



上海交通大学
SHANGHAI JIAO TONG UNIVERSITY



学生创新中心
Student Innovation Center



BERT模型

学生创新中心：肖雄子彦



BERT

学习目标：

- **理解**BERT模型相比其他NLP模型的**优势**（真正的双向）
- **掌握**BERT编码器的两个**预训练任务**（MLM\NSP）
- **了解**BERT模型的应用方向和主要任务



BERT

Google AI团队新发布的BERT模型，在机器阅读理解顶级水平测试SQuAD1.1中表现出惊人的成绩：
在两个衡量指标上全面超越人类，并在11种不同NLP测试中创出最佳成绩。
BERT将为NLP带来里程碑式的改变，也是NLP领域近期最重要的进展。

SQuAD1.1 Leaderboard

Since the release of SQuAD1.0, the community has made rapid progress, with the best models now rivaling human performance on the task. Here are the ExactMatch (EM) and F1 scores evaluated on the test set of v1.1.

Rank	Model	EM	F1
	Human Performance <i>Stanford University</i> (Rajpurkar et al. '16)	82.304	91.221
1 Oct 05, 2018	BERT (ensemble) Google A.I.	87.433	93.160
2 Oct 05, 2018	BERT (single model) Google A.I.	85.083	91.835
2 Sep 09, 2018	nlnet (ensemble) Microsoft Research Asia	85.356	91.202
2 Sep 26, 2018	nlnet (ensemble) Microsoft Research Asia	85.954	91.677



Thang Luong
@lmthang

正在关注

A new era of NLP has just begun a few days ago: large pretraining models (Transformer 24 layers, 1024 dim, 16 heads) + massive compute is all you need. BERT from @GoogleAI: SOTA results on everything arxiv.org/abs/1810.04805. Results on SQuAD are just mind-blowing. Fun time ahead!

● 翻译推文

SQuAD1.1 Leaderboard

谷歌团队Thang Luong：BERT模型开启了NLP的新时代

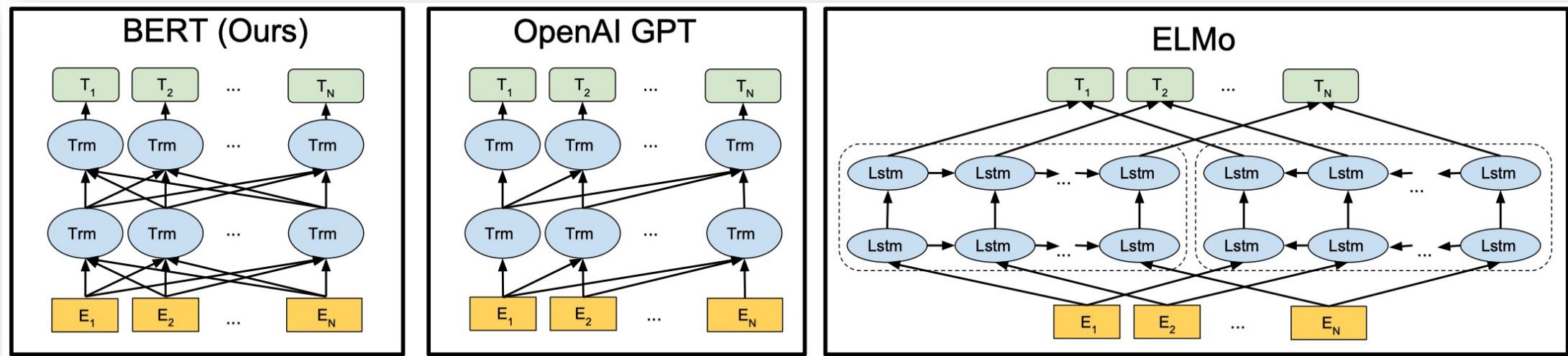


BERT

BERT: Bidirectional Encoder Representations from Transformers

(论文深入浅出，要言不烦，推荐阅读)

翻译过来就是双向Transformer编码表达，那什么是双向？



BERT, OpenAI GPT和ELMo之间的比较

- BERT使用双向的Transformer
- OpenAI GPT使用从左到右的Transformer
- ELMo使用经过独立训练的从左到右和从右到左LSTM的串联来生成下游任务的特征。
- 三个模型中，只有BERT表示在所有层中共同依赖于左右上下文

