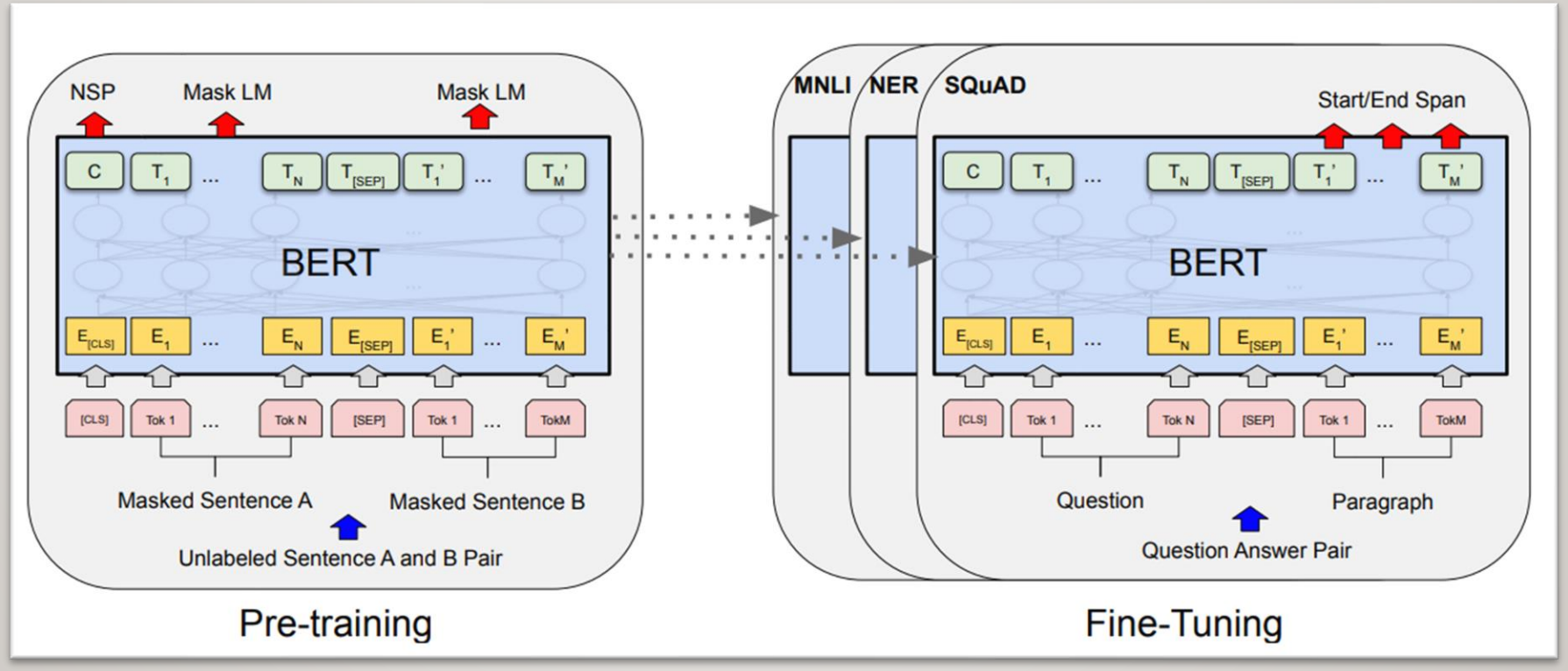
真-极度易懂的BERT

作者: 骰子AI

2022-4

BERT

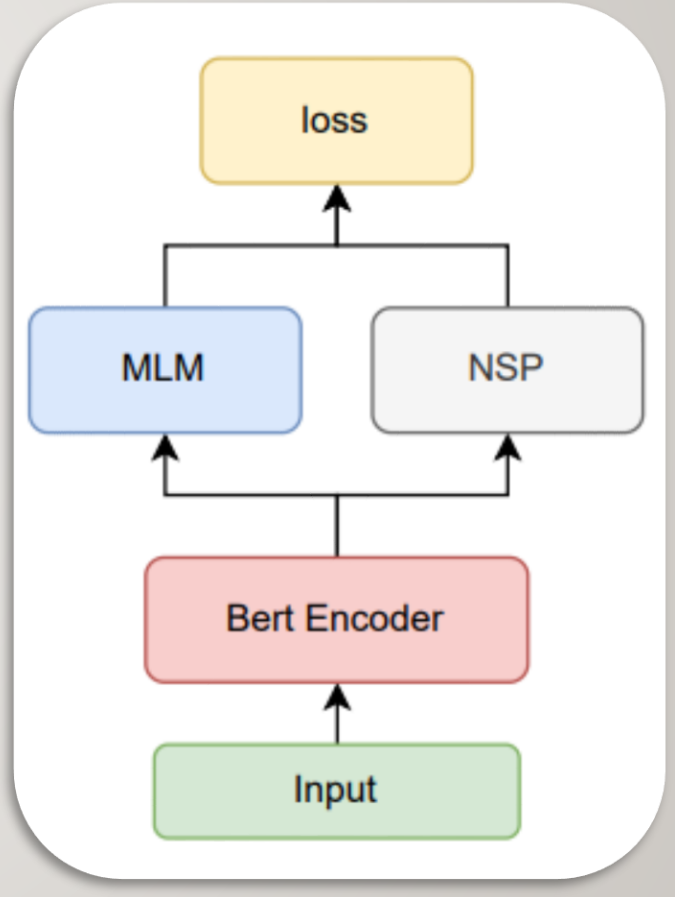
- 2019年Google AI 发布论文BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding。
