如果提供了适当的归属，谷歌在此授予许可

arXiv:1706.03762v7 [cs.CL]2023年8月2日

复制本文中的表格和图表，仅用于新闻报道或新闻报道

学术著作。



**注意力是你所需要的**

**阿什瓦瓦尼**\*

谷歌大脑

avaswani@google.com

**诺姆沙泽尔**\*

谷歌大脑

noam@google.com

**尼基帕尔马**\*

谷歌研究

nikip@google.com

**雅各布乌斯科里特**\*

谷歌研究

usz@google.com

**狮子琼斯**\*

谷歌研究

llion@google.com

**艾丹NGomez.** \* †

多伦多大学

aidan@cs.toronto.edu

**Łukasz凯撒**\*

谷歌大脑

lukaszkaiser@google.com

**伊丽亚波洛苏金**\* ‡

illia.polosukhin@gmail.com

**摘要**

主要的序列转导模型是基于复杂的循环或卷积神经网络，其中包括一个编码器和一个解码器。性能最好的模型还通过一种注意机制将编码器和解码器连接起来。我们提出了一种新的简单的网络架构，变压器，完全基于注意机制，完全取消递归和卷积。在两个机器翻译任务上的实验表明，这些模型在质量上更优越，同时更并行，需要的训练时间明显更少。我们的模型达到28.4 BLEU WMT 2014英德翻译任务，改善现有的最佳结果，包括集成，超过2 BLEUOn WMT 2014英法翻译任务，我们的模型建立一个新的单模型的最先进的BLEU得分41.8训练后3.5天8图形处理器，一小部分的培训成本最好的模型。. 我们成功地将变压器应用于具有大量和有限训练数据的英语选区解析，可以很好地推广到其他任务。

\*同样的贡献。清单顺序是随机的。雅各布提议用自我注意力取代rnn，并开始努力评估这一想法。阿什什和伊利亚一起设计并实现了第一个变压器模型，并关键地参与了这项工作的各个方面。Noam提出了比例点积注意力、多头注意力和无参数位置表示，成为几乎每一个细节的另一个人。Niki设计、实现、调整和评估了我们原始代码库和张量2张量中的无数模型变体。Llion还实验了新的模型变体，负责我们的初始代码库，以及有效的推理和可视化。Lukasz和Aidan花了无数天的时间设计张量2张量的各个部分，取代了我们早期的代码库，大大提高了结果，大大加速了我们的研究。

†在谷歌Brain公司完成的工作。

‡在谷歌研究期间完成的工作。

第31届神经信息处理系统会议（NIPS 2017），长滩，美国加州。

**1介绍**

[循环神经网络，长短期记忆[13]和门控循环[7]神经网络](#bookmark1)特别是，已经牢固地建立为最先进的方法在序列建模和[转导问题，如语言建模和机器翻译[35，](#bookmark2)[2,](#bookmark3)[5].众多](#bookmark4)此后，人们的努力继续推动循环语言模型和编解码器的边界[架构[38，](#bookmark5)[24,](#bookmark6)[15]](#bookmark7).

循环模型通常会沿着输入和输出序列的符号位置进行因素计算。将位置与计算时间中的步骤对齐，它们生成一个隐藏序列

lahtgl.ca

序列长度，因为内存限制限制了示例之间的批处理。最近的工作已经实现[通过[21]和条件分解显著提高计算效率](#bookmark8)[计算[32]，同时也提高了模型在后者的情况下的性能。基本的](#bookmark9)然而，顺序计算的约束仍然存在。

注意机制已经成为各种任务中引人注目的序列建模和转导模型的一个组成部分，允许对依赖关系进行建模，而不考虑它们之间的距离[输入或输出序列[2，](#bookmark3)[19].然而，除了少数情况下，所有的[27]，这种注意机制](#bookmark10)与递归网络一起使用。

在这项工作中，我们提出了变压器，一个避免重复出现的模型架构，而是完全依赖于一个注意机制来绘制输入和输出之间的全局依赖关系。变压器允许更多的并行化，在8个P100gpu上进行了短短12个小时的训练后，可以达到一种新的翻译质量技术状态。

**2背景**

减少顺序计算的目标也构成了扩展神经GPU的基础[[16]，ByteNet [18]和ConvS2S [9]，它们都使用卷积神经网络作为基础构建](#bookmark11)块，并行计算所有输入和输出位置的隐藏表示。在这些模型中，关联来自两个任意输入或输出位置的信号所需的操作数量随着位置之间的距离而增长，ConvS2S呈线性增长，ByteNet呈对数增长。这使得[学习遥远位置[12]之间的依赖关系更加困难。在变压器中，这是](#bookmark12)减少到恒定数量的操作，尽管以由于平均注意力加权位置而降低有效分辨率为代价，我们用第节所述的多头注意力来抵消这一影响[3.2.](#bookmark13)

自我注意，有时被称为内注意，是一种注意机制，将单个序列的不同位置联系起来，以计算序列的表示。自我注意已经成功地应用于各种任务中，包括阅读理解、抽象总结、[文本隐含和学习过程中与任务无关的句子表征[4，](#bookmark14)[27,](#bookmark10)[28,](#bookmark15)[22]](#bookmark16).

