链滴

生动说明 Transformer，BERT，预训练模型的含义和关系

作者：aopstudio

原文链接：https://ld246.com/article/1667123460783

来源网站：链滴

很多知识，尽管在学会了之后发现原来是多么的简单，但是当你刚接触的时候则是完全一头雾水。

上一篇文章中我举了Java环境变量的配置例子来说明这一点，那是好几年前我读大一时候的事了。而近在自然语言处理知识的学习上，我又遇到了这种情况。

在我学习自然语言处理的入门教程时，很多教程都把Transformer和BERT连在一起讲，并且最后还加句“BERT实际上就是Transformer的编码器”，而且也不介绍除了BERT之外的其他预训练模型。这编排和说法导致我搞不清楚**Transformer和BERT到底是什么关系，预训练模型到底是个啥。**我一度以为是因为整个Transformer的效果不如只取它的编码器部分好，所以才提出了BERT。经过苦苦的搜和学习，终于理清了它们之间的关系。

Transformer和BERT到底是什么

Transformer应当是和CNN、RNN这些网络结构同级的一种新的网络结构，它的主要贡献是彻底抛了之前CNN、RNN等网络结构，而提出了只使用自注意力机制来搭建网络。这也是它的论文为什么“Attention is all you need”的原因，说的更直白点，这篇论文应该叫“You don't need CNN,RN,LSTM... Attention is all you need”。

BERT是一种预训练语言模型，它的主要贡献是提出了预训练的思想，即使用互联网中海量的文本数来对模型进行预训练，用户在使用时直接把预训练好的模型拿过来在具体的任务上进行微调训练就可达到不错的效果。

用学生学习的例子来解释神经网络学习的过程

我们假设不同的网络结构，如CNN，RNN，Transformer等就是一个个的学生。Transformer这个学就是一个天赋异禀的选手，它的大脑结构和其他学生的大不相同，因此它的学习潜力非常大。

自然语言处理的不同任务是不同的题型，而训练的过程就是教会神经网络怎么做题。不同的学生因为脑结构（网络结构）的不同导致它们的擅长的地方不一样。

在预训练的思想没有提出来之前，这些学生的学习模式是针对特定的任务使用对应的数据集进行训练比如翻译任务，我们给它一段原文和译文，让它自己去做题和对答案，它对完答案后就会不断模仿正答案去修改自己的思路（即网络参数）。在做过上个成千上万道题后，它就能回答个八九不离十了。

但是这种训练方式的问题也很明显，就是你练哪种题型，神经网络就只会哪种题型。比如一个在翻译务上效果非常好的网络，对于分类任务的效果说不定就很差。

在预训练的思想提出来之后，训练的方式进行了革命性的颠覆。现在我们不再只是教神经网络做某种定的题，而是教它们从源头上学会单词的意思以及语法内容。我们希望它们去“读书”，也就是去互网上的海量文本中进行学习。互联网上最好的预训练样本之一就是维基百科，它内容丰富、语句规范而且包罗万象。当然，就算对于同一本课本，不同的老师教法也可以不同。BERT就是其中一位老师它的教法是让学生们对课本内容做完形填空。比如一个句子：“Steve Jobs was the co-founder, chirman, and CEO of Apple.” 老师将其中的CEO这个单词遮住，变成 “Steve Jobs was the co-founer, chairman, and <mask> of Aple.” 学生需要在被遮住位置填出正确的单词。当学生对于绝大部分题目中被遮住的位置都能填上合的单词后，就认为学生已经学会了各个单词的含义了。另外，BERT还有一种教法是下一句预测（NS），即给出两个句子，需要学生判断第二个句子是不是第一个句子的后一句内容。

当然，有的同学就会提出质疑：“这种预训练的方式相比之前针对特定任务训练的方式不过是改了一题型而已，为什么就能说这种方式让神经网络真正理解了文本而前面一种只是做题机器呢？”在此，说一下我个人的观点。首先，预训练任务的样本通常不需要手工标注，因此其数量级可以非常大。像BRT使用了整个维基百科的文本来进行预训练，这是一个非常巨大的数量级，有句俗话叫“大力出奇迹，学了这么多东西，总能理解到点啥吧。其次，预训练任务的题目也不是随便选的。像BERT选的完

填空和下一句预测这两种题目就算用来给人类学习某种语言，是不是感觉也挺合适？而如果选翻译或分类，首先这两种任务无法自动构造出大量精确的数据集，需要人工构造，其次翻译任务主要学习的两种语言之间的对应关系，而分类任务只能学到一个句子属于哪一类，对于学习某种特定语言中的词和语法的效果很明显没有完形填空和下一句预测好。

不过，完形填空和下一句预测是BERT这位老资历的老师提出来的教法，它们也不一定就是效果最好。除了BERT之外，后起之秀RoBERTa,BART,UniLM等老师都提出了各自不同的教法，即训练方式，且都取得了超越BERT的效果。有兴趣的同学可以自行了解。

预训练的意义和争议

当BERT预训练完成后，我们就可以认为它已经具备了基本的语言理解能力。这时候针对具体的任务如文本分类，再用特定的数据集对BERT模型进行微调训练（即再微调一下模型的参数），就相当于于一个已经学会基本语义和语法的学生再教他做具体的题目，那么效果就会比直接用具体任务来训练得多。这就是预训练的意义所在。

不过这种预训练数据量极大，相对应需要的算力也就极大，这也是目前深度学习科研令人诟病的一个方，即用算力堆模型出论文。这样写出来的论文，从实验数据上来看非常漂亮，但已经不清楚到底是为模型真的好还是训练数据足够多。而且这种研究只有大公司才有能力做，相当于设置了一个学术壁。

BERT实际上就是Transformer的编码器

在理解了上述了内容之后，再来详细说明一下“BERT实际上就是Transformer的编码器”到底是怎么事。

BERT希望能够教出一个厉害的学生，于是首先它选了天资优越的Transformer作为学生，即使用了Transformer的网络结构作为预训练模型的基本框架结构。

Transformer使用了“编码器-解码器”结构，不精确地来说，我们可以把它看成一个学生的左脑和右。而BERT只把其中的左脑拿了过来，即只使用了Transformer的编码器部分作为自己的模型结构。至为什么这样，我个人估计有两种原因：一种原因是出于效果考虑，认为解码器部分不仅不会对学习有助，反而可能会干扰学习；另一种原因出于成本考虑，即如果同时训练两个部分，消耗的算力和时间本会比只训练一个部分大。

BERT在确定了模型结构是Transformer的编码器之后，再使用上述提到的完形填空和下一句预测这两预训练方式对模型进行预训练（实质就是调整模型的参数），最终得到一个优秀的模型。

总结

综上，我认为教程中应当把Transformer网络结构和BERT预训练的思想分开来看待，而不是安排在一，并且还加一句“BERT实际上就是Transformer的编码器”造成混淆。

实际上，预训练模型并不一定非得用Transformer这种网络结构，使用CNN、RNN等也是可行的。么为什么好像从来没有听说过基于CNN或RNN的预训练模型？这是因为Transformer这种网络结构天赋和潜能上就已经超越了CNN、RNN等，训练起来的效果也更好。所以目前的预训练模型基本都使用Transformer或其变种作为其网络结构。而如果未来某一天出现了一种全新的网络结构，它的天和潜能比Transformer更大，那么到时候所有的预训练模型，就都会使用这种网络结构来做预训练。