第一种范式 GPT：因果模型/从左至右的Transformer模型

第二种范式 BERT：双向Transformer编码器，使用**掩蔽语言模型**训练，让模型同时看到左边和右边的所有文本

fine-tuning：在BERT的输出后接入一个神经网络分类器来完成命名实体识别，问答等下游任务。

- 从直觉上看，在预训练阶段学习一种语言模型，该模型实例化了丰富的单词意义表示，从而使模型**更容易学习**（“被微调到”）下游语言理解任务的需求。
- **预训练-微调范式**是机器学习中所谓的**迁移学习**的一个实例：从一个任务或领域获取知识，然后将其应用（迁移）来解决一个**新任务**的方法

11.1 Bidirectional Transformer Encoders

上下文嵌入：上下文中的单词的表示。

静态嵌入：word2vec或GloVe，学习了对词汇表中每个唯一的单词w的单一向量嵌入。

对于上下文嵌入，如通过BERT等掩码语言模型学习到的嵌入，每个单词w每次在不同的上下文中出现时，都会用不同的向量表示。虽然第10章的因果语言模型也使用了上下文嵌入，但由**掩蔽语言模型创建的嵌入却特别有用**

GPT用于上下文生成、总结和机器翻译很有效，但用于序列分类和标记问题有明显缺点。如果我们想赋予准确的标签，我们需要考虑右边的上下文，而不只是左边。

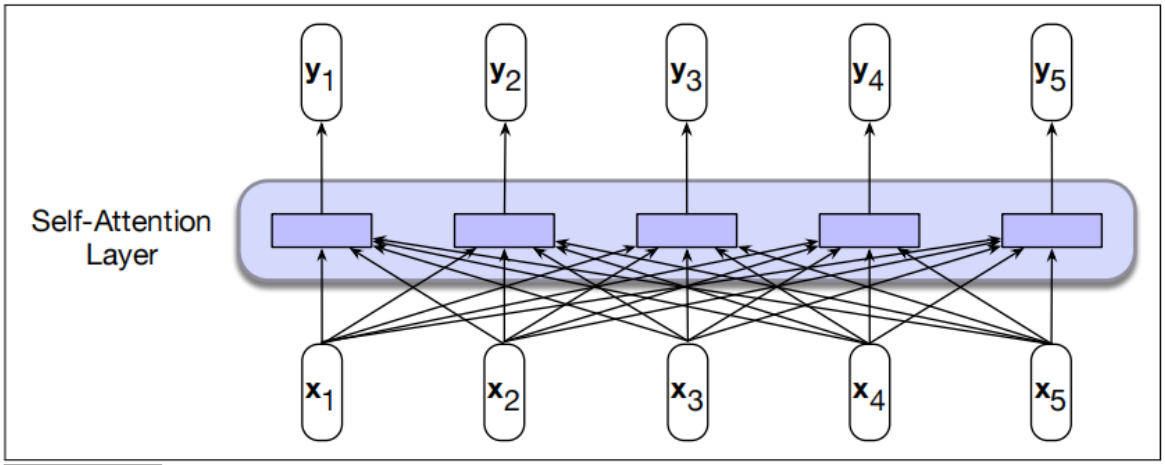

Figure 11.2 Information flow in a bidirectional self-attention model. In processing each element of the sequence, the model attends to all inputs, both before and after the current one.

双向编码器使用自注意力来将输入嵌入序列$(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)$映射到相同长度的输出嵌入序列$(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n)$，其中输出向量已经使用来自整个输入序列的信息进行上下文化。

