计划发布包含所有超参数和实现细节的代码。此外，我们打算让我们的论文成为一个活文档，并鼓励研究人员(作者和更广泛的社区)贡献和继续更新这篇论文(规则和限制在附录中描述)。我们还尽我们所能地实现了所有模型。我们经常咨询被纳入模型的原始开发人员。

3.3 定量结果

基于我们的结果，我们观察到(1)LRA中所有提出的任务都具有相当大的挑战性，(2)不同xformer模型之间的模型性能存在显著差异。

ListOps任务(10-way classification)被证明是相当困难的，最好的模型只获得37%。与随机机会的巨大差距表明，模型确实在学习任务。我们注意到，xformer模型的归纳偏差在这个任务上发挥了实质性的作用，其中大约一半的xformer模型能够获得> 30%的性能，而其余的模型仅获得略高于随机机会。这可能意味着，某些受效率启发的归纳偏差可能比其他偏差更擅长处理分层数据。例如，我们的实验结果似乎表明，基于核的模型(例如，Performer，线性transformer)在分层结构的数据上可能不那么有效。

文本分类的结果字节级分类被证明是困难的和具有挑战性的，特别是在没有使用预训练或上下文嵌入的情况下。最好的模型只获得了65.90的准确率。线性Transformer在这项任务上表现良好，Performer模型也一样。与ListOps任务相反，基于内核的快速模型似乎在此任务上做得很好。

检索结果不同模型在这个任务上的分数也相当低(平均55%)，表明任务的难度。vanilla Transformer模型仅达到57.46%的准确率，一些xformer变体的得分非常接近随机。表现最好的模型是稀疏变压器，第二好的是BigBird。我们发现遵循固定稀疏模式的模型在这项任务上做得很好。基于低秩分解和核的模型表现相对较差。

Model	Steps per second				Peak Memory Usage (GB)			
	1K	2K	3K	4K	1K	2K	3K	4K
Transformer	8.1	4.9	2.3	1.4	0.85	2.65	5.51	9.48
Local Attention	9.2 (1.1x)	8.4 (1.7x)	7.4 (3.2x)	7.4 (5.3x)	0.42	0.76	1.06	1.37
Linformer	9.3 (1.2x)	9.1 (1.9x)	8.5 (3.7x)	7.7 (5.5x)	**0.37**	**0.55**	0.99	**0.99**
Reformer	4.4 (0.5x)	2.2 (0.4x)	1.5 (0.7x)	1.1 (0.8x)	0.48	0.99	1.53	2.28
Sinkhorn Trans	9.1 (1.1x)	7.9 (1.6x)	6.6 (2.9x)	5.3 (3.8x)	0.47	0.83	1.13	1.48
Synthesizer	8.7 (1.1x)	5.7 (1.2x)	6.6 (2.9x)	1.9 (1.4x)	0.65	1.98	4.09	6.99
BigBird	7.4 (0.9x)	3.9 (0.8x)	2.7 (1.2x)	1.5 (1.1x)	0.77	1.49	2.18	2.88
Linear Trans.	9.1 (1.1x)	9.3 (1.9x)	8.6 (3.7x)	7.8 (5.6x)	**0.37**	0.57	**0.80**	1.03
Performer	**9.5** (1.2x)	**9.4** (1.9x)	**8.7** (3.8x)	**8.0** (5.7x)	**0.37**	0.59	0.82	1.06

表2:所有模型的批处理大小一致为32的所有Xformer模型的基准结果。除了每秒的步骤外，我们还报告了与括号中vanilla Transformer相比的相对速度的增加/减少。内存使用率指的是每个设备在每个TPU设备上的内存使用率。基准测试在4x4 TPU V3芯片上运行。

图像分类的结果在图像分类任务中，大多数模型的表现相当相似(模型性能之间的低方差)。此任务的最佳模型是稀疏变压器，其次是Performer。Linformer和改革者在这项任务上表现不佳。在一个相关的注意点上，我们还观察到大多数模型在泛化到测试方面很困难，即使它们设法过拟合训练集。虽然我们在每个单一模型上广泛尝试了不同的正则化技术，但它们在训练集和测试集上的性能之间存在相当大的差距(更多细节见附录)。