BERT

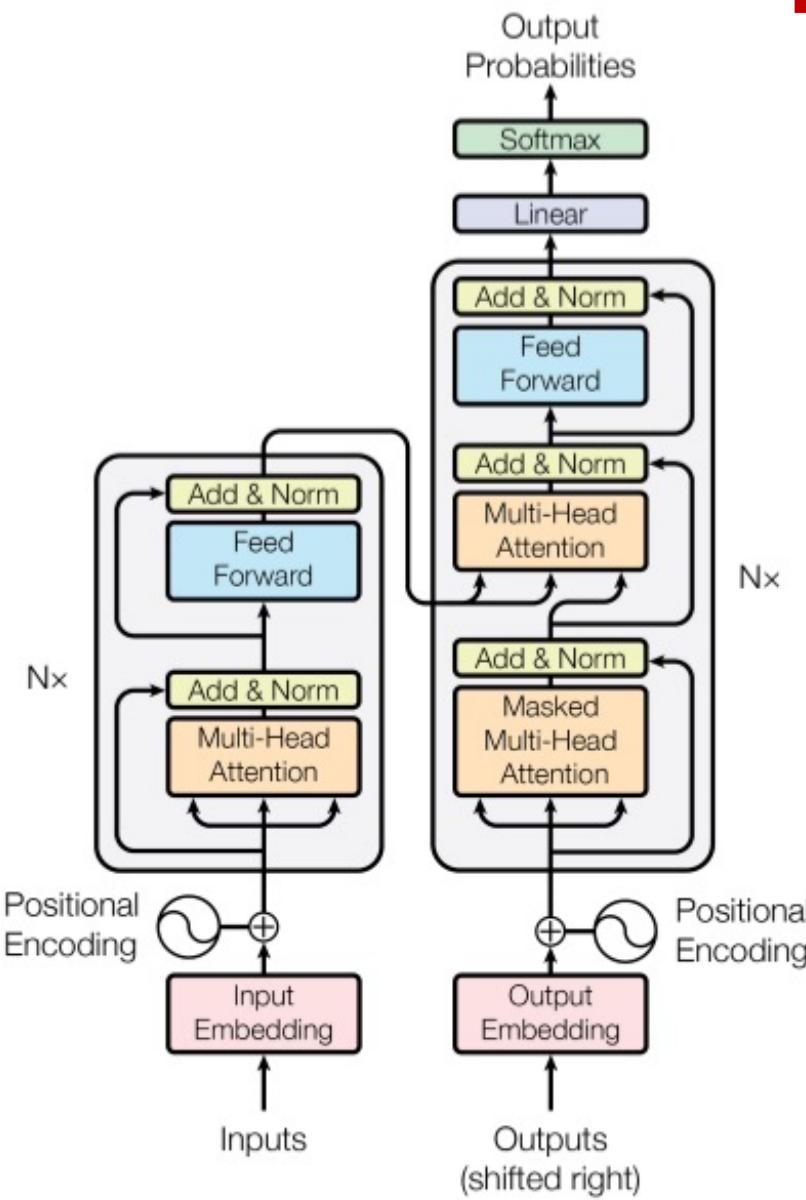
- ELMo采用的是双向的LSTM的架构，能够抓取到左右上下文的语义
- 但是GPT采用的是单向的注意力机制，导致了每个token只能关注左侧的语境，在文献中被称为“Transformer解码器”
- 而BERT采用了双向的双向的自注意机制，所以被称为“Transformer编码器”

Model	获取长距离语义信息程度	能否左右上下文语义	是否可以并行
Word2Vec	1	能	能
单向LSTM	2	不能	不能
ELMo	2	能	不能
GPT	3	不能	能
BERT	3	能	能

