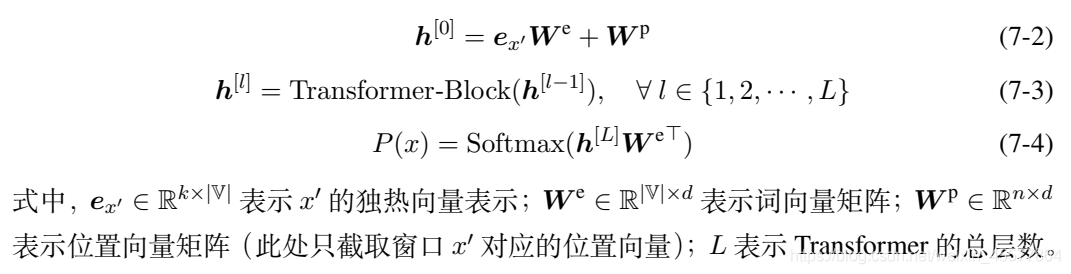
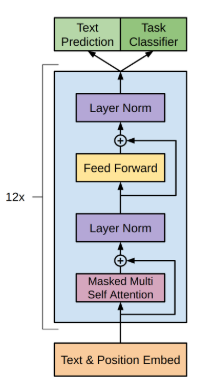
### GPT1