在Pathfinder / Path-X上的结果表明，所有模型在Pathfinder任务上都达到了合理的性能。平均性能为72，最好的模型Performer获得了77.05%的准确率。在所有模型中，局部注意力模型的表现最差。
所有模型都未能解决Path-X任务，成功率最高为50%。我们发现这很有趣，因为这本质上是一个与标准探路者相同的任务，尽管具有更长的序列长度。因此，我们观察到，任务的极端长度可以显著阻碍模型学习任何有意义的东西。我们将Path-X留在我们的基准套件中，希望在极端长度的序列建模方面刺激未来的进展。

3.4 效率基准

在本节中，我们报告我们运行的效率指标。为了简单起见，我们使用字节级别的文本分类基准，并报告序列长度{1K, 2K, 3K, 4K}的运行时间和内存消耗。我们在所有运行中使用32个批量大小，并在4x4 TPU V3芯片上进行实验。我们强调，这在很大程度上还是取决于硬件和实现细节(更多细节可以在附录中找到)。

Results on Speed Table 2报告了我们在xformer模型上的效率基准。我们注意到，低秩模型和基于核的模型通常是最快的。总体上最快的模型是Performer模型(Choromanski等人，2020a)，在4k序列长度上比变形金刚快5.7倍。Linformer (Wang等人，2020)和线性变压器(Katharopoulos等人，2020)紧随其后，几乎和表演者一样快(快5.5到5.6倍)。根据我们的实现，最慢的模型是Reformer模型(Kitaev等人，2020)，它在4K序列长度下的速度约为vanilla Transformer的80%，在1K序列长度下的速度为其一半。

在我们的基准测试中，内存占用最小的模型是Linformer模型，每个TPU设备的内存占用为0.99GB，而在N = 4K的情况下，普通变形金刚的内存占用为9.48GB。这大约减少了10倍的内存占用。与速度相似，表演者和线性变压器也相对紧凑，几乎与线性变压器一样紧凑。其他模型(Local Attention, Reformers，大鸟，synthesizers)与香草变形金刚相比仍然不那么需要内存，但效率相对较低

(内存消耗方面)与线性变压器、表演者和线性变压器相比。我们还注意到，Linformer和Performer等模型的内存消耗扩展得非常好，在3K和4K时的内存使用大致相等。

3.5 总体结果:没有放之四海而皆准

根据我们的分析，大鸟模型在LRA得分方面的定性表现最好，即整合了所有五个任务。虽然与其他模型相比，大鸟在任何单独的任务上都做得不是很好，但它在所有任务上都有良好的表现。表演者和线性变压器在某些任务上表现很好，但他们的平均表现被ListOps任务降低了。图3显示了定性性能、模型速度和内存占用之间的权衡。虽然大鸟表现良好，但它的速度几乎与普通的变形金刚相似。另一方面，像Local Attention这样的模型是以较低的定量性能为代价的。在这些模型中，基于内核的变体，即Performer、Linformer和linear Transformer似乎能够在速度和性能方面做出更好的权衡，同时具有合理的内存使用。