端到端记忆网络基于循环注意机制，而不是序列对齐递归，并已被证明在简单语言问题回答方面表现良好[语言建模任务[34]](#bookmark17).

然而，据我们所知，变压器是第一个完全依赖于自我关注来计算其输入和输出的表示的转导模型，而不使用序列对齐的rnn或卷积。在下面的章节中，我们将描述变压器，激励[自我关注，并讨论其相对于[17，](#bookmark18)[18]和[9]](#bookmark11).

**3模型体系结构**

[大多数竞争性的神经序列转导模型都有一个编码器-解码器结构[5，](#bookmark4)[2,](#bookmark3)[35]](#bookmark2).在这里，编码器映射了一个符号表示的输入序列(x1,..., xn)到一个连续表示序列z = (z1,..., zn).给定z，解码器然后生成一个输出序列(y1,...,ym)的符号，一次一个元素。在每一步中，该模型都是自回归的[[10]，在生成下一个符号时，使用先前生成的符号作为附加输入。](#bookmark19)

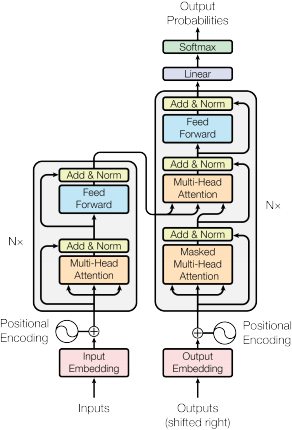


图1：变压器模型架构。

变压器遵循这种整体架构，使用堆叠的自注意和点级的全连接层，如图的左半部分和右半部分所示[1,](#bookmark20)各自地

**3.1编码器和解码器堆栈**

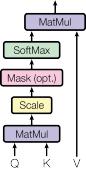
编码器：编码器由N个=6个相同的层组成。每个层都有两个子层。第一种是一个多头的自我注意机制，第二种是一个简单的，位置-[明智的全连接的前馈网络。我们在每个节点周围使用一个残差连接[11]](#bookmark21)[这两个子层，然后是层归一化[1]。即，每个子层的输出为](#bookmark22)LayerNorm（x +子层(x)），其中子层(x)是由子层本身实现的函数。为了方便这些剩余连接，模型中的所有子层以及嵌入层都会产生维数为d的输出模型= 512.

解码器：解码器也由N = 6个相同的层组成。除了每个编码器层中的两个子层外，解码器还插入第三个子层，该子层对编码器堆栈的输出执行多头注意。与编码器类似，我们在每个子层周围使用剩余连接，然后进行层归一化。我们还修改了解码器堆栈中的自注意子层，以防止位置关注后续的位置。这种掩蔽，加上输出嵌入被一个位置偏移的事实，确保了对位置i的预测只能依赖于小于i的位置的已知输出。

**3.2注意**

注意函数可以描述为将一个查询和一组键-值对映射到一个输出，其中查询、键、值和输出都是向量。输出以加权和计算

缩放点产品注意事项



多头注意力

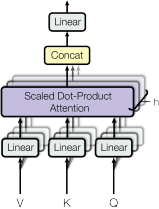


图2：（左）比例点产品注意力。（右）多头注意力由几个并行运行的注意层组成。

其中，分配给每个值的权重是由查询与相应键的兼容性函数计算的。

**3.2.1扩展点产品关注**

我们将我们的特别关注称为“比例点产品关注”[2).输入的内容包括：](#bookmark23)维度d的查询和键k，和维数d的值v.我们用所有键计算查询的点积，用每个键除以√dk，并应用一个softmax函数来获得这些值的权重。

在实践中，我们同时计算一组查询上的注意函数，并将它们打包成一个矩阵Q。键和值也被打包在矩阵K和V中。我们计算输出的矩阵为：

注意事项（Q、K、V) = softmax（）V (1)

[两个最常用的注意函数是加性注意[2]和点积(多](#bookmark3)-显而易见的)注意。点积注意与我们的算法相同，除了比例因子 一个隐藏的层。虽然两者在理论复杂性上相似，但点积的关注是在实践中更快、更节省空间，因为它可以使用高度优化来实现矩阵乘法码。

而对于较小的d值k这两种机制的表现相似，在d值更大的情况下，加法注意优于点积注意k[[3].我们怀疑，对于较大的值](#bookmark24)dk，点产品增长巨大，将softmax函数推到它有的区域极小的梯度[4 .为了抵消这种效应，我们用](#bookmark25).

**3.2.2多头注意**

而不是用d来执行单一的注意功能模型-维度键、值和查询，我们发现用不同的、学习到的线性投影对d的线性投影是有益的k, dk和dv尺寸，分别。在每个查询、键和值的投影版本上，我们并行地执行注意函数，生成dv…维的

4为了说明为什么点积变得很大，假设q和k的分量是独立随机的均值为0和方差为1的变量.然后是他们的点积，q·k =：1qiki，的均值为0，方差为dk .