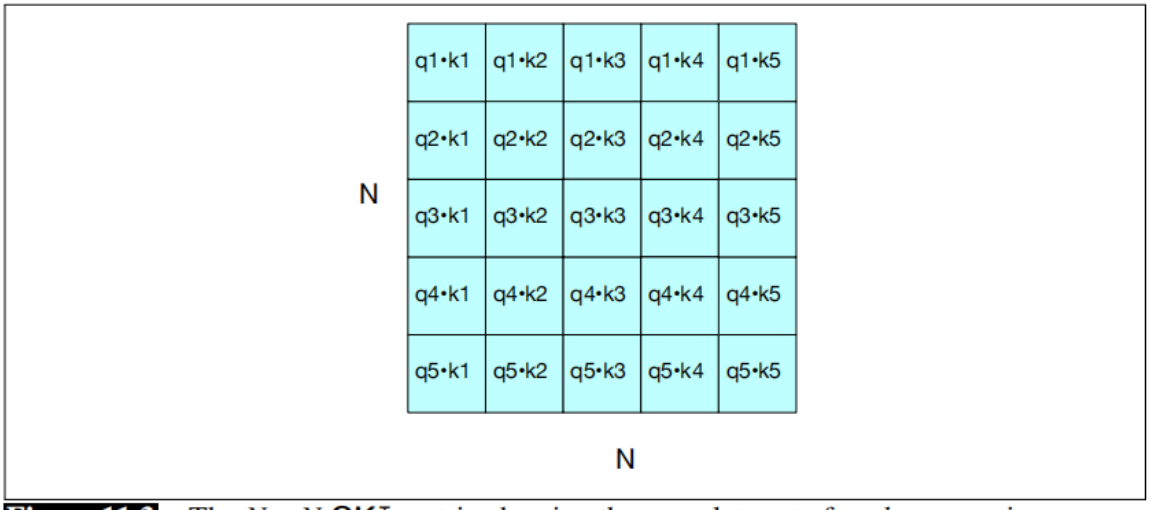

Figure 11.3 The $N \times N$ $\mathbf{QK}^\intercal$ matrix showing the complete set of $q_i \cdot k_j$ comparisons.

子词分词和位置编码得到输入编码

BERT细节：

- 一个由使用WordPipece算法生成的30,000个token组成的子单词词汇表
- 768大小隐藏层
- 12层Transformer块，每个层有12个多头注意层。
- 100M参数
- 输入长度会带来二次方的时间空间开销，需要设置恒定输入长度，BERT设置为512

11.2 Training Bidirectional Encoders

预测下一个词任务显得太平常，因为已经可以看到全文了。

提出新的任务：填空任务cloze task

也就是说，给定一个缺少一个或多个元素的输入序列，学习任务是**预测缺失的元素。**更准确地说，在训练过程中，模型**被剥夺了输入序列中的一个或多个元素**，并且必须为每个缺失项的词汇表生成一个概率分布。然后，我们使用每个**模型预测中的交叉熵损失**来驱动学习过程

可以推广至破坏输入后让模型还原输入的方法，可以使用的包括：掩蔽，替换，重新排序，删除，外部插入

Masking Words

Masked Language Modeling：MLM模型从训练语料库中提取一系列句子，其中从每个训练序列中随机选择样本（即一些词语）用于学习任务。一旦被选中，样本将以三种方式之一使用：