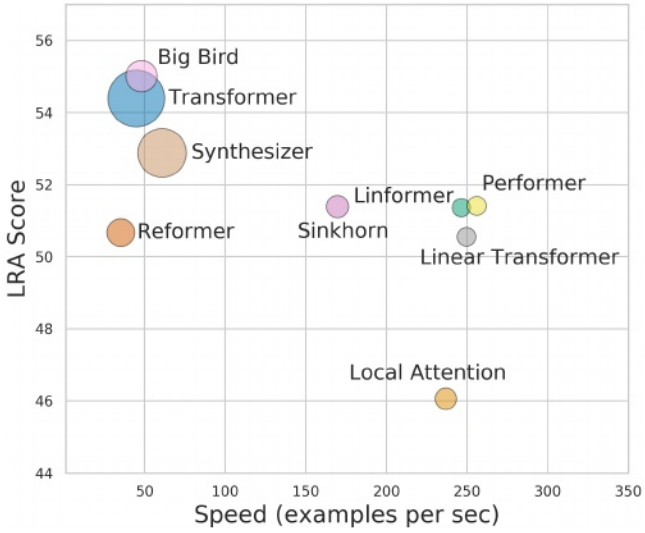


图3:不同模型的性能(y轴)、速度(x轴)和内存占用(圆的大小)。

4 相关工作

4.1 高效变压器

Transformer模型的普及，以及众所周知的内存密集型特点，刺激了这方面的大量创新。该领域的早期工作通常考虑固定模式(局部窗口)方法(Liu et al.， 2018;Parmar et al.， 2018)。最近提出了更先进的模型，包括组合模式(Child et al.， 2019;Ho et al.， 2019;Beltagy等，2020;Zaheer等人，2020)，学习模式(Kitaev等人，2020;Roy等人，2020)，以及最近基于核的模型(Katharopoulos等人，2020;Choromanski等人，2020a)或低秩近似(Wang等人，2020)。为简洁起见，我们建议感兴趣的读者参考(Tay et al.， 2020c)，以了解这一研究方向的详细综述。

4.2 现有基准

生成式建模/语言建模这种生成式建模任务需要预测下一个字符、单词或像素，并且是xformer评估的主要内容(Roy等人，2020;Kitaev等人，2020)。然而，关于此类任务实际上编码了多少远程信号一直存在争议(Rae & Razavi, 2020)。

经过注意力增强的LSTM语言模型很少会关注超过前面七个上下文单词的内容(Daniluk等人，2017)，并且已知LSTM语言模型的样本会迅速退化为通用文本。另一方面，最近的模型，如Transformer-XL (Dai等人，2019)，被观察到对大约900个token的上下文敏感，来自大规模模型的样本(Radford等人，2019)在更长的序列中保持一致的主题。然而，即使是这样的最新模型，也可以通过限制注意力范围来改进(Rae & Razavi, 2020)。总而言之，虽然标准语言建模数据集包含一些长程信号，需要执行长程共指解决、用事件进行推理、话语理解等(Ruder等人，2019)，但它似乎被更强的短期词共现信号所掩盖，因此难以评估[5]

另一个常用的评估任务是问题回答(QA;Zaheer等人，2020)。特别是开放域QA通常要求模型回答基于长上下文(如整个维基百科文档)的问题(Joshi等人，2017;Kwiatkowski et al.， 2019)甚至是书籍(Ko$^{\vee}$cisk´y et al.， 2018)。其他数据集被明确设计为需要多个“跳数”的推理(Welbl等人，2018;Yang et al.， 2018)。成功的方法往往是高度工程化的、计算成本昂贵的系统，需要预训练和单独的检索模型(Lee等人，2019;Guu et al.， 2020)。

自然语言理解/胶水任务对自然语言理解(NLU)任务的评估也很常见(Wang et al.， 2020)。大多数这些数据集中的示例，如MultiNLI (Williams et al.， 2018)和SST (Socher et al.， 2013)由单句和平均少于100个token组成。

5 结论

我们提出了远程竞技场(Long Range Arena, LRA)作为评价高效变压器研究进展的新基准。我们的新基准具有挑战性，并探索了模型处理不同数据类型和结构(如文本、数学和视觉数据)的能力。我们的基准由1K到16K token的任务组成。首次对最近提出的10个高效的Transformer模型进行了广泛的并排比较。实验结果表明，即使对于远程Transformer模型，这些任务也非常具有挑战性。整体结果表明，没有放之四海而皆准的解决方案，必须在模型质量和速度/内存方面做出权衡。我们计划开源我们的代码和基准，以促进未来的基准测试、研究和模型开发。

6 确认

我们要感谢以下同事:Krzysztof Choromanski, Richard Song, Tamas Sarlos对Performer设置的建议。David Dohan和Manzil Zaheer在BigBird实现上的帮助。Anselm Levskaya为改革者提供一些有用的参考代码。Orhan Firat提供有用的指针。唐嘉熙、Jai Gupta、秦真、郑彻、赵哲、胡安达成、Thomas Unterthiner、Marc Najork、Aurko Roy、Kevin Murphy、Ashish Vaswani、Niki Parmar、mohammad Taghi Saffar、Noah Fiedel和Peter J Liu，进行一般性反馈和讨论。我们也要感谢Drew Linsley，他为我们建立探路者基准提供了帮助和信息。

参考文献

Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020.