输出值。这些值被连接起来并再次投影，从而得到最终的值，如图所示[2.](#bookmark23)

多头注意允许模型共同关注来自不同位置的不同表示子空间的信息。用一个注意力头，平均可以抑制这一点。

多头（Q、K、V）=Concat（头1头h)WO

在那里头部i=注意(QWiQ, KWiK, VWiV)

其中的投影是参数矩阵WiQ∈ Rd模型×dk, WiK∈ Rd模型×dk, WiV∈ Rd模型×dv和WO∈ Rhdv×d模型.

在这项工作中，我们使用了h = 8平行的注意层，或头部。对于我们使用的每一个k= dv= d模型/h = 64.由于每个头部的维数降低，其总计算代价与全维的单头注意相似。

**3.2.3在我们模型中的应用**

该变压器以三种不同的方式使用多头注意力：

在“编码器-解码器注意”层中，查询来自上一个解码器层，内存密钥和值来自编码器的输出。这允许解码器中的每个位置都参与输入序列中的所有位置。这模拟了序列到序列模型中典型的编码器-解码器注意机制，例如[[38,](#bookmark5)[2,](#bookmark3)[9]](#bookmark11).

该编码器包含自我注意层。在自注意层中，所有的键、值和查询都来自同一个位置，在这种情况下，是编码器中上一层的输出。编码器中的每个位置都可以处理编码器上一层中的所有位置。

类似地，解码器中的自注意层允许解码器中的每个位置关注解码器中的所有位置，直到并包括该位置。我们需要防止解码器中向左的信息流，以保持自回归特性。我们实现这个

sfrtgaet[.](#bookmark23)输入值中的所有值

**3.3位置级前馈网络**

除了注意子层外，我们的编码器和解码器中的每个层都包含一个完全连接的前馈网络，它分别和相同地应用于每个位置。这由两个线性变换组成，中间有一个ReLU激活。

FFN (x) = max (0,xW1+ b1)W2+ b2(2)

虽然线性转换在不同的位置上是相同的，但它们每层都使用不同的参数。另一种描述它的方法是用核大小为1的两个卷积。输入和输出的维数为d模型= 512，内层的维数为dff= 2048.

**3.4嵌入和软max**

与其他序列转导模型类似，我们使用学习到的嵌入来将输入标记和输出标记转换为维数为d的向量模型.我们还使用通常的学习线性变换和softmax函数来转换解码器输出到预测的下一个令牌概率。在我们的模型中，我们在两个嵌入层和预softmax之间共享相同的权重矩阵[线性变换，类似于[30]。在嵌入层中，我们将这些权重乘以](#bookmark27)√d模型.

表1：不同层类型的最大路径长度、每层复杂度和最小的顺序操作数。n是序列长度，d是表示维数，k是卷积的核大小，r是受限自注意条件下邻域的大小。

|  |  |  |  |
| --- | --- | --- | --- |
| 层类型 | 每层复杂度 | 顺序操作 | 最大路径长度 |
| 自注意力 | O(n2· d) | O(1) | O(1) |
| 经常性 | O(n · d2) | O(n) | O(n) |
| 卷积 | O(k · n · d2) | O(1) | O（日志k(n)) |
| 自我注意（受限） | O(r · n · d) | O(1) | O(n/r) |

**3.5位置编码**

由于我们的模型不包含递归性和卷积，为了使模型利用序列的顺序，我们必须注入一些关于序列中标记的相对或绝对位置的信息。为此，我们在编码器和解码器堆栈的底部向输入嵌入中添加“位置编码”。位置编码具有相同的维度d模型作为嵌入，使两者可以求和。位置编码有很多选择，[学习和固定[9]](#bookmark11).

在这项工作中，我们使用不同频率的正弦和余弦函数：

体育(pos,2i)=罪(pos/100002i/d模型)

体育(pos,2i+1)= cos (pos/100002i/d模型)

其中pos是位置，i是维度。也就是说，位置编码的每个维度都对应于一个正弦曲线。波长形成了一个从2π到10000·2π的几何级数。我们选择这个函数是因为我们假设它将允许模型很容易地学习参加的相对位置，因为对于任何固定的偏移量k，PEpos+k可以表示为的线性函数吗体育pos.

[我们还实验使用学习到的位置嵌入[9]来代替，并发现这两者](#bookmark11)各版本产生了几乎相同的结果(见表[3](#bookmark29)行(E)。我们选择正弦版本，因为它可能允许模型推断比训练中遇到的序列长度。

**4.自我关注的原因**

在本节中，我们将比较自我注意层的各个方面与通常用于映射一个变长的符号表示序列的循环层和卷积层(x1,..., xn)到另一个等长的序列(z1,..., zn)，带xi, zi∈ Rd，如在一个典型的序列转导编码器或解码器中的一个隐藏层。为了激励我们使用自我注意，我们考虑了三个需求。

一个是每层的总计算复杂度。另一个是可以并行化的计算量，通过所需的顺序操作的最小数量来衡量。

第三个是网络中长期依赖关系之间的路径长度。学习长期依赖关系是许多序列转导任务中的一个关键挑战。影响学习这种依赖关系能力的一个关键因素是信号在网络中向前和向后穿过的路径的长度。输入中任何位置组合之间的路径越短[和输出序列，就越容易学习远程依赖关系[12]。因此，我们也比较](#bookmark12)由不同层类型组成的网络中任意两个输入和输出位置之间的最大路径长度。