<https://www.jianshu.com/p/160c4800b9b5>



BERT

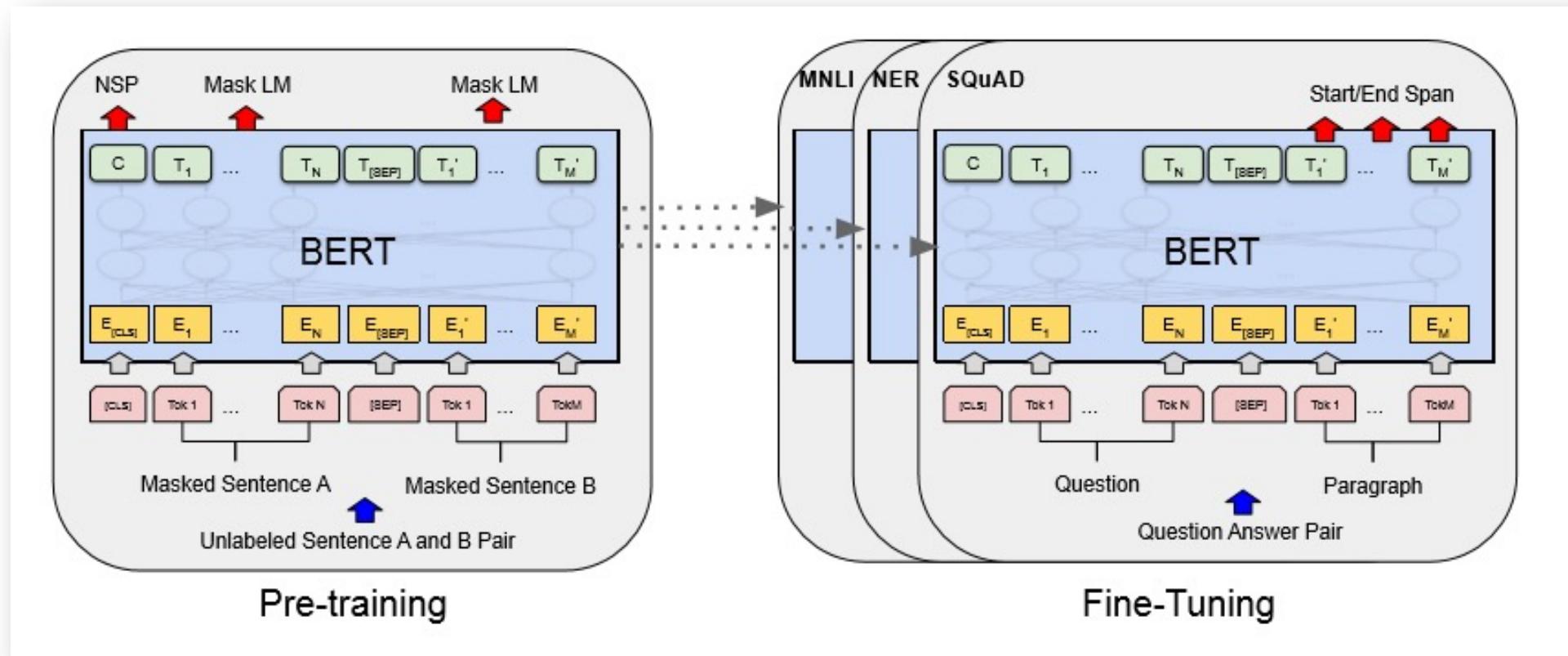


BERT 的五个关键词

- Pre-training
- Deep
- Bidirectional
- Transformer
- Language Understanding

作者：“存在通用的语言模型”
(language representation model)
先预训练通用模型
然后再应用到具体任务中

BERT



BERT通过理解所有层的上下文来预训练深度双向的语言表征方法。
因此，预训练的BERT表示可以通过一个额外的输出层进行微调（fine-tuning），
适用于广泛下游任务，比如问答任务和语言推理。

Corpus?

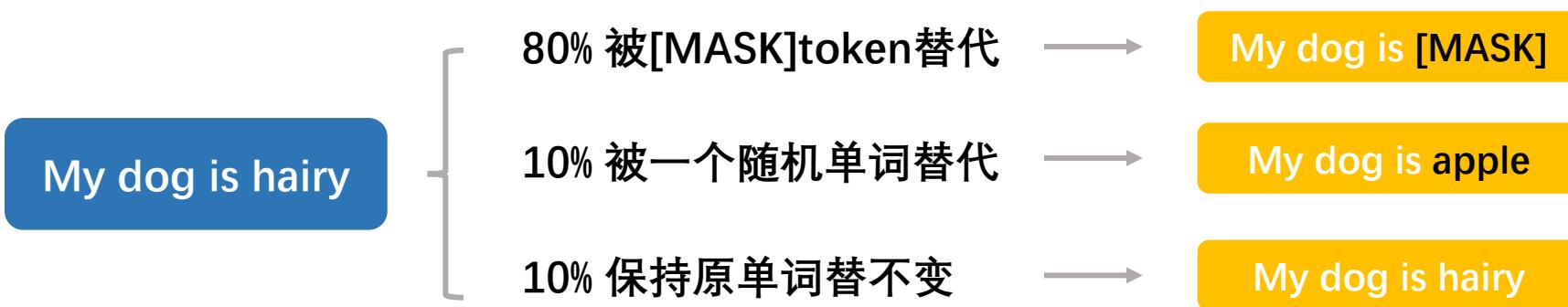
预训练目标之一：MLM

1. 遮蔽语言模型 (Masked Language Model, MLM)