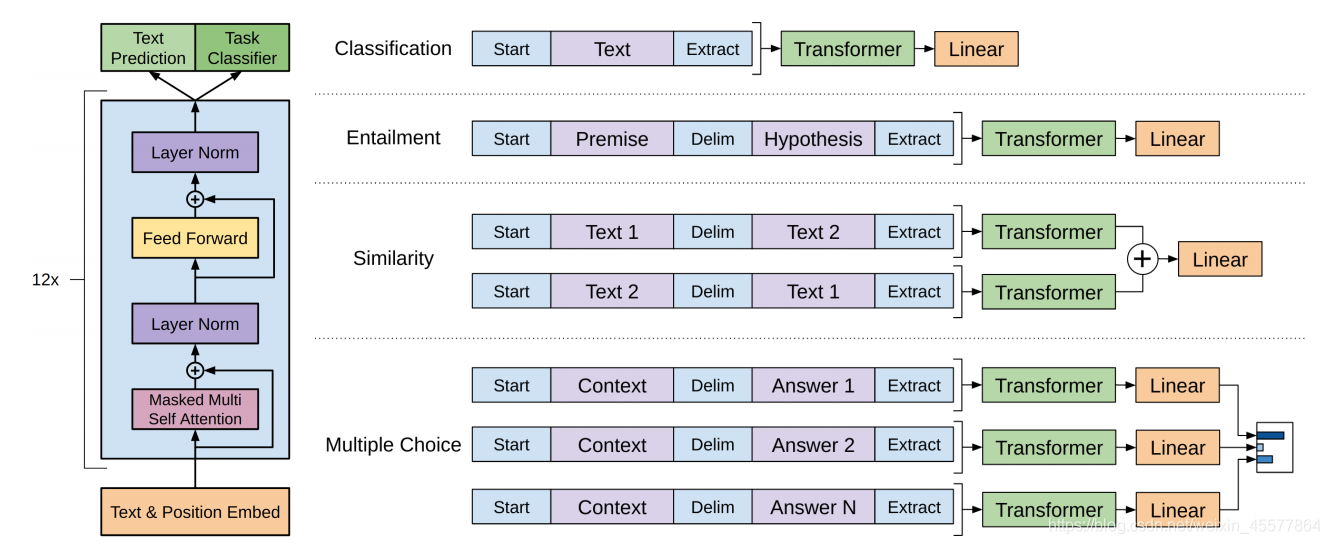
预训练：

GPT使用的是常规语言建模的方法，给定前面k个词，对下一个词做最大似然估计。k是超参数，表示窗口大小；表示神经网络参数。使用随机梯度下降优化似然函数。



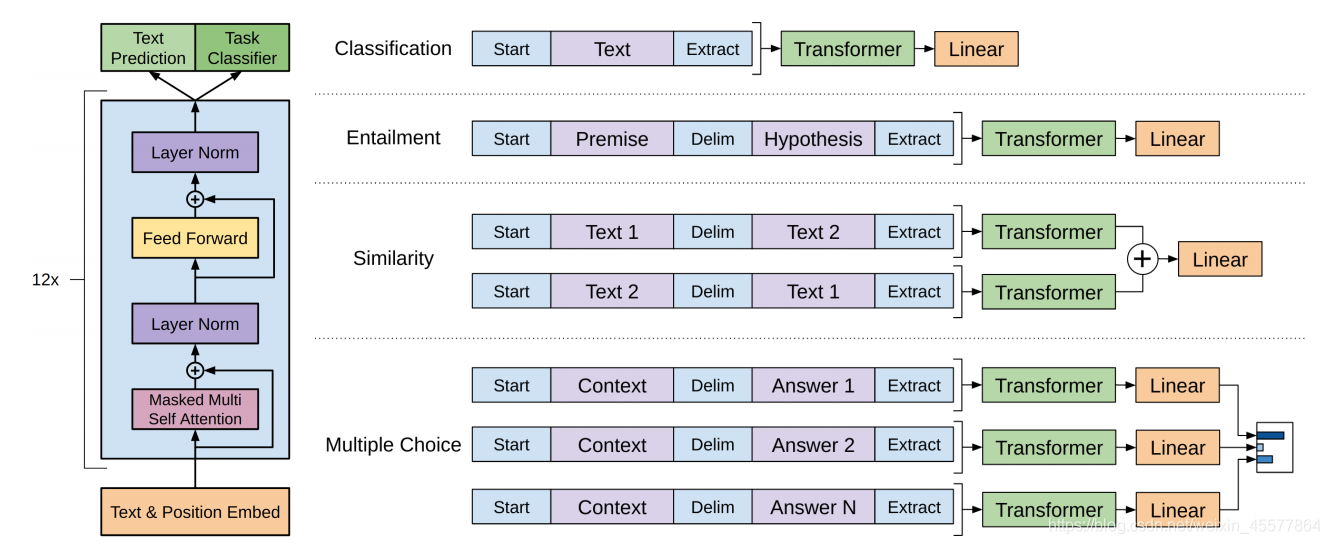


是Text&Position Embed层的输出，然后经过12个transformer的decoder块，得到下一个词的概率分布，以此计算，然后梯度反向传播。