如表所示[1、一个自注意层以恒数的顺序连接所有位置](#bookmark28)已执行的操作，而循环层需要O (n)个顺序操作。在计算复杂度方面，自注意层比循环层更快

长度n小于表示维数d，这是机器翻译中最先进的模型使用的句子表示的情况，如词块[[38]和字节对[31]表示，以提高涉及的任务的计算性能.](#bookmark30) 很长的序列，自我注意可以被限制在只考虑输入序列中围绕各自输出位置的一个邻域。这将增加最大路径长度到O（n/r）。我们计划在今后的工作中进一步研究这种方法。

核宽度为k < n的单个卷积层并不能连接所有的输入和输出位置对。在连续内核的情况下，这样做需要一堆O（n/k）卷积层，或者是O (logk[(n))在扩张卷积[18]的情况下，增加最长路径的长度](#bookmark31)在网络中的任意两个位置之间。卷积层通常比循环层更昂贵，相差k倍[.然而，可分离的卷积[6]，降低了复杂度](#bookmark32)在相当大的程度上，对O (k·n·d+·d2. )然而，即使使用k = n，可分离卷积的复杂性也等于自注意层和点向前馈层的组合，这是我们在模型中采用的方法。

. 作为附带好处，自我注意可以产生更可解释的模型。我们检查来自我们的模型的注意力分布，并在附录中提出和讨论例子。不仅个体的注意力头清楚地学会了执行不同的任务，而且许多人似乎表现出与句子的句法和语义结构相关的行为。

**5培训**

本节描述了我们的模型的训练机制。

**5.1培训数据和批处理**

我们在标准的WMT 2014英德数据集上进行了训练，该数据集约有450万人[句子对。句子使用字节对编码的[3]进行编码，它有一个共享的源](#bookmark24)-目标词汇量约为37000个令牌。对于英法数据，我们使用了明显更大的WMT 2014英法数据集，其中包含3600万个句子，并将标记分解为32000个字块[词汇[38]。句子对按近似的序列长度组合在一起。每次培训](#bookmark5)该批处理包含一组句子对，其中包含大约25000个源标记和25000个目标标记。

**5.2硬件和时间表**

我们在一台使用8个NVIDIA P100gpu的机器上训练我们的模型。对于我们使用整个论文中描述的超参数的基础模型，每个训练步骤大约需要0.4秒。我们总共训练了基本模型10万步或12个小时。对于我们的大模型，(在表的底线上描述[3)，步长时间为1.0秒。这些大型模型接受了30万步的训练](#bookmark29)(3.5天）。

**5.3优化器**

[我们使用了Adam优化器[20]和β](#bookmark33)1= 0.9, β2= 0.98和ϵ= 10−9.我们在整个训练过程中改变了学习率，计算公式如下：

lrate = d·最小值(step\_num−0.5, step\_num · warmup\_steps− 1.5) (3)

这对应于第一个warmup\_steps训练步骤的学习率线性增加，然后按步数的平方反比成比例降低。我们使用了warmup\_steps = 4000。

**5.4正则化**

我们在培训过程中采用了三种类型的正规化：

表2：在2014年英语对德语和英语对法的新测试中，变压器获得了更好的BLEU分数，但培训成本只有培训成本的一小部分。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模型 | 有蓝色霉菌花纹的 | | 培训成本（FLOP） | |
| EN-DE | EN-FR | EN-DE | EN-FR |
| [ByteNet [18]](#bookmark31) | 23.75 |  |  |  |
| [Deep-Att + PosUnk [39]](#bookmark36) |  | 39.2 |  | 1.0 · 1020 |
| [GNMT + RL [38]](#bookmark5) | 24.6 | 39.92 | 2.3 · 1019 | 1.4 · 1020 |
| [ConvS2S [9]](#bookmark11) | 25.16 | 40.46 | 9.6 · 1018 | 1.5 · 1020 |
| [MoE [32]](#bookmark9) | 26.03 | 40.56 | 2.0 · 1019 | 1.2 · 1020 |
| [Deep-Att + PosUnk集成系统[39]](#bookmark36) |  | 40.4 |  | 8.0 · 1020 |
| [GNMT + RL集成系统[38]](#bookmark5) | 26.30 | 41.16 | 1.8 · 1020 | 1.1 · 1021 |
| [ConvS2S集成[9]](#bookmark11) | 26.36 | **41.29** | 7.7 · 1019 | 1.2 · 1021 |
| 变压器（基础模型） | 27.3 | 38.1 | **3.3 ·** | **1018** |
| 变压器（大） | **28.4** | **41.8** | 2.3 · | 1019 |

**残余辍学率**[我们将每个子层的输出应用辍学[33]，然后将其添加到](#bookmark37)子层输入和归一化。此外，我们还将辍学法应用于编码器和解码器堆栈中的嵌入和位置编码的和。对于基本模型，我们使用的速率P落下= 0.1.

标签平滑在训练过程中，我们采用了值ϵ的标签平滑ls= 0.1[[36].这个](#bookmark38)伤害了困惑，因为模型学会了更不确定，但提高了准确性和BLEU分数。

