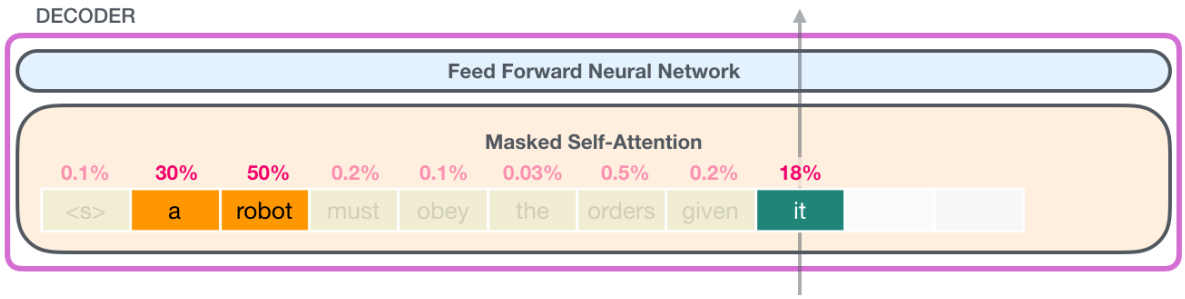
1. 自回归语言模型（Autoregressive LM, AR）

通常讲的"语言模型"(language model)其实就是根据上文内容预测下一个可能跟随的单词，也就是常说的自左向右的语言模型任务。或者反过来，就是根据下文预测前面的单词(比如ELMO使用了从左到右、从右到左的语言模型)，这种类型的language model被称为**自回归语言模型(AR)**。

ELMO尽管看上去利用了上文、也利用了下文，但是本质上仍然是自回归语言模型。的确，ELMO做了两个方向，但是是分别有两个方向的自回归语言模型，然后把LSTM的两个方向的隐节点状态*拼接*到一起，来体现双向语言模型这个事情的。（所以说，ELMO是"伪双向"，而BERT是"真双向"）。因为ELMO的这种融合模式过于简单，所以效果其实并不是太好。

1.1 GPT

GPT也是典型的自回归语言模型。GPT其实就是Transformer的**decoder**部分（其实和transformer的decoder还有些不一样）经过大规模数据预训练之后得到的模型。GPT的"单向"源自于它在Transformer的decoder中用**masked** self-attention来遮挡住了当前词后面的那些词，防止在预测下一个词的时候"看到"下一个词是什么。例如下图中的"it"只能对其之前的token来分配注意力权重。



GPT的总体结构：

GPT是Transformer Decoder稍加修改、并配以特定的下游模型得到的。"稍加修改"指的是，把用于引入encoder输入的Multi-head Self-Attention砍掉，只保留**Masked** Multi-self Attention和FFN（其实也可以理解为GPT用的是Transformer Encoder，只不过把Multi-head Self Attention 换成了**Masked** Multi-head Self-attention）；"特定下游模型"指的是线性变换+分类这样的简单结构。

Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

无监督训练

GPT的无监督预训练是基于语言模型的，给定一个无标签的序列

$$\mathcal{U} = \{u_1, \cdots, u_n\}$$

，语言模型的优化目标是最大化下面的似然值：

$$L_1(\mathcal{U}) = \sum_i \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta) \tag{1}$$

其中，网络最后一层的输出概率分布为

$$h_0 = UW_e + W_p$$
$$h_l = \texttt{transformer\_block}(h_{l-1}) \forall i \in [1, n]$$
$$P(u) = \texttt{softmax}(h_n W_e^T)$$

有监督微调

$$P(y | x^1, \ldots, x^m) = \texttt{softmax}(h_l^m W_y).$$

用预测值和真实的y值计算交叉熵损失

$$L_2$$

同时，加入语言模型的loss $L_1(C)$作为**auxiliary loss**有助于**①增强泛化能力、同时②提升收敛的速度。**因此完整的损失函数为：

$$L_3(C) = L_2(C) + \lambda L_1(C)$$

1.2 自回归语言模型的优缺点

- 缺点：只能利用上文或者下文的信息，不能同时利用上文和下文的信息
- 优点：对于生成类NLP任务，比如文本摘要、机器翻译等，在实际生成内容的时候，就是从左向右的，自回归语言模型天然匹配这个过程。而Bert这种DAE模式，在生成类NLP任务中，就面临训练过程和应用过程不一致的问题，导致生成类的NLP任务到目前为止都做不太好。

之后会讲的TransformerXL and XLNet也都是自回归语言模型。

2. 自编码语言模型 (Autoencoder LM, AE)

相比AR而言，Bert通过在输入X中随机Mask掉一部分单词，然后预训练过程的主要任务之一是根据上下文单词来预测这些被Mask掉的单词，这是典型的**Denoising Autoencoder(DAE)**的思路。那些被Mask掉的单词就是在输入侧加入的所谓噪音。类似Bert这种预训练模式，被称为DAE LM。

这种DAE LM的优缺点正好和自回归LM反过来，它能比较自然地融入双向语言模型，同时看到被预测单词的上文和下文。对于很多NLP任务而言，典型的比如阅读理解，在解决问题的时候，是能够同时看到上文和下文的，所以当然应该把下文利用起来。在Bert原始论文中，与GPT1.0的实验对比分析也可以看出来，BERT相对GPT 1.0的性能提升，主要来自于双向语言模型与单向语言模型的差异。

缺点是: 在输入侧引入[Mask]标记，导致预训练阶段和Finetuning不一致的问题，因为Fine-tuning阶段是没有标记的。

大师兄：词向量之GPT-1，GPT-2和GPT-3

张俊林：XLNet:运行机制及和Bert的异同比较