- 被替换为token[MASK]（填空）
- 被词汇表中的另一个token取代，基于token的一元语法概率随机抽样（纠错）
- 保持不变（还原）

15%的输入token被选出。这些token中80%被用来掩蔽，10%替换，10%不变

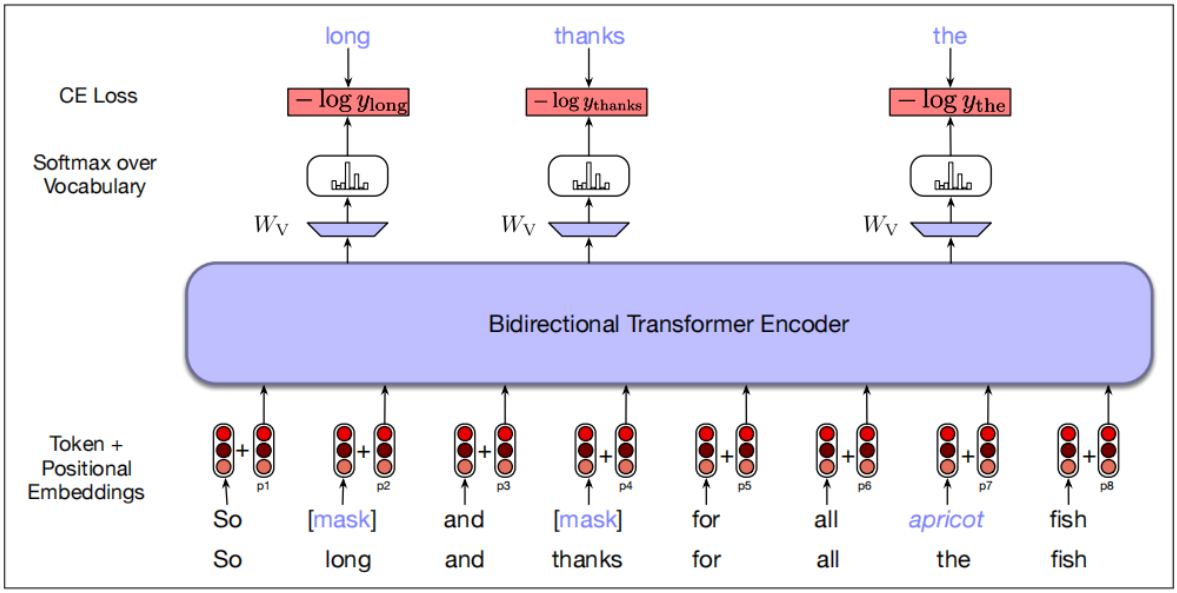

Figure 11.5 Masked language model training. In this example, three of the input tokens are selected, two of which are masked and the third is replaced with an unrelated word. The probabilities assigned by the model to these three items are used as the training loss. (In this and subsequent figures we display the input as words rather than subword tokens; the reader should keep in mind that BERT and similar models actually use subword tokens instead.)

某一个batch所有的采样token损失求平均

Masking Spans

问题回答、句法解析、共指关系和语义角色标记应用都涉及到成分或短语的识别和分类。

一个间隔span指选取的一个或多个连续词语序列。在SpanBERT中，间隔长度从一个几何分布中随机采样，最多为10，倾向于较短长度。给定间隔长度，起止位置被均匀采样

按照BERT方法确定间隔所有token的替换方式并进行替换。

Span Boundary Objective(SBO)：学习边界的token表示，将间隔的前一个单词和后一个单词的输出拼接上某一个间隔内的位置编码输入一个FFN。

$$
\begin{aligned}
s &= \text{FFN}([y_{s-1}; y_{e+1}; p_{i-s+1}]) \\
z &= \text{softmax}(Es)
\end{aligned}
$$

$$
\begin{aligned}
L(x) &= L_{MLM}(x) + L_{SBO}(x) \\
L_{SBO}(x) &= -logP(x|x_s, x_e, p_x)
\end{aligned}
$$