Joshua Ainslie, Santiago Ontanon, Chris Alberti, Philip Pham, Anirudh Ravula, and Sumit Sanghai. Etc: Encoding long and structured data in transformers. *arXiv preprint arXiv:2004.08483*, 2020.

[5] Datasets such as LAMBADA (Paperno et al., 2016) more explicitly test for context understanding but are still restricted to comparatively short contexts of five sentences on average.

Iz Beltagy, Matthew E Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Jared Davis, Tamas Sarlos, David Belanger, Lucy Colwell, and Adrian Weller. Masked language modeling for pro-teins via linearly scalable long-context transformers. *arXiv preprint arXiv:2006.03555*, 2020a.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020b.

Zihang Dai, Zhilin Yang, Yiming Yang, William W. Cohen, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context. In *Proceedings of ACL 2019*, 2019.

Micha☒ Daniluk, Tim Rockt, Johannes Welbl, and Sebastian Riedel. Frustratingly Short Attention Spans in Neural Language Modeling. In *Proceedings of ICLR 2017*, 2017.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pp. 55–64, 2016.

Kelvin Guu, Kenton Lee, Zora Tung, and Panupong Pasupat. REALM: Retrieval-Augmented Lan-guage Model Pre-Training. In *Proceedings of ICML 2020*, 2020.

Jonathan Ho, Nal Kalchbrenner, Dirk Weissenborn, and Tim Salimans. Axial attention in multidi-mensional transformers. *arXiv preprint arXiv:1912.12180*, 2019.

Sara Hooker. The hardware lottery. *arXiv preprint arXiv:2009.06489*, 2020.

R. Houtkamp and P. R. Roelfsema. Parallel and serial grouping of image elements in visual percep-tion. *J Exp Psychol Hum Percept Perform,*, 2010.

Jeremy Howard and Sebastian Ruder. Universal Language Model Fine-tuning for Text Classifica-tion. In *Proceedings of ACL 2018*, 2018.

Jyun-Yu Jiang, Mingyang Zhang, Cheng Li, Michael Bendersky, Nadav Golbandi, and Marc Najork. Semantic text matching for long-form documents. In *The World Wide Web Conference*, pp. 795– 806, 2019.

Mandar Joshi, Eunsol Choi, Daniel S Weld, Luke Zettlemoyer, and Paul G Allen. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of ACL 2017*, 2017.

Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and Franc¸ois Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. *arXiv preprint arXiv:2006.16236*, 2020.

Kazuya Kawakami, Chris Dyer, and Phil Blunsom. Learning to discover, ground and use words with segmental neural language models. In *Proceedings of ACL 2019*, pp. 6429–6441, 2019.

Junkyung Kim*, Drew Linsley*, Kalpit Thakkar, and Thomas Serre. Disentangling neural mecha-nisms for perceptual grouping. In *International Conference on Learning Representations*, 2020.

Nikita Kitaev, Lukasz Kaiser, and Anselm Levskaya. Reformer: The efficient transformer. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=rkgNKkHtvB.

Tom´a s˘Ko˘cisk´y, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, G´abor Melis, and Edward Grefenstette. The NarrativeQA Reading Comprehension Challenge. *Transactions of the Association for Computational Linguistics*, 2018.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin Kenton Lee, Kristina Toutanova, Llion Jones Matthew Kelcey, Ming-Wei Chang, Andrew M Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural Questions: a Benchmark for Question Answering Research. In *Transactions of the ACL*, 2019.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent Retrieval for Weakly Supervised Open Domain Question Answering. In *Proceedings of ACL 2019*, 2019.

Drew Linsley, Junkyung Kim, Vijay Veerabadran, Charles Windolf, and Thomas Serre. Learning long-range spatial dependencies with horizontal gated recurrent units. In *Advances in neural information processing systems*, pp. 152–164, 2018.

Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. Generating wikipedia by summarizing long sequences. *arXiv preprint arXiv:1801.10198*, 2018.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguis-tic representations for vision-and-language tasks. In *Advances in Neural Information Processing Systems*, pp. 13–23, 2019.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pp. 142–150, Portland, Oregon, USA, June 2011. Association for Computational Linguistics. URL http: //www.aclweb.org/anthology/P11-1015.

Nikita Nangia and Samuel R Bowman. Listops: A diagnostic dataset for latent tree learning. *arXiv preprint arXiv:1804.06028*, 2018.

Denis Paperno, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fern. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of ACL 2016*, 2016.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, ☒ukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. *arXiv preprint arXiv:1802.05751*, 2018.

Dragomir R Radev, Pradeep Muthukrishnan, Vahed Qazvinian, and Amjad Abu-Jbara. The acl anthology network corpus. *Language Resources and Evaluation*, 47(4):919–944, 2013.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language Models are Unsupervised Multitask Learners. 2019.

Jack W Rae and Ali Razavi. Do Transformers Need Deep Long-Range Memory? In *Proceedings of ACL 2020*, pp. 7524–7529, 2020.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019.

Alexander Rives, Siddharth Goyal, Joshua Meier, Demi Guo, Myle Ott, C Lawrence Zitnick, Jerry Ma, and Rob Fergus. Biological structure and function emerge from scaling unsupervised learning to 250 million protein sequences. *bioRxiv*, pp. 622803, 2019.

Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. Efficient content-based sparse attention with routing transformers. *arXiv preprint arXiv:2003.05997*, 2020.

Sebastian Ruder, Matthew E Peters, Swabha Swayamdipta, and Thomas Wolf. Transfer learning in natural language processing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Tutorials*, pp. 15–18, 2019.

Richard Socher, Alex Perelygin, Jean Y Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of EMNLP 2013*, pp. 1631–1642. Citeseer, 2013.

Hao Tan and Mohit Bansal. LXMERT: Learning Cross-Modality Encoder Representations from Transformers. In *Proceedings of EMNLP 2019*, 2019.

Yi Tay, Dara Bahri, Donald Metzler, Da-Cheng Juan, Zhe Zhao, and Che Zheng. Synthesizer: Rethinking self-attention in transformer models. *arXiv preprint arXiv:2005.00743*, 2020a.

Yi Tay, Dara Bahri, Liu Yang, Donald Metzler, and Da-Cheng Juan. Sparse sinkhorn attention. *arXiv preprint arXiv:2002.11296*, 2020b.

Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *arXiv preprint arXiv:2009.06732*, 2020c.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, ☒ ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. Constructing Datasets for Multi-hop Read-ing Comprehension Across Documents. In *Transactions of the Association for Computational Linguistics*, 2018.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. A Broad-Coverage Challenge Corpus for Sentence Understanding through Inference. In *Proceedings of NAACL-HLT 2018*, 2018. URL http://arxiv.org/abs/1704.05426.

Liu Yang, Mingyang Zhang, Cheng Li, Michael Bendersky, and Marc Najork. Beyond 512 tokens: Siamese multi-depth transformer-based hierarchical encoder for document matching. *CoRR*, abs/2004.12297, 2020. URL https://arxiv.org/abs/2004.12297.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In *Proceedings of EMNLP 2018*, 2018.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. Big bird: Transformers for longer sequences. *arXiv preprint arXiv:2007.14062*, 2020.

一个附录

A.1 LRA任务

本节介绍每个任务的详细信息和超参数。我们还计划在发布模型和基准实现的同时发布配置文件，这些文件可以用来重现论文中报告的结果。

A.1.1 LISTOPS

按照生成步骤(Nangia & Bowman, 2018)，我们生成了我们自己的这个任务的长版本。我们为这个任务使用了2k的序列长度。我们所有的xformer模型都有512个嵌入维度，8个头，6层和2048个前馈维度。我们为5K步训练所有模型。使用[CLS] token并映射到10类Softmax层进行分类。