- BERT分为 预训练(Pre-training)与微调(Fine-tune)。预训练简单来说就是通过两个任务联合训练得到Bert模型。而微调便是在预训练得到Bert模型基础上进行各种各样的NLP任务。



BERT预训练

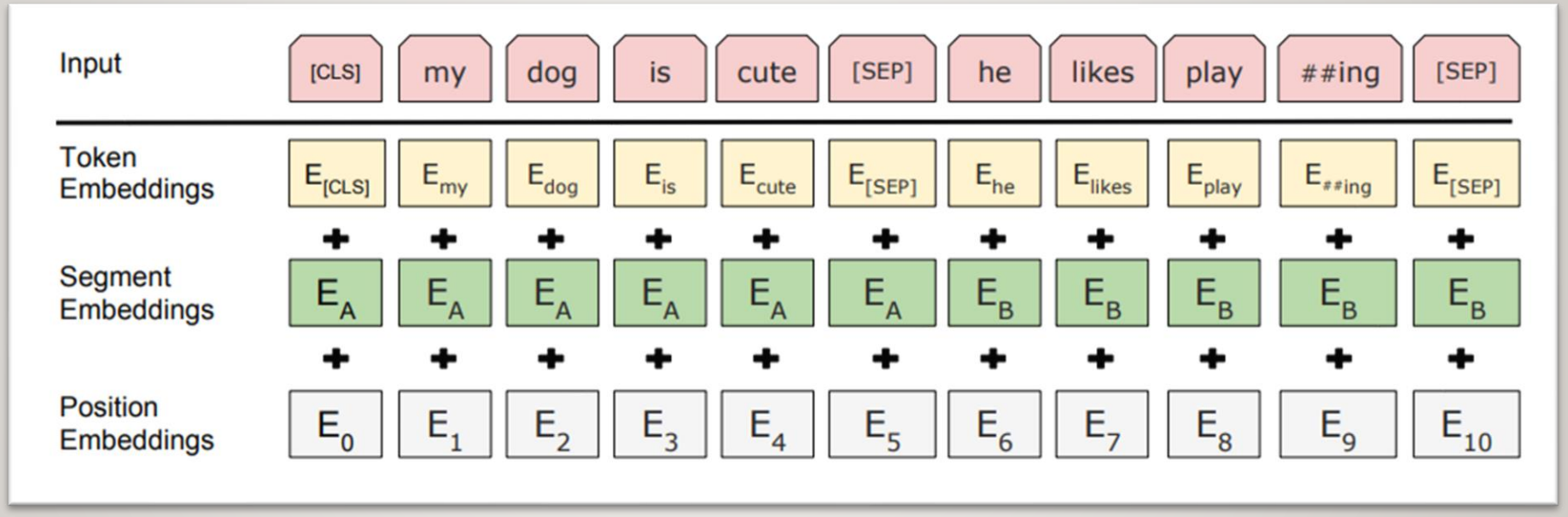
- 预训练大致的过程如右图所示。输入经Bert Encoder层编码后，进行MLM与NSP的任务，产生一个联合训练的损失函数从而迭代更新整个模型中的参数。