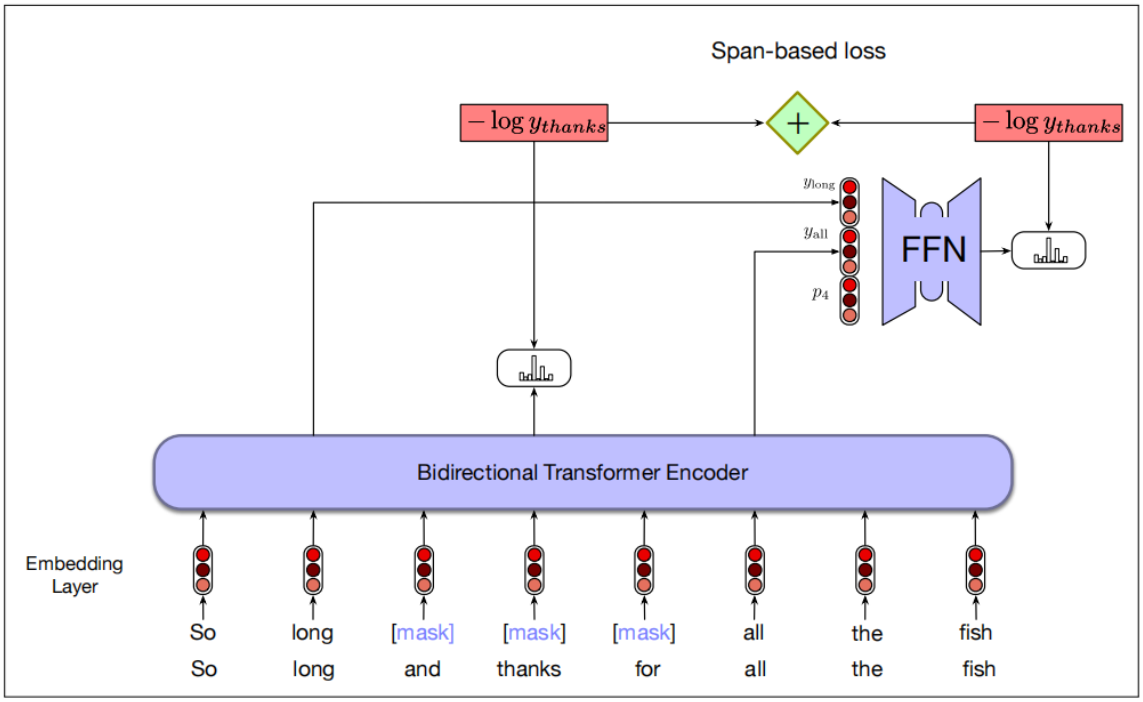

Figure 11.6 Span-based language model training. In this example, a span of length 3 is selected for training and all of the words in the span are masked. The figure illustrates the loss computed for word *thanks*; the loss for the entire span is based on the loss for all three of the words in the span.

Next Sentence Prediction

有些任务需要检测句子对之间的关系，比如paraphrase detection（判断两个句子语义是否一致），entailment（两个句子的说法是否相互支持或矛盾），discourse coherence（两个句子是否是连贯的两句话）

下一句预测，50%正例，50%负例，二分类问题。

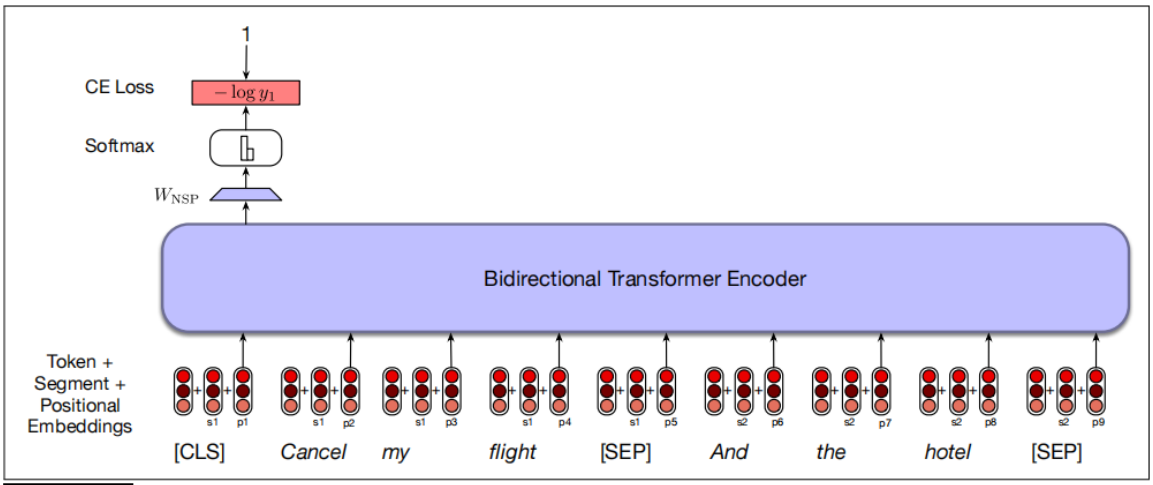

Figure 11.7 An example of the NSP loss calculation.

Training Regimes

早期模型使用BooksCorpus—0.8B词语的书籍文本语料库，2.5B词语的英语维基百科文本语料，总共3.3B词语。

最先进的模型使用了大几个数量级的语料

句子对从语料库中收集，使用1：1策略。对句子对组合的新句子进行掩蔽并计算MLM和NSP任务。模型需要40个epoch达到收敛

这个预训练过程包括学习到的**单词嵌入**以及**双向编码器的参数**

Contextual Embeddings

给定一个模型和一个新句子，模型的输出由每个词语上下文嵌入构成，可以为需要词义的任务提供上下文表征

上下文编码可以代表上下文中某个token的意义。对于输入x_i，我们可以直接使用$\boldsymbol{y}_i$作为词义表征，也可以用模型的最后四层的$\boldsymbol{y}_i$的平均作为词义表征