A.1.2 字节级文档分类

我们使用IMDb评论数据集(Maas et al.， 2011)和一个序列长度为{1K, 2K, 3K, 4K}的token用于所有模型。我们在这四个序列长度中选择了最好的结果。我们使用[cls] token进行预测。来自xformer编码器的所有[cls]令牌都被传递到具有ReLU激活的两层MLP中。MLP发出一个用于二元分类的2类逻辑。我们优化了softmax交叉熵损失函数。所有xformer模型都使用相同数量的层、头和隐藏维度进行参数化，即8个头、512个隐藏维度和位置FFN层的d = 2048。我们为所有的xformer使用6层。学习率为0.05，权重衰减为0.1。我们使用Adam与warm - up。所有模型都是为20K步和32个批次大小进行训练的。

A.1.3 字节级的文档匹配

我们使用ACL anthology网络进行相关文章匹配任务。我们使用每个文档的序列长度为4K(两个序列总共8K个token)。两个编码器共享参数。与文档分类类似，我们使用来自xformer编码器的[cls] token。设X_1为来自文档1的[cls] token嵌入，X_2为来自文档2的[cls] token嵌入，通过以下方式计算最终得分:

$$Y = \text{MLP}([X_1, X_2, X_1 * X_2, X_1 - X_2]) \tag{1}$$

其中，MLP(.)是具有relu激活函数的两层MLP。代替更长的序列长度，我们使用批大小为32，嵌入维度为128,4个头，FFN维度为512和4层。模型用Adam训练了5K步，学习率为0.5。

A.2 图像分类

我们使用灰度(单通道)CIFAR10作为图像分类数据集，有10个类。输入图像的分辨率为32 ×32，在将输入图像展平后，我们用1024像素的序列输入我们的xformer编码器。与我们的其他分类任务类似，在xformer编码器的顶部有一个分类器头，由具有ReLU激活的两层MLP组成。Softmax交叉熵被用于优化模型的参数。我们训练了200个epoch的模型，并对不同的超参数进行了广泛的扫描，发现以下值可以在所有xformers中获得最佳的平均性能:3层，4个头，128作为FFN块的隐藏维度，64作为查询/键/值的隐藏维度，最终学习率为0.01。

A.2.1 泛化的差距

对于图像分类基准，在第3节中，我们提到大多数模型都很难泛化到测试集。表3展示了不同模型的训练和测试精度，对于几乎所有这些模型，两个分数之间的差距相当高。

虽然这个任务对于传统模型来说很容易解决(例如，在没有数据增强的情况下，灰度CIFAR10上的宽网精度为89.21)，但对于基于transformer的模型来说却相当困难

Model	test accuracy	train accuracy
Transformer	42.44	69.45
Local Attention	41.46	63.19
Sparse Trans.	**44.24**	66.74
Longformer	42.22	71.65
Linformer	38.56	97.23
Reformer	38.07	68.45
Sinkhorn Trans.	41.23	69.21
Synthesizer	41.61	**97.31**
BigBird	40.83	71.49
Linear Trans.	42.34	65.61
Performer	42.77	73.90

表3:在图像分类任务上测试和训练不同模型的精度。

使用这种设置的模型。自然，人们可以通过不同的设置找到提高性能的方法。例如，在我们的设置中，模型不被告知像素强度的原创性，并将它们作为独立符号使用。我们观察到，对于大多数这些模型来说，学习反映这一属性的嵌入是相当困难的(图)。如果我们简单地用CNN梗替换嵌入层，我们会看到性能的模仿提升(例如，用对流梗替换vanilla Transformer的嵌入层，用3 ×3 kernel，我们得到75.32的准确率)。
另一个可以带来更好性能的修改是在Transformer模型中纳入具有平移不变性的空间表示(例如，向vanilla Transformer添加2D相对位置嵌入，我们得到61.72的精度)。然而，添加这些类型的更改使设置偏离了我们基准中这项任务的初始点。