**6结果**

**6.1机器翻译**

在WMT 2014英德翻译任务中，表中的大变压器模型(变压器（大）[2)的性能超过了以前报道的最佳模型（包括集成模型）](#bookmark35)2.0 BLEU，建立了一个新的最先进的BLEU分数28.4。该模型的配置列在表的底线中[3.培训需要3](#bookmark29).5 每天使用8p100gpu。即使是我们的基础模型也超过了之前发表的所有模型和集合，其训练成本只是任何竞争模型的一小部分。

在WMT 2014年的英法翻译任务中，我们的大模型获得了41.0分的BLEU分数，优于之前发布的所有单个模型，其训练成本不到之前最先进的模型的1/4。为英法英语训练的变压器（大）模型使用了辍学率P落下=为0.1，而不是0.3。

对于基本模型，我们使用了一个通过将最后5个检查点平均得到的单一模型，这些检查点每隔10分钟编写一次。对于大型模型，我们取了最后20个检查点的平均值。我们使用光束大小为4和长度惩罚α = 0.6[[38].这些超参数](#bookmark5)是在开发集上进行实验后选择的。我们将推理过程中的最大输出长度设置为输入长度+ 50[，但在可能时尽早终止[38]](#bookmark5).

表[2](#bookmark35)总结了我们的结果，并将我们的翻译质量和培训成本与文献中的其他模型架构进行了比较。我们通过将训练时间、使用的GPU的数量以及对每个GPU的持续单精度浮点容量的估计相乘，来估计用于训练模型的浮点运算的数量[5 .](#bookmark39)

**6.2模型变化**

为了评估变压器的不同组件的重要性，我们以不同的方式改变了我们的基础模型，测量了英德语翻译的性能变化

5我们使用的K80、K40、M40和P100的值分别为2.8、3.7、6.0和9.5 TFLOPS。

表3：变压器体系结构的变化情况。未列出的值与基本模型的值相同。所有指标都在2013年新的英语-德语翻译开发集上。根据我们的字节对编码，列出的困惑是每个字段的，不应该与每个单词的困惑相比较。

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | N | d模型dffh dkdvP落下ϵls | | | 火车  步骤 | PPL (dev) | BLEU（开发） | 帕拉姆×106 |
| 基础 | 6 | 512 2048 8 64 64 0.1 0.1 | | | 100K | 4.92 | 25.8 | 65 |
| (A) |  | 1  4  16  32 | 512  128  32  16 | 512  128  32  16 |  | 5.29  5.00  4.91  5.01 | 24.9  25.5  25.8  25.4 |  |
| (B) |  | 16  32 | | |  | 5.16  5.01 | 25.1  25.4 | 58  60 |
| (C) | 2  4  8 | 256  1024  1024  4096 | 32  128 | 32  128 |  | 6.11  5.19  4.88  5.75  4.66  5.12  4.75 | 23.7  25.3  25.5  24.5  26.0  25.4  26.2 | 36  50  80  28  168  53  90 |
| (D) |  | 0.0 0.2  0.0 0.2 | | |  | 5.77  4.95  4.67  5.47 | 24.6  25.5  25.3  25.7 |  |
| (E) |  | 位置嵌入代替正弦线 | | |  | 4.92 | 25.7 |  |
| 大的 | 6 | 1024 4096 16 0.3 | | | 300K | **4.33** | **26.4** | 213 |

开发集，2013年最新。我们使用了前一节所述的波束搜索，但没有进行检查点平均。我们在表中展示了这些结果[3.](#bookmark29)

在表中[3](#bookmark29)第(A)行，我们改变注意头的数量和注意键和值维度，保持计算量不变，如第节所述[3.2.2.](#bookmark26)虽然单头人的注意力比最佳设置差0.9 BLEU，但质量也会下降。

在表中[3](#bookmark29)行(B)时，我们观察到减少了注意键的大小k伤害模型质量。这表明，确定兼容性并不容易，而一个比点产品更复杂的兼容性功能可能是有益的。我们在(C)和(D)行中进一步观察到，正如预期的那样，较大的模型更好，并且退出对于避免非常有助于过拟合。在(E)行，我们替换我们的[正弦位置编码与学习到的位置嵌入[9]，并观察到几乎相同](#bookmark11)结果到基础模型。

**6.3英语选区解析**

为了评估变压器是否可以推广到其他任务，我们在英语选区解析上进行了实验。这项任务提出了具体的挑战：输出受到强大的结构约束，并且明显长于输入的时间。此外，RNN序列对序列[模型还不能在小数据机制[37]中获得最先进的结果](#bookmark40).