静态嵌入表示**单词类型（词汇项）的含义**，**上下文嵌入**表示**单词标记token**的含义：特定上下文中**特定单词类型的实例**。因此，上下文嵌入可以用于测量**上下文中两个单词的语义相似度**等任务，并且在需要单词意义模型的语言任务中很有用。

11.3 Transfer Learning through Fine-Tuning

预训练语言模型的优势在于可以从海量文本中获得泛化能力

微调通过预训练模型上加入少量应用相关的参数用于特定的应用程序。一般来说，微调会冻住或仅仅修改少量预训练模型参数

Sequence Classification

RNN使用最后一个输入元素的隐藏层作为最终分类

BERT中，[CLS]标记作为句子编码，在输入句子开头加入。最后一层的[CLS]输出向量代表整个输入句子并作为**分类头**（一个logistic回归或一个神经网络分类器）的输入

$$\mathbf{y} = \text{softmax}(\mathbf{W_C y_{CLS}})$$

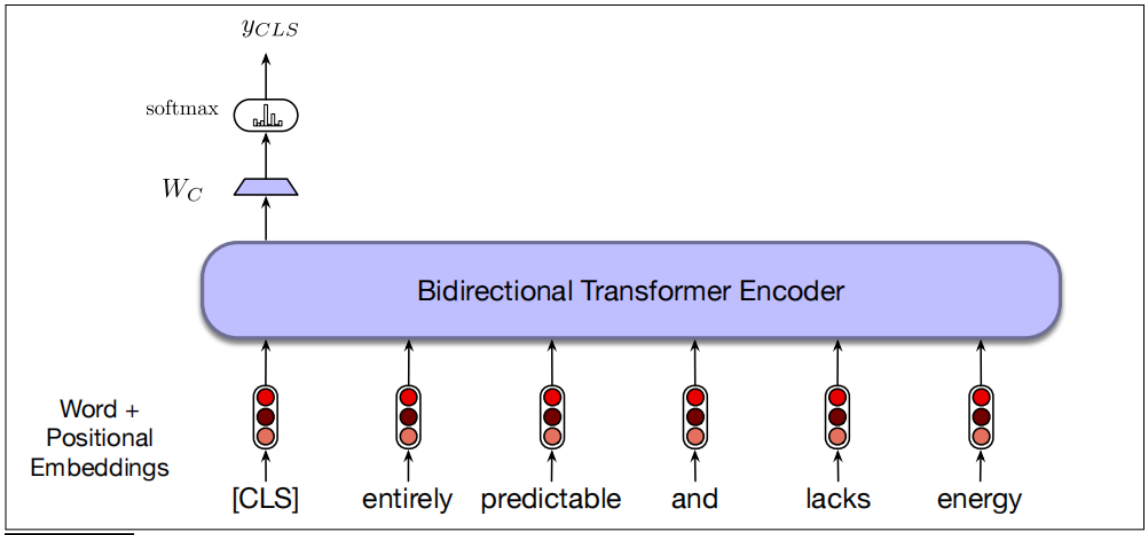

Figure 11.8 Sequence classification with a bidirectional transformer encoder. The output vector for the [CLS] token serves as input to a simple classifier.

Pair-Wise Sequence Classification

任务：logical entailment, paraphrase detection discourse analysis

entailment——natural language inference（NLI）/recognizing textual entailment：MultiNLI 数据集

- Neutral
  - a: Jon walked back to the town to the smithy.
  - b: Jon traveled back to his hometown.
- Contradicts
  - a: Tourist Information offices can be very helpful.
  - b: Tourist Information offices are never of any help.
- Entails
  - a: I'm confused.
  - b: Not all of it is very clear to me.

[CLS]+sentence1+[SEP]+sentence2拼接成一句话输入模型，后接一个分类器

Sequence Labelling

任务：词性标记或基于bio的命名实体识别

每个词的输出向量被传入一个分类器得到概率分布，经过softmax得到最终的类别。可以直接使用贪心算法分别计算每个输入的最终类别，也可以输入CRF将全局的标签级转换纳入考虑

问题：子词编码与这一类问题的词语级标注有矛盾

解决方法：

- 训练时，将每个子词的标签都设置为原词的标签
- 解码时，最简单是使用第一个子词的类别。复杂的是将多个子词的分布一起考量

Fine-tuning for Span-Based Applications

面向间隔的应用主要聚焦于生成和操作连续标记序列的表征

典型的操作包括识别感兴趣的间隔，根据某些标签方案对间隔进行分类，以及确定已发现的间隔之间的关系。应用程序包括命名实体识别、问题回答、语法解析、语义角色标记和共指代消歧。