随机遮盖或替换一句话里任意字词，然后让模型通过理解上下文、预测那个被遮盖或替换的部分，
(使得每个单词能够在多层上下文中间接的看到自己)

实际操作：随机把一句话中 15% 的 *token* 替换成以下内容：

- 1) 这些 *token* 有 80% 的几率被替换成 [mask]；
- 2) 有 10% 的几率被替换成任意一个其他的 *token*；
- 3) 有 10% 的几率原封不动.



预训练目标之一：MLM

- 之后让模型预测和还原被遮盖掉或替换掉的部分

模型最终输出的隐藏层的计算结果的维度是

$[batch_size, seq_len, embedding_dim]$

矩阵得到

$[batch_size, seq_len, vocab_size]$



我们在后面接一层用于分类，权重为 W_{vocab}

$[embedding_dim, vocab_size]$

我们用 W_{vocab} 完成隐藏维度到字向量数量的映射

在 $vocab_size$ 维做 softmax

得到模型的预测结果

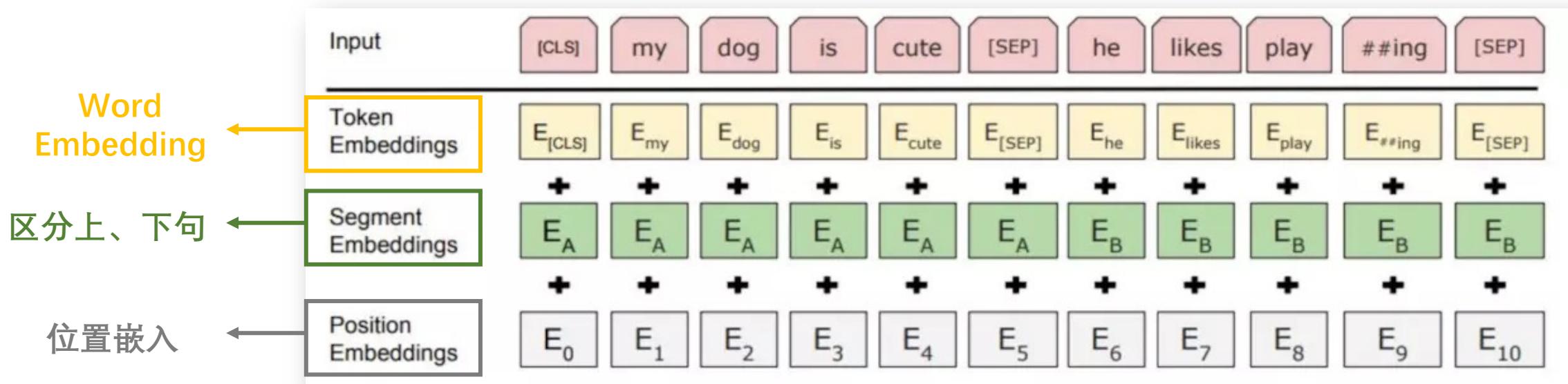
和 Label 做损失、反传梯度

预训练目标之二：NSP

2. 上下句预测 (Next Sentence Prediction, NSP)

首先我们拿到属于上下文的一对句子，也就是两个句子，之后我们要在这两段连续的句子里面加一些特殊 *token* : *[cls]* 上一句话, *[sep]* 下一句话. *[sep]*

也就是在句子开头加一个 *[cls]*，在两句话之中和句末加 *[sep]*，具体如下图：



- WordPiece 嵌入 (Wu et al., 2016) 30,000个token的词汇表
- Positional Embeddings, 最大支持序列长度为512

预训练目标之二：NSP

Is Next

[cls] The man goes to the shop [sep] He buys some fruits [sep]

Not Next

[cls] The man went to the shop [sep] My dog is cute [sep]

在实际的训练中，我们让上面两种情况出现的比例为 1:1，

然后，我们要初始化 *segment embeddings*

如：

给上句全 0 的 *token*，下句全 1 的 *token*，让模型得以判断上下句的起止位置

[cls] The man goes to the shop [sep] He buys some fruits [sep]

0 0 0 0 0 0 0 1 1 1 1

上面 0 和 1 就是 *segment embeddings* .



