20190727

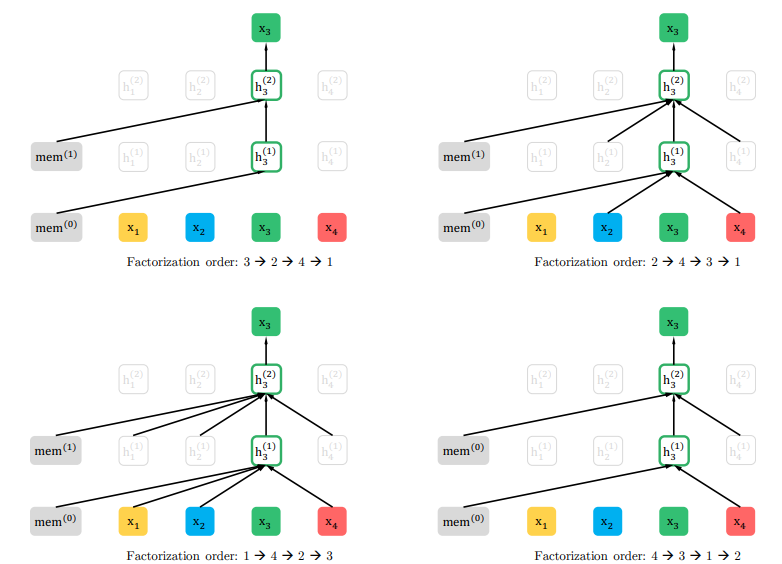
XLNET原理及应用

机器学习方法兴起后语言模型研究主要有两种无监督目标函数

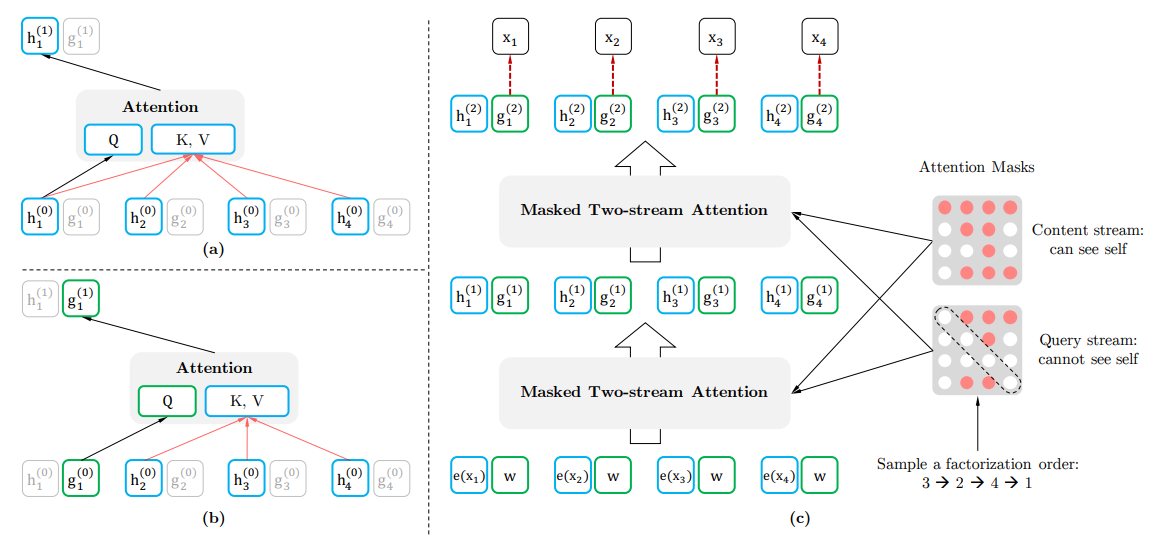
第一种是自回归函数，假设序列数据存在线性关系，一般是使用从左向右（单向）的语言模型方法，即通过RNN或是其他算法利用上下文预测前一个或者后一个的单词。这个阶段的开始是利用RNN,CNN结构进行预测，顶峰是ELMo模型，利用两个方向的LSTM隐状态拼接来进行前后文的预测。但ELMo的缺点在于拼接方式过于简单，效果不是很好

第二种是自编码机制，直接将输入复制到输出，比如bert就是利用在输入中随机mask掉一部分词，预训练时来预测mask的单词内容，这种方法可以比较好地同时预测上文和下文的内容。 但是从另一个角度看，因为预训练时mask掉了某些词，使得预训练和fine-tuning时数据不一致，同时因为mask有可能mask掉逻辑相关的词，使得预测变得难以确定。