1. BERT Encoder: 采取默认12层的Transformer Encoder Layer对输入进行编码。
2. MLM：掩蔽语言模型(Masked Language Modeling)，遮盖句子中若干个词通过周围词去预测被遮盖的词。
3. NSP：下一个句子预测(Next Sentence Prediction)，判断句子B在文章是否属于句子A的下一个句子。



BERT ENCODER

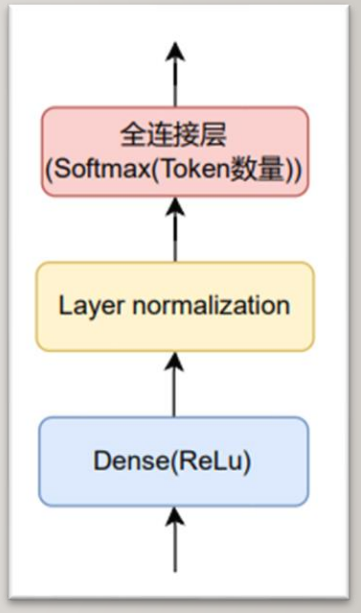
- 输入：
  1. 一对句子。例如 [“my”, “dog” , “is”, “cute”], [“he”, “likes”, “play”,”##ing”]。
  2. 给句首添加<CLS>符号，两句句子中间添加<SEP>符号，句末也添加<SEP>符号，组成一组输入。变为[“<CLS>”, “my”, “dog” , “is”, “cute”, “<SEP>”,“he”, “likes”, “play”,”##ing”, “ <SEP>”]。
  3. 在Embedding层将输入组合成三种Embedding的相加，分别是Token级别、句子级别、位置级别的Embedding。具体如下图所示：



- 中间：若干层的Transformer Encoder Layer, 默认为12。
- 输出：编码好的张量。形状为[Batch Size, Seq lens, Emb dim]

MLM

- 输入：Bert Encoder层的输出，需要预测的词元位置。
  - 需要预测的词元位置指的是[Batch Size,词元位置数量]的一个张量。
- 中间：一个MLP的结构。默认的形式如右图所示。
- 输出：序列类别分布张量。形状为[Batch Size, 需要预测的词元位置数量, 总Token数量]。



- 采样：
  - 每一对句子中随机选择15%位置的词语进行遮盖动作。
  - 遮盖时80%替换为”<mask>”, 10%替换为随机词元, 10%不变。过程中<CLS> 与 <SEP>不会被替换。
  - 例如[“<CLS>”, “my”, “dog” , “is”, “cute”, “<SEP>”,“he”, “likes”, “play”,”##ing”, “ <SEP>”]
    变为： [“<CLS>”, “my”, “<mask>” , “is”, “dog”, “<SEP>”,“he”, “likes”, “play”, “<mask>” ,“ <SEP>”]