预训练目标之二：NSP

- 注意力机制让每句话中每个字对应的那条向量里，融入了这句话所有字的信息

那么，在最终隐藏层里，`[cls]token` 所对应的向量，里面也是含有整个句子信息的。

我们可以直接把这条向量拿出来，进行后续判断任务。

模型最终输出的隐藏层的维度是：

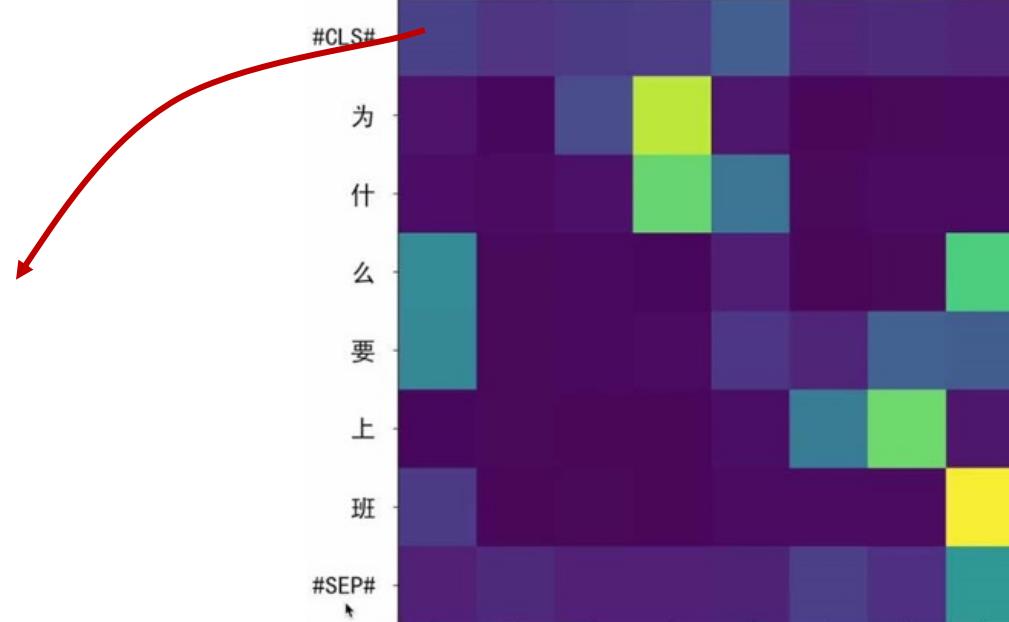
`[batch_size, seq_len, embedding_dim]`

我们要取出 `[cls]token` 所对应的那条向量
(`seq_len` 维度的第 0 条)

$cls_vector = X_{hidden}[:, 0, :]$

$y = \text{sigmoid}(\text{Linear}(cls_vector))$

完成从 `embedding_dim` 维度到 1 的映射，
再用 `sigmoid` 函数激活，从而进行二分类训练。



https://github.com/aespresso/a-journey-into-math-of-ml/tree/master/04_transformer_tutorial_2nd_part

BERT 模型的预训练

BERT将层数（即Transformer blocks）表示为L，将隐藏大小表示为H，将self-attention heads的数量表示为A。

在所有情况下，Feed-forward的大小设置为 $4H$ ，即 $H = 768$ 时为3072， $H = 1024$ 时为4096。

论文主要报告了两种模型大小的结果：

BERT Model	L : Transformer	H : Hidden units	A : Heads	Total Para
Base	12	768	12	110 M
Large	24	1024	16	340 M