我们用d模型= 1024在华尔街日报（WSJ）的部分[宾州树库[25]，大约4万个训练句子。我们还在半监督的环境下训练了它，](#bookmark41)使用更大的高可信度和伯克利解析器语料库，其中大约有1700万个句子[[37].我们在《华尔街日报》中使用了16K令牌的词汇和32K令牌的词汇](#bookmark40)用于半监督设置。

我们只做了少量的实验来选择辍学，包括注意和残差(部分[5.4)，学习速率和光束大小在第22节开发集，所有其他参数](#bookmark34)与英语到德语的基本翻译模式保持不变。在推理过程中，我们

表4：变压器可以很好地推广到英国选区分析（结果在《华尔街日报》第23节上）

|  |  |  |
| --- | --- | --- |
| **解析器** | **训练** | **WSJ 23 F1** |
| [葡萄酒和凯撒啤酒。(2014) [37]](#bookmark40) | 仅限《华尔街日报》，有区别 | 88.3 |
| [彼得罗夫等人。(2006) [29]](#bookmark43) | 仅限《华尔街日报》，有区别 | 90.4 |
| [朱等人。(2013) [40]](#bookmark44) | 仅限《华尔街日报》，有区别 | 90.4 |
| [戴尔等人。(2016) [8]](#bookmark45) | 仅限《华尔街日报》，有区别 | 91.7 |
| 变压器（4层） | 仅限《华尔街日报》，有区别 | 91.3 |
| [朱等人。(2013) [40]](#bookmark44) | 半监督 | 91.3 |
| [黄和哈珀（2009）[14]](#bookmark46) | 半监督 | 91.3 |
| [麦克洛斯基等人。(2006) [26]](#bookmark47) | 半监督 | 92.1 |
| [葡萄酒和凯撒啤酒。(2014) [37]](#bookmark40) | 半监督 | 92.1 |
| 变压器（4层） | 半监督 | 92.7 |
| [Luong等。(2015) [23]](#bookmark48) | 多任务 | 93.0 |
| [戴尔等人。(2016) [8]](#bookmark45) | 能生产的 | 93.3 |

将最大输出长度增加到输入长度+ 300。我们使用的光束尺寸为21，α=为0.3，仅用于WSJ和半监督设置。

我们的结果见表[4](#bookmark42)结果表明，尽管缺乏特定于任务的调优，但我们的模型表现得非常好，产生了比以前所有报告的模型更好的结果，除了[递归神经网络语法[8]](#bookmark45).

[与RNN序列到序列模型[37]相比，变压器的性能优于伯克利](#bookmark40)-[解析器[29]，即使只在《华尔街日报》的训练集上进行训练](#bookmark43).

**7结论**

在这项工作中，我们提出了变压器，第一个完全基于注意的序列转导模型，用多头自注意取代了编解码器结构中最常用的循环层。

对于翻译任务，变压器的训练速度可以比基于循环层或卷积层的架构要快得多。在WMT 2014英德语和WMT 2014英法翻译任务中，我们实现了一种新的艺术状态。在前一项任务中，我们最好的模型甚至优于所有之前报道的集成。

我们对基于注意力的模型的未来感到很兴奋，并计划将它们应用到其他任务中。我们计划将变压器扩展到涉及文本以外的输入和输出模式的问题，并研究局部的、受限的注意机制，以有效地处理大型输入和输出，如图像、音频和视频。减少世代是我们的另一个研究目标。

我们用于训练和评估我们的模型的代码可以在[https://github.com/tensorflow/tensor2tensor.](https://github.com/tensorflow/tensor2tensor)

我们非常感谢纳尔·卡尔奇布伦纳和斯蒂芬·古斯提供的富有成效的评论、修正和灵感。

**参考文献**

[1]吉米雷巴，杰米瑞安基罗斯，和杰弗里E辛顿。层规范化*arXiv预印本*[*arXiv:1607.06450,*](http://arxiv.org/abs/1607.06450)2016.

[2]，赵庆云，和本虎。. 神经机器翻译通过联合学习对齐和翻译CoRR，abs/1409.0473,2014。

[3] Denny Britz，安娜戈尔迪，明通龙，和Quoc V。黎巴嫩对神经机器翻译体系结构的大规模探索。CoRR，abs/1703.03906,2017年。

[4]建鹏，李东和拉帕塔。用于机器读取的长时间短期内存网络。*arXiv预印本*[*arXiv:1601.06733,*](http://arxiv.org/abs/1601.06733)2016.

[5]、巴特、卡格拉·古尔切尔、霍尔格和本吉奥。使用rnn编解码器学习短语表示来进行统计机器翻译。CoRR，abs/1406.1078,2014年。

[6]弗朗索瓦球。具有深度可分离卷积的深度学习。*arXiv预印本*[*arXiv:1610.02357,*](http://arxiv.org/abs/1610.02357)2016.

钟俊荣、葛志乐、赵庆云、本家耀。. 门控递归神经网络对序列模型的实证评价CoRR，abs/1412.3555,2014。

[8]克里斯·戴尔，昆科罗，米格尔·巴莱斯特罗斯和诺亚·A。铁匠递归的神经网络语法。*.* 在NAACL的过程中，2016。

[9]乔纳斯格林，迈克尔奥利，大卫格兰奇尔，丹尼斯雅拉茨，和扬N。多芬。卷积序列到序列学习。*arXiv预印本*[*arXiv:1705.03122v2, 2017*](http://arxiv.org/abs/1705.03122).