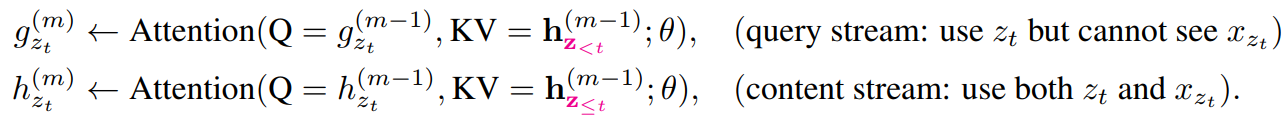
因此，研究者提出了XLNET算法。为了实现使用单向方法也可以兼顾上文和下文的内容，研究者抛弃了bert的mask方法，如果想预测x位置的词，则将除x位置之外的词位置打乱，这样处于x位置的词就可以在单向的模型中学习到自己前文和后文的词的内容，示意图如下：



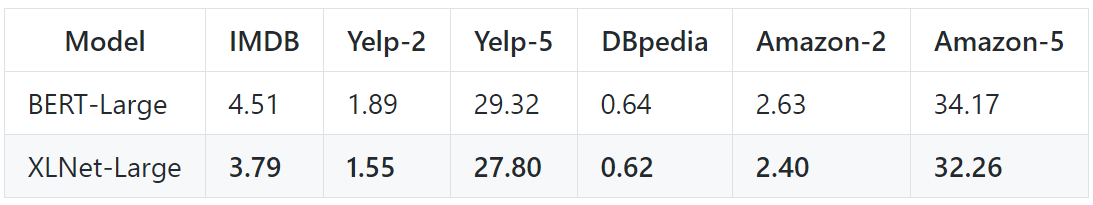
当然，在文本序列输入时，并没有立即改变文本中词的顺序，并且也不会选择所有的排列方式进行计算，因为n!的值实在太大，而是在后续计算时通过一种双流自注意力机制(two-stream self-attention)实现的，示意图如下：



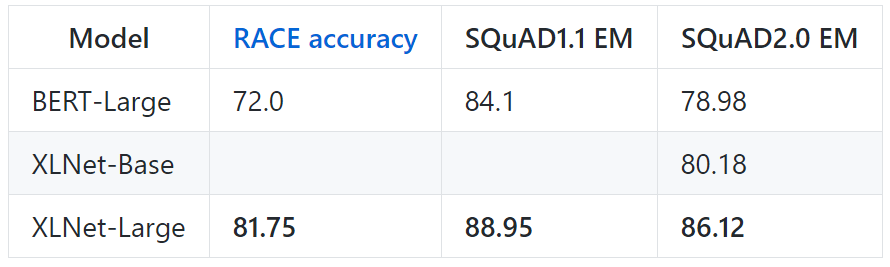
其中，a图表示一个内容流attention，也就是标准的self-attention，b图表示query流attention，这种attention不接受关于内容的访问信息，c图表示整个双流attention。其公式如下图：



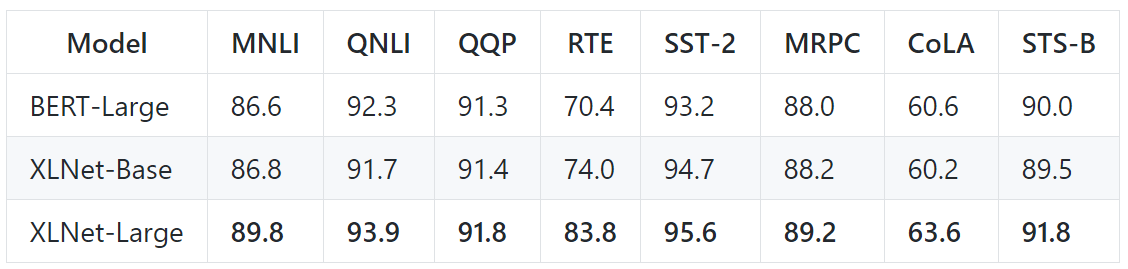
XLNET在文本分类上的错误率与bert比较的内容如下：



在阅读理解方面在SQuAD数据集上与bert的对比如下：



在GLUE自然语言理解任务上与bert的对比如下：



从XLNET得到的启示：

无论是BERT还是XLNET，都是从解决传统模型单向学习这一问题的角度出发来进行学习的。因此我想是否可以将文本进行倍增操作，比如将文本句子的后面拓宽成回文文本或倍增文本(如：西安交通大学→西安交通大学大通交安西；西安交通大学||西安交通大学)，再送入传统模型进行学习，这样文本的每一个词都可以在一次学习中关注到自己前方和后方的文本内容，但是又可以避免过大的开销。还是希望继续进行文献调研，来找到有意思的想法。