需要充裕的
计算资源！

Google 开源了 BERT 的代码：

<https://github.com/google-research/bert>

大家可以下载在维基百科语料上使用 TPU 预训练好的模型（包括中文 BERT 预训练模型）

在具体下游任务中加入几个Dense层微调。

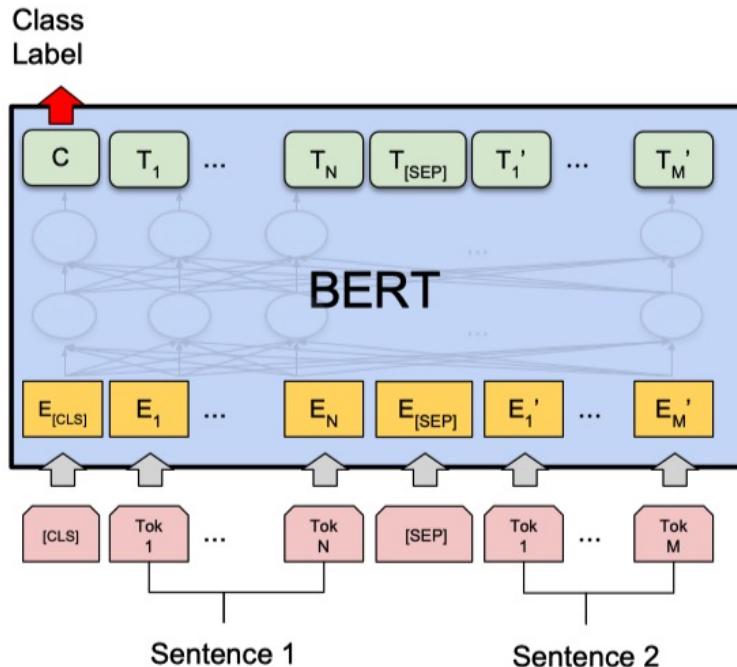


BERT 模型的思考

- 深度学习就是表征学习 (Deep learning is representation learning) "We show that pre-trained representations eliminate the needs of many heavily engineered task-specific architectures". 在11项BERT刷出新境界的任务中，大多只在预训练表征 (pre-trained representation) 微调 (fine-tuning) 的基础上加一个线性层作为输出。
- 规模很重要 (Scale matters) : “One of our core claims is that the deep bidirectionality of BERT, which is enabled by masked LM pre-training, is the single most important improvement of BERT compared to previous work” . 这种遮挡 (mask) 在语言模型上的应用对很多人来说已经不新鲜了，但确是BERT的作者在如此超大规模的数据+模型+算力的基础上验证了其强大的表征学习能力。这样的模型，甚至可以延伸到很多其他的领域。
- 预训练价值很大 (Pre-training is important) : “We believe that this is the first work to demonstrate that scaling to extreme model sizes also leads to large improvements on very small-scale tasks, provided that the model has been sufficiently pre-trained” . 预训练已经被广泛应用在各个领域了 (e.g. ImageNet for CV, Word2Vec in NLP) ，多是通过大模型大数据。BERT模型的预训练是用Transformer做的，可能换成LSTM或GRU可能也不会有太大性能上的差别（除了并行能力）。