A.2.2 由vanilla transformer实现的倾斜嵌入可视化

图4展示了在灰度CIFAR10测试上，普通变换模型为图像分类任务学习到的像素强度和位置嵌入的可视化效果。

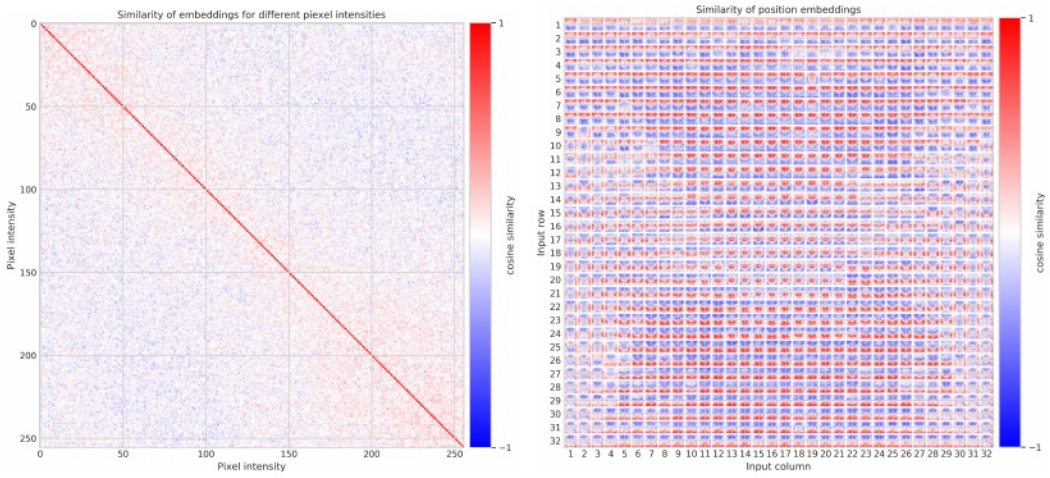


图4:左:为每个像素强度学习的嵌入之间的余弦相似度。右:每个贴片显示了所指示行和列的像素的位置嵌入与所有其他像素的位置嵌入之间的余弦相似度。

在左侧，我们可以看到学习到的嵌入对像素强度的两两相似性。虽然对于接近的像素值有更高的相似性，但来自这些学习到的嵌入的模式并不能完美地反映像素强度的有序性。在右边，我们可以看到不同输入位置的位置嵌入的两两相似性。我们可以看到，越低的

两个像素之间的距离是，它们学习到的位置嵌入就越相似。然而，在学习的嵌入中，y轴上的空间接近性比x轴上的距离更能保持。

A.3 探路者

探路者任务探测模型检测输入特征之间的长距离空间依赖性的能力。为了解决这个任务，模型需要识别目标轮廓，并从一端追踪到另一端。尽管探路者在视觉上是一项简单的任务，但已有研究表明，路径形状的杂乱和变化使得CNN模型的任务变得困难(Linsley等人，2018; Kim* et al.， 2020)。

探路者任务为二值分类任务，输入图像分辨率为32 ×32。与图像分类任务类似，我们在将输入图像展平后，给我们的xformer编码器提供1024像素的序列。xformer编码器顶部的分类器头也是一个ReLU激活的两层MLP，我们使用Softmax交叉熵损失进行优化。我们对我们的模型进行了200个epoch的训练。xformer模型使用的超参数如下:4层，8个头，128作为FFN块的隐藏维度，128作为查询/键/值隐藏维度，学习率为0.01。

A.3.1 来自vanilla transformer的注意力图的可视化

考虑到transformer有许多具有全局感受野的单元，与具有局部感受野的模型相比，它们具有更好的解决任务的潜力。图5显示了以令牌(CLS令牌)作为查询给出的一组示例的注意力分布。我们可以看到，注意力模块从输入中的不同位置收集信息，以便能够追踪目标路径。