一般指定任务相关的最大长度限制L，保证首尾序号差$j-i<L$。对于输入x，合法的间隔记为$S(x)$

首先需要得到每个间隔的表征，大多数使用间隔边界表征和间隔摘要表征，为了计算一个统一的间隔表征，直接将这些表征拼接在一起。

最简单：首尾token表征作为边界表征，内部token表征的平均值作为摘要表征

$$
\begin{aligned}
g_{ij} &= \frac{1}{(j-i)+1}\sum_{k=i}^{j} h_k \\
\text{spanRep}_{ij} &= [h_i; h_j; g_{i,j}]
\end{aligned}
$$

没有考虑首尾token的区别，因此更好的是设计两个FFN

$$
\begin{aligned}
s_i &= \text{FFN}_{start}(h_i) \\
e_j &= \text{FFN}_{end}(h_j) \\
\text{spanRep}_{ij} &= [s_i; e_j; g_{i,j}]
\end{aligned}
$$

简单的取中间内容的平均也无法取得好效果，因此需要使用自注意力层得到摘要表征

$$\mathbf{g}_{ij} = \text{SelfAttention}(\mathbf{h}_{i:j})$$

binary span identifification，span classifification，span relation classifification

$$y_{ij} = \text{softmax}(\text{FFN}(g_{ij}))$$

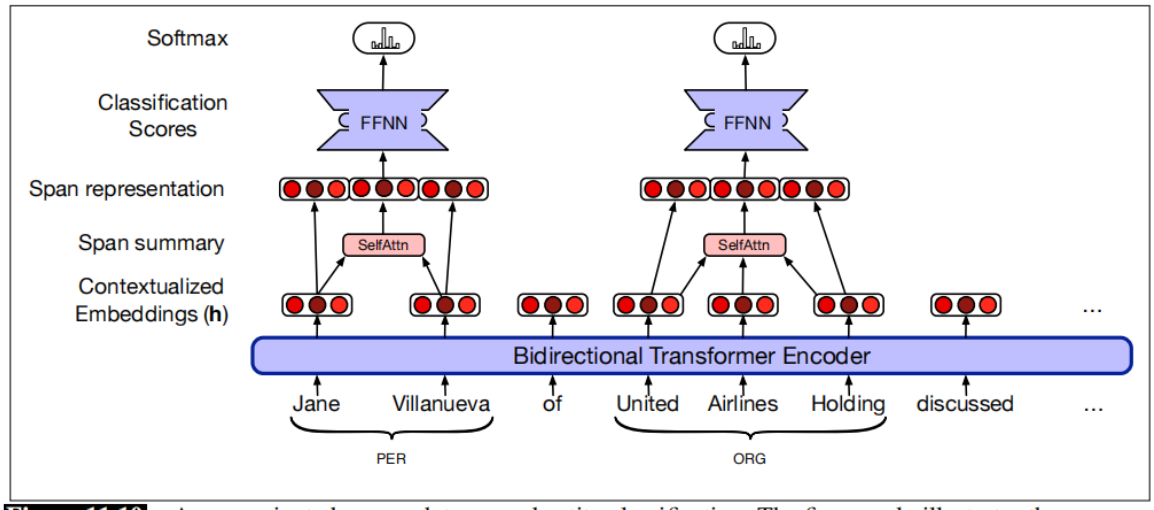

Figure 11.10 A span-oriented approach to named entity classification. The figure only illustrates the computation for 2 spans corresponding to ground truth named entities. In reality, the network scores all of the $\frac{T(T-1)}{2}$ spans in the text. That is, all the unigrams, bigrams, trigrams, etc. up to the length limit.

使用基于间隔的方法优势：

- 可以避免出现一个词错整个实体标注错误的情况，只需要一次分类。
- 基于间隔的方法的第二个优点是，它们可以自然地适应命名实体嵌入。比如United Airlines 和 United Airlines Holding都是合法的，但是BIO标注没法同时标注两个，基于间隔的方法却可以，因为不同间隔是独立标注的