fine-tuning：

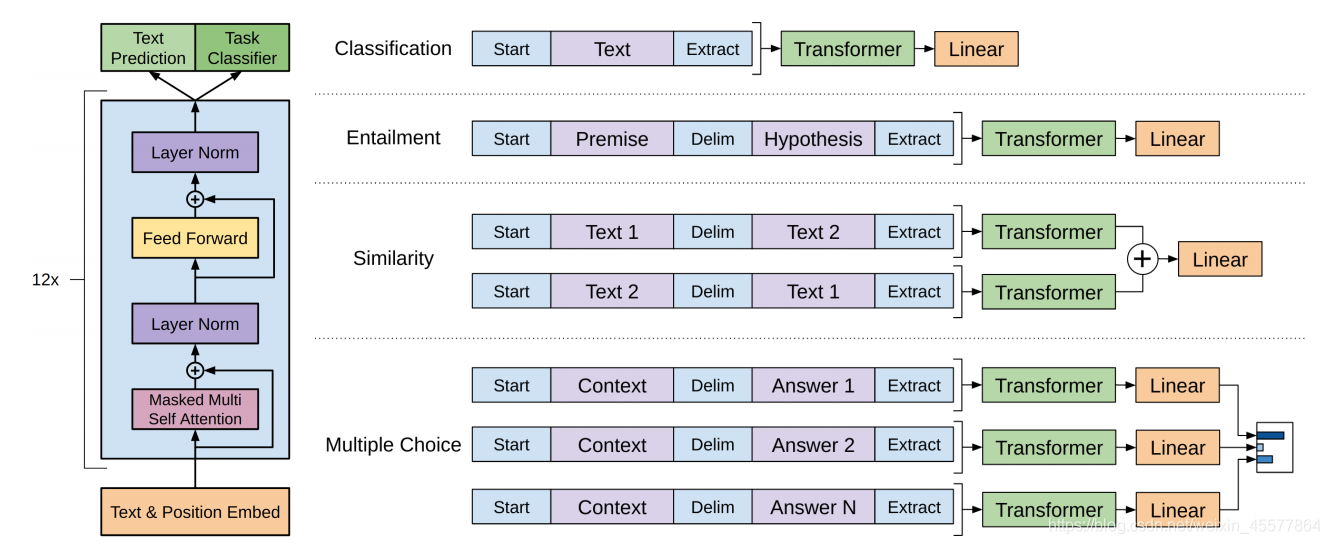
下游任务分为四类。

1. 分类



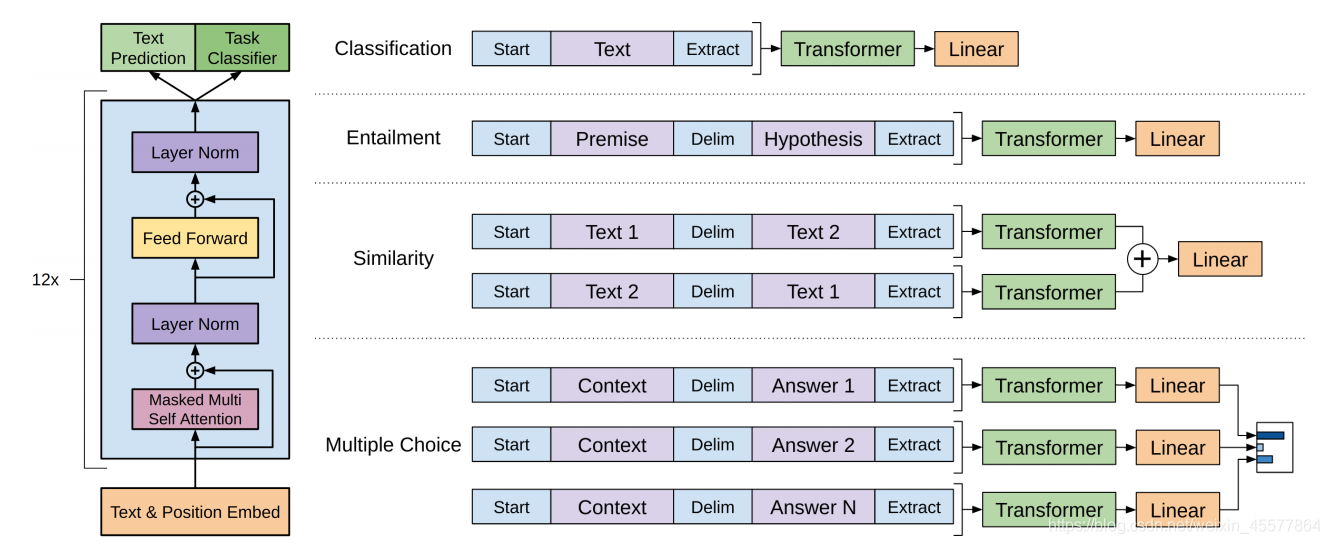
分类任务给每个文本添加初始词源start和抽取词源extract。将文本放入transformer对extract抽取特征。预训练时预测的是下一个词的概率分布，这里预测的是所属类的概率分布。在预训练时输出维度和词表一致，在分类任务中输出维度是类别数量。