[10]亚历克斯坟墓。用递归神经网络生成序列。*arXiv预印本*

[*arXiv:1308.0850,*](http://arxiv.org/abs/1308.0850)2013.

何[11]凯明、张翔宇、任少卿、孙吉安。用于图像识别的深度残差学习。发表在IEEE计算机视觉和模式识别会议论文集上，第770-778页，2016年。

[12]9霍克雷特，本吉奥，保罗弗拉斯科尼，和尤尔根施米杜贝尔。《循环网中的梯度流动：学习长期依赖的困难》，2001年。

[13]9霍克雷特和雨尔根施米德贝尔。长期记忆。神经计算法，9(8)：1735-1780年，1997年。

[14]中强·黄和玛丽·哈珀。使用跨语言的潜在注释的自我训练PCFG语法。在2009年自然语言处理中的经验方法会议的论文集中，第832-841页。ACL，2009年8月。

[15]先生，酒店老板，舒斯特，谢泽，吴永辉。探索语言建模的局限性。*arXiv预印本*[*arXiv:1602.02410,*](http://arxiv.org/abs/1602.02410)2016.

[16]Łukasz Kaiser和Samy Bengio。主动记忆能取代注意力吗？*《神经信息处理系统的研究进展》，（NIPS），2016年。*

[17]Łukasz Kaiser和伊利亚·苏特斯科弗。神经gpu学习算法。*2016年，国际学习代表会议（ICLR）。*

[18]纳尔·卡尔什布伦纳，拉斯·埃斯佩霍尔特，凯伦·西蒙扬，亚伦·范德诺德，亚历克斯·格雷夫斯和科雷·卡武克的线性时间神经机器翻译预印本. . [*arXiv:1610.10099v2*](http://arxiv.org/abs/1610.10099), 2017.

[19]尹金，卡尔丹顿，龙勋，和亚历山大M。猛冲结构化的注意力网络。*2017年，国际会议。*

[20]·迪德里克·金玛和吉米·巴。一种随机优化的方法。在ICLR，2015年。

[21] Oleksii·库查耶夫和鲍里斯·金斯伯格。针对LSTM网络的因子分解技巧。*arXiv预印本*[*arXiv:1703.10722,*](http://arxiv.org/abs/1703.10722)2017.

林周汉、冯敏、多斯桑托斯、俞、冰翔、周文文、本书。一个结构化的自注意句子嵌入器. [*arXiv:1703.03130,*](http://arxiv.org/abs/1703.03130)2017.

[23]Minh-通龙，引用，伊利亚苏茨克，奥里奥尔，和卢卡斯凯撒。. 多任务序列到序列学习。*arXiv预印本*[*arXiv:1511.06114,*](http://arxiv.org/abs/1511.06114)2015.

[24]Ming-阮，休范和克里斯托弗·曼宁。基于注意力的神经机器翻译的有效方法。*arXiv预印本*[*arXiv:1508.04025,*](http://arxiv.org/abs/1508.04025)2015.

[25]米切尔P马库斯，玛丽安马辛基维奇，和比阿特丽斯桑托里尼建造一个大型注释.

英语语料库：宾夕法尼亚大学树树银行。计算语言学，19(2)：313-330,1993。

[26]的大卫·麦克洛斯基，尤金·查尼亚克和马克·约翰逊。有效的解析自我训练。*在naacl人类语言技术会议论文集，主要会议，第152-159页。*ACL，2006年6月。

[27] Ankur Parikh，奥斯卡·塔克斯特罗姆，迪潘扬达斯，和雅各布·乌斯科利特。一个可分解的注意力模型。*在自然语言处理中的经验方法中，2016年。*

[28] Romain保卢斯，熊先生和理查德·索舍尔。一个用于抽象摘要的深度强化模型。*arXiv预印本*[*arXiv:1705.04304,*](http://arxiv.org/abs/1705.04304)2017.

[29]斯拉夫彼得罗夫，里昂巴雷特，罗曼蒂波，和丹克莱恩。学习准确、紧凑和可解释的树注释。*在第21届计算语言学国际会议和第44届ACL年会》，第433-440页。ACL公司，2006年7月。*

[30] Ofir出版社和Lior Wolf出版社。使用输出嵌入来改进语言模型。*arXiv预印本*[*arXiv:1608.05859,*](http://arxiv.org/abs/1608.05859)2016.

[31] Rico森恩里希，巴里哈多，和亚历山德拉伯奇。带有子词单位的罕见词的神经机器翻译。*arXiv预印本*[*arXiv:1508.07909,*](http://arxiv.org/abs/1508.07909)2015.

[32] Noam谢泽尔，阿扎莉亚·米尔霍塞尼，马齐亚兹，安迪·戴维斯，库克勒，杰弗里·辛顿和杰夫·迪恩。极其庞大的神经网络：稀疏门控的专家混合层。*arXiv预印本*[*arXiv:1701.06538,*](http://arxiv.org/abs/1701.06538)2017.