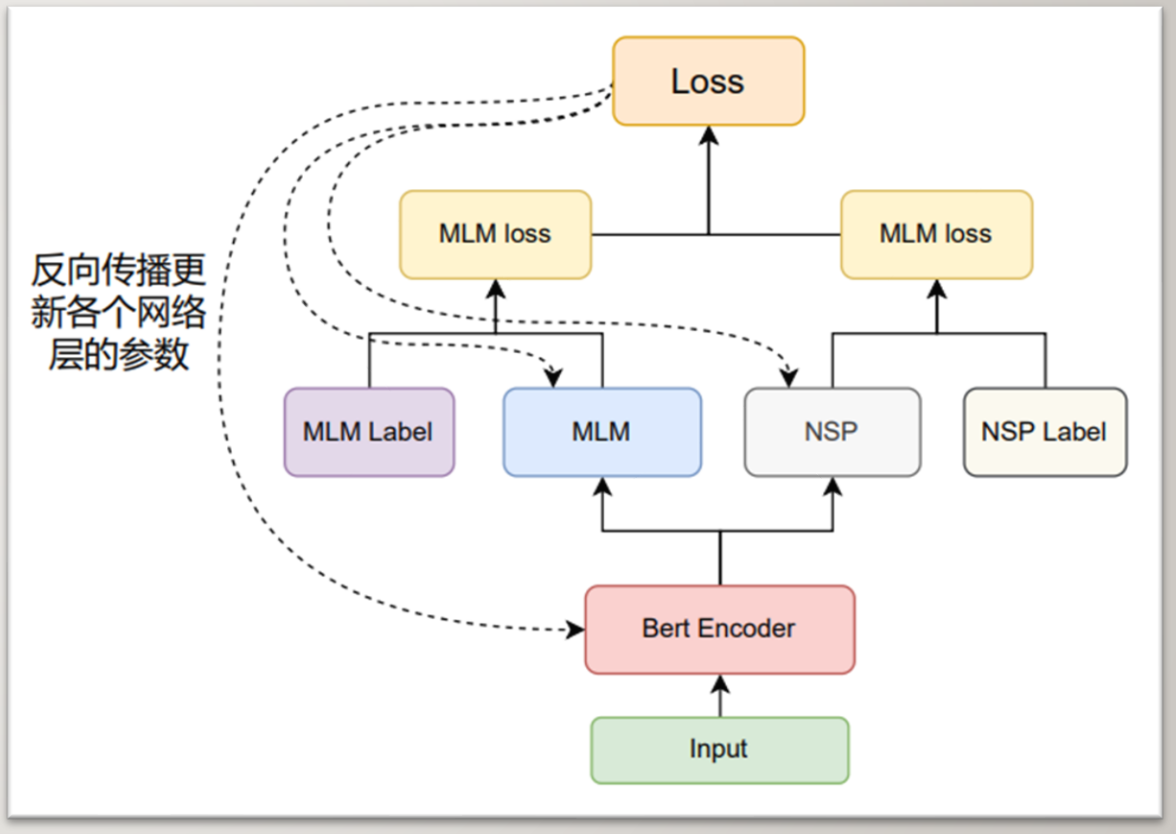
NSP

- 输入：Bert Encoder层输出<CLS>位置的张量(也就是序列中的首位)，形状为[Batch Size, Emd dim]。
- 中间：一个MLP的结构。默认是一个输出维度为2的，激活函数为Softmax的全连接层。
- 输出：类别分布张量。形状为[Batch Size, 2]。

- 采样：采样时，50%的概率将第二句句子随机替换为段落中的任意一句句子。

训练过程

1. 先通过文本数据整理出：(没有考虑padding)
   - ① 索引化的Tokens(已经随机选取了MLM与NSP任务数据)
   - ② 索引化的Segments
   - ③ MLM任务要预测的位置
   - ④ MLM任务的真实标注
   - ⑤ NSP任务的真实标注
2. 将①②输入Bert Encoder得到编码后的向量，以下称为⑥
3. 将⑥与③输入MLM网络得到预测值与④计算交叉熵损失函数。
4. 在⑥中取出<CLS>对应位置的张量输入NSP网络得到值与⑤计算交叉熵损失函数。
5. 将MLM的loss与NSP的loss相加得到总的loss，并反向传播更新所有模型参数。