1. 蕴含



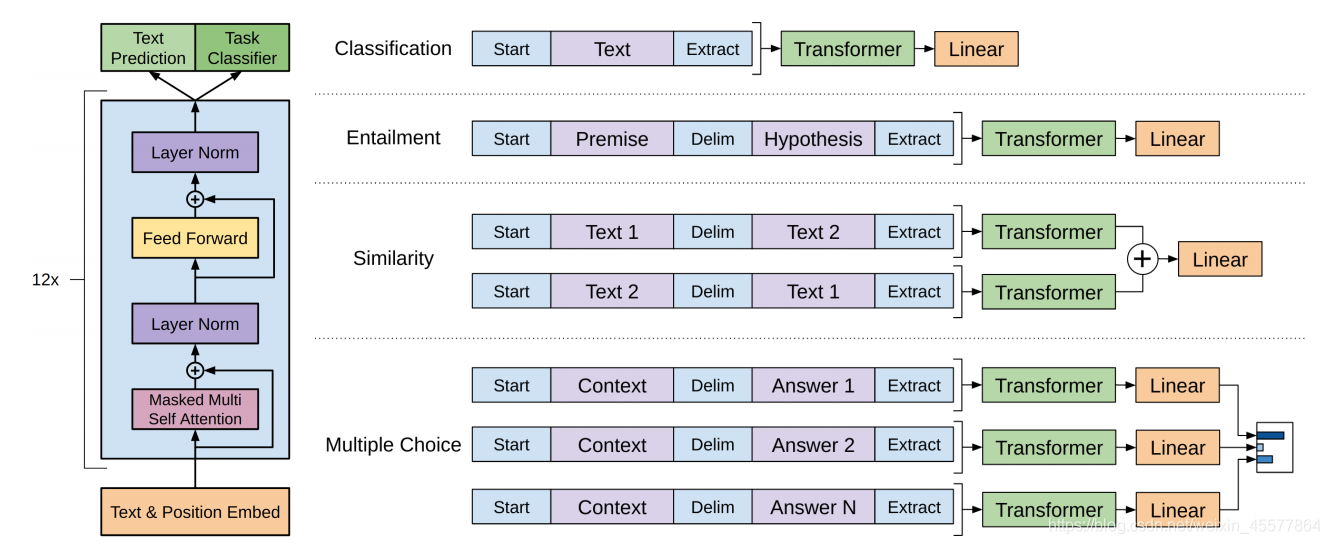
蕴含关系讨论的是句子premise能不能支持hypothesis。比方说premise是“汤姆追逐杰瑞”，hypothesis分别是“汤姆喜欢杰瑞”，“汤姆讨厌杰瑞”，“汤姆是杰瑞的邻居”，extract的特征分别应该是“不支持”，“支持”，“无关”，相当于一个三分类任务。

1. 相似



相似关系是一种对称关系，所以要做两次。Linear层的任务相当于做二分类。

1. 多选



多选问题将文本和选项连接作为输入。每个Linear的输出是一个数，表示答案对于文本的置信度。对所有置信度做softmax得到正确答案的概率分布。

上面的transformer是GPT训练的decoder。不论下游任务是什么，transformer结构不变。下游任务使用的损失函数是。是精调损失，是预训练任务损失。这是为了防止过渡擦除预训练学习到的通用知识，保留一定的通用性。

GPT2

GPT2的架构和GPT1相比没有太多变化。GPT2使用一个很大的数据集做了一个很大的模型，目标是zero-shot，即下游任务中的数据集无需额外标注。在GPT1的下游任务中，添加了Start，Extract，Delim三个额外标注。这三个额外标注是预训练时词库中没有出现的，需要在微调中学习。但是由于GPT2中微调任务无标注，不能学习到这些额外的词，所以需要把它们改成提示。比方说再英语到法语的翻译中可以改成translate to french，English text，French text。这需要模型足够大，能够理解提示符的意义。虽然随着模型的增大在下游任务中的性能确实逐渐变强，但是在很多任务中与最好的模型相比仍有很大差距。

GPT3

GPT3的目标是few-shot，可以有少量的标注。GPT3的模型相当大，因此在下游任务中不做任何梯度更新和微调。