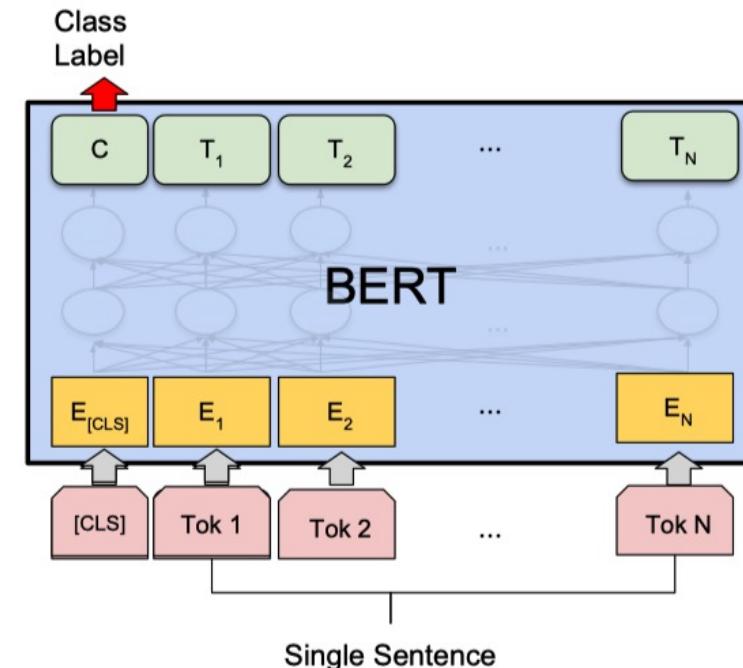


BERT 模型的应用



(a) Sentence Pair Classification Tasks:
MNLI, QQP, QNLI, STS-B, MRPC,
RTE, SWAG

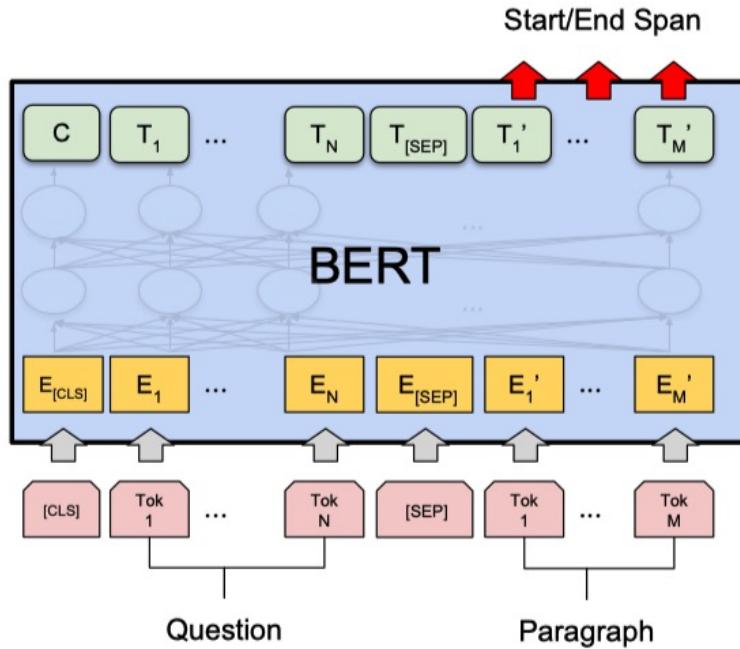
MNLI：预测第二句与第一句比是矛盾还是中立
QQP：Quara上提的两个问题是等价的
MRPC：写的注释是否和原文表达语义相等



(b) Single Sentence Classification Tasks:
SST-2, CoLA

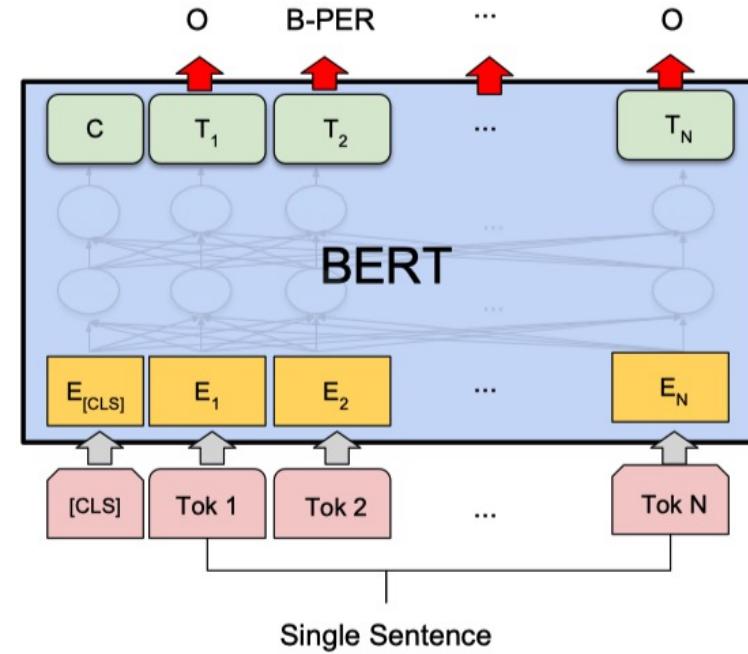
SST-2：电影评论中的情感分析
CoLA：预测英文句子在语言上是否“可接受”

BERT 模型的应用



(c) Question Answering Tasks:
SQuAD v1.1

SQuAD :
Stanford Question Answering Dataset
机器阅读理解数据集



(d) Single Sentence Tagging Tasks:
CoNLL-2003 NER

NER : 命名实体识别 (经典问题)
如从一句话中识别出人名、地名，
从电商搜索中识别出产品的名字…

• T h a n k s •

学生创新中心：肖雄子彥



上海交通大学
SHANGHAI JIAO TONG UNIVERSITY



学生创新中心
Student Innovation Center