FINE TUNE

- 微调主要是指通过预训练得到的BERT Encoder网络接上各种各样的下游网络进行不同的任务。

- 原论文中有4大类任务，如右图所示：

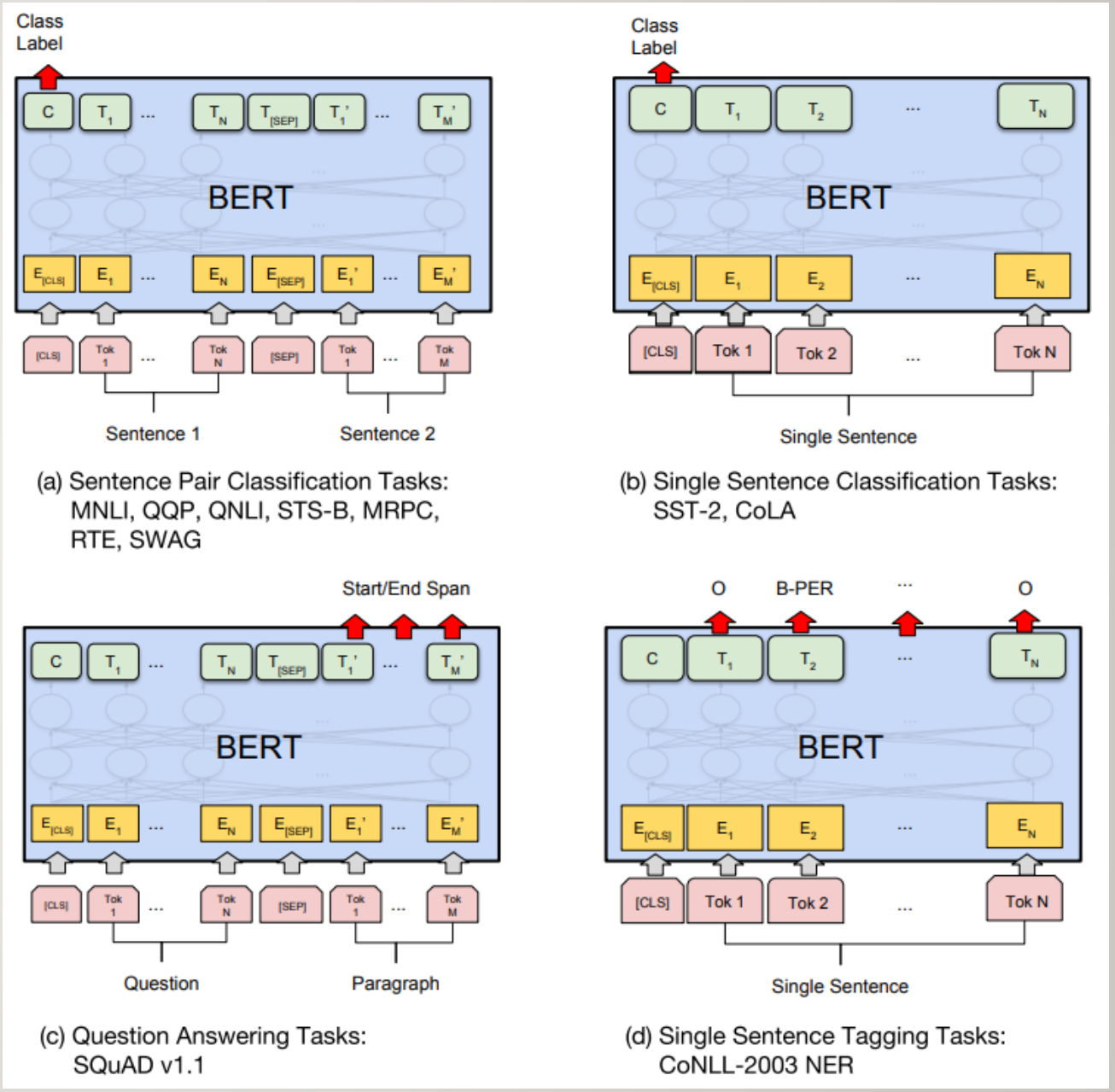
a) 句子对分类，将经过Encoder层编码后的<CLS>对应位置的向量输入进一个多分类的MLP网络中即可。

b) 单句分类，同上。

c) 根据问题得到答案，输入是一个问题与一段描述组成的句子对。将经过Encoder层编码后的每个词元对应位置的向量输入进3分类的MLP网络，而类别分别是Start(答案的首位)，End(答案的末尾)，Span(其他位置)。

d) 命名实体识别，将经过Encoder层编码后每个词元对应位置的向量输入进一个多分类的MLP网络中即可。

- 除原论文中给出的四大类任务，也可以结合实际场景设计更多的微调任务。



FAQ与总结

1. BERT名字中的Bidirectional 双向体现在哪？
2. BERT的位置编码与Transformer位置编码的区别是什么？
3. 为什么BERT的Word Embedding具备一词多义的信息？

- BERT使用的注意事项。