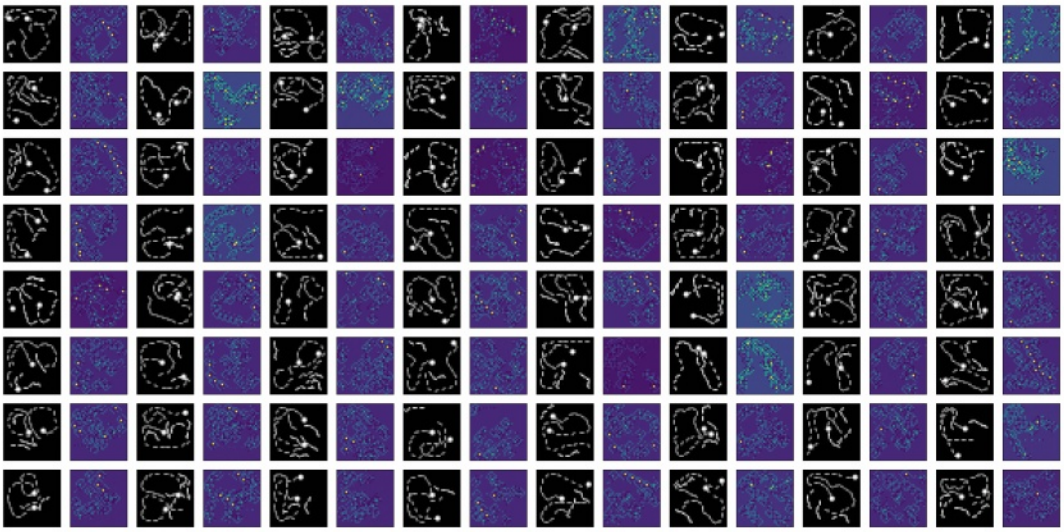


图5:来自探路者任务的不同例子的注意图。在一个普通Transformer模型中，给定最后一层的CLS令牌作为查询，每个映射表示在所有头部上的平均注意力分布。请注意，对于可视化，我们使用attention-rollout (Abnar & Zuidema, 2020)以获得更精确的输入归因。

我们还在LRA中包含了一个探路者- x，它类似于探路者，但输入分辨率更高，即输入序列更长。在探路者- x上，我们尝试了两种设置来训练我们的模型，第一次从头开始训练模型，第二次评估在探路者上训练的模型。在这两种情况下，我们发现没有一个模型能够处理/泛化到16K输入长度。

B模型和实现

本节描述了我们实现的细节。代码主要是用JAX和FLAX编写的。在本节中，我们注意到关于模型的某些实现的具体细节。我们计划稍后以自述文件或脚本的形式发布超参数。

B.1 模型实现的简要概述

虽然大多数细粒度的细节计划在发布的代码中可用，但我们提供了正在评估的xformer模型的一些设置的简要概述。对于局部注意力，我们不使用重叠块。对于Sinkhorn Transformer块内的局部注意力，我们也不重叠窗口。对于linformer，投影在键和值之间共享，但不跨多个层。对于Performer模型，我们的实现使用FAVOR+，这是Choromanski et al. (2020b)论文中的最新版本。

B.2 我们实现的特殊情况

本节介绍了我们实现细节中的几个特殊情况。多样化的transformer套件带来了大量的硬件约束和实现细节。要成功，Transformer模型还需要“赢得”硬件彩票(Hooker, 2020)，即随时支持ops、内核或加速器支持，以利用其技术设计的优势。本节讨论了一些权衡和边缘情况，使几个模型的比较具有挑战性。最后，我们认为简单是一种优点，不需要任何特殊支持对于高效的Transformer模型是一件积极的事情。

在CUDA内核上，CUDA内核很麻烦，并且是特定于GPU硬件的，这使得它很难在TPU pod上实现或使用。一般来说，这些在实际应用中被认为是不可取的和不方便的。因此，稀疏变压器和长变换器使用等效实现来模拟性能。这是通过应用等效的掩码实现的。出于这个原因，我们没有对稀疏变压器和Longformer进行速度基准测试。

Reformer的实现优化了ops以支持Reformer的许多功能是至关重要的。因此，Reformer的实现与其他Transformer模型略有不同。我们不是计算具有批量大小维度B和头部维度H的张量(即B ×H × N ×d)，而是计算N ×d维度张量的注意力函数。之后，我们通过VMAP在批尺寸和头部尺寸上并行化这个函数。

C公平比较的建议

我们欢迎在任何任务上重新评估我们的模型。然而，我们认为一些超参数是不可变的，以确保与所有模型的公平比较。在提出新模型的情况下，只要(1)模型大小保持不变，(2)不进行预训练，(3)不改变基本设置(例如，将字符级别更改为单词级别或向图像任务添加空间信息)，论文中的LRA表就可以复制。我们将在https://github.com/google-research/long-range-arena上提供更多详细信息。