[33] Nitish斯里瓦斯塔瓦，杰弗里·辛顿，亚历克斯·克里热夫斯基，伊利亚·萨斯科弗和鲁斯兰·萨拉库迪诺夫。辍学生：一种防止神经网络过拟合的简单方法。《机器学习研究杂志》，15(1)：1929-1958年，2014年。

[34]圣巴亚尔苏赫巴托，亚瑟斯拉姆，杰森韦斯顿，和罗伯费格斯。端到端内存网络在C。. 北达科罗来纳州。劳伦斯，D. D。李，M。杉山和R。*加内特，编辑，《神经信息处理系统的进展》，第28页，第2440-2448页。*柯伦联合公司，2015年。

[35] Ilya报价，报价和报价。用神经网络进行序列学习。《神经信息处理系统的进展》，第3104-3112页，2014年。

[36]克里斯蒂安·谢格迪，文森特·万霍克，谢尔盖·约菲，乔纳森·施伦斯和兹比格涅夫·沃伊纳。

重新思考计算机视觉的初始架构，corr，abs/1512.00567,2015。.

*[37]维尼亚尔和凯撒，古，彼得罗夫，萨斯克弗，和辛顿。语法作为一种外语。关于神经信息处理系统的研究进展，2015年。*

吴永辉、舒斯特、陈志峰、乐、诺鲁子、沃尔夫冈、克里昆、曹袁、秦高、麦克利等。谷歌的神经机器翻译系统：弥合人类和机器翻译之间的差距。*arXiv预印本*[*arXiv:1609.08144,*](http://arxiv.org/abs/1609.08144)2016.

周[39]杰、曹英、王旭光、李鹏、徐伟。. 具有神经机器平移的快进连接的深度循环模型CoRR，abs/1606.04199,2016。

朱[40] Muhua、张悦、陈文亮、张敏、朱静波。快速和准确的位移-减少成分解析。*第51届年度会议记录（第1卷：长篇），第434-443页。*ACL，2013年8月。

**注意可视化**

它

是

在

这

精神

那个

a

多数

的

美国人

政府

有

已经通过的

新的

法律

自…以后

2009

制造

那

登记

或

选举

过程

更

困难的

.

<EOS>

<pad>

<pad>

<pad>

<pad>

<pad>

<pad>

它

是

在

这

精神

那个

a

多数

的

美国人

政府

有

已经通过的

新的

法律

自…以后

2009

制造

那

登记

或

选举

过程

更

困难的

.

<EOS>

<pad>

<pad>

<pad>

<pad>

<pad>

<pad>

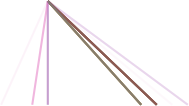




图3：第6层第5层编码器自注意中遵循长距离依赖关系的注意机制实例。许多注意力的头脑关注动词“制造”的遥远依赖，完成短语“制造……更困难”。这里只显示了“制造”这个词。不同的颜色代表不同的头。最好的颜色。



这个

法

愿意

从不

是

完美的

,

但是

它的

申请

应该

是

刚才

-

这

是

什么

我们

是

找不到的

,

在

我的

意见

.

<EOS>

<pad>

这个

法

愿意

从不

是

完美的

,

但是

它的

申请

应该

是

刚才

-

这

是

什么

我们

是

找不到的

,

在

我的

意见

.

<EOS>

<pad>

这个

法

愿意

从不

是

完美的

,

但是

它的

申请

应该

是

刚才

-

这

是

什么

我们

是

找不到的

,

在

我的

意见

.

<EOS>

<pad>

这个

法

愿意

从不

是

完美的

,

但是

它的

申请

应该

是

刚才

-

这

是

什么

我们

是

找不到的

,

在

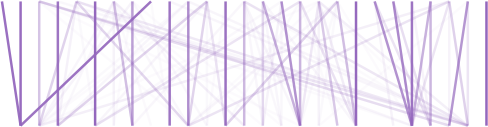
我的

意见

.

<EOS>

<pad>



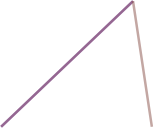




图4：两个注意头，也在第6层的第5层，显然涉及到回指解析。顶部：充分关注头部5。底部：将注意力与“its”这个词中的注意力头5和6分开。请注意，人们对这个词的关注非常尖锐。



他们的是

法律法律

将会吗

从来都没有过

是

完美的

,

但是

完美的

,

但是

它是它自己的人

应用程序和应用程序

应该应该

是

只是只是

- -

这个这个

是是

具体内容是什么？

我们，我们

是

找不到的

,

在

我的

意见

.

<EOS>

<pad>

是

找不到的

,

在

我的

意见

.

<EOS>

<pad>

这个

法

愿意

从不

是

完美的

,

但是

它的

申请

应该

是

刚才

-

这

是

什么

我们

是

找不到的

,

在

我的

意见

.

<EOS>

<pad>

这个

法

愿意

从不

是

完美的

,

但是

它的

申请

应该

是

刚才

-

这

是

什么

我们

是

找不到的

,

在

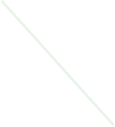
我的

意见

.

<EOS>

<pad>



|  |  |  |
| --- | --- | --- |
|  |  |  |





图5：许多注意力头表现出的行为似乎与句子的结构有关。我们给出了两个这样的例子，来自层6的编码器自我注意的两个不同的头部。这些头脑显然学会了执行不同的任务。