智能问答系统实践

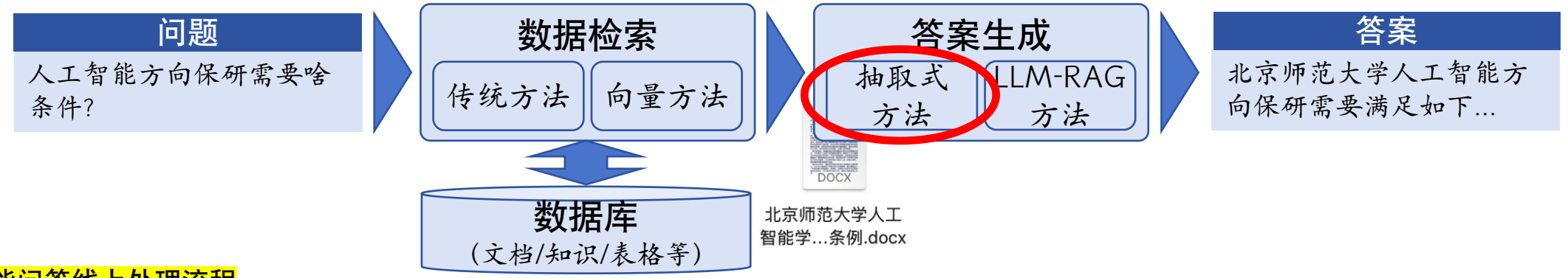
第五课：答案抽取

姜文斌
北京师范大学人工智能学院
2025.03.27

我的位置

智能问答系统

针对用户提出的自然语言问题，从数据库中检索相关信息，并依据相关信息作出回答



智能问答线上处理流程

智能问答线下处理模块

问答数据库构建	数据检索模块构建	答案生成模块构建	效果评估模块构建
基于传统方法的数据建库	传统语义匹配模型构建	抽取式答案生成模型构建	文档检索效果评估
基于向量方法的数据建库	向量语义匹配模型构建	RAG式答案生成模型构建	问答整体效果评估

目录

自然语言处理

- 自然语言处理（NLP）包含很多具体的基础任务和高阶任务

- 基础任务示例
  - **词法分析**：将给定字符序列解析为词语序列，是上层NLP任务的基础
  - **文本分类**：将给定文本划分到特定的类别，可以建模很多任务如情感分析等
- 高阶任务示例
  - **语义分析**：将给定字符序列解析为语义结构，经典NLP任务中最困难之一
  - **机器翻译**：将给定字符序列翻译为另一种语言，经典NLP任务重最困难之一

机器学习时代的NLP

- 不同任务各有适宜的机器学习模型，且需要为模型手工设计特征模板
- 示例：词法分析（下雨天地面积水 > 下雨 天 地面 积水）
  - 模型选择：感知机模型，最大熵模型，支持向量机模型…
  - 特征设计：字符N元组，如针对输入“下雨天地面积水”，$F(地, C_{-2}C_{-1}) = 雨天$
  - 标签设计：$L(地) = B\text{-}Noun$，即“地”是一个名词的起始字符
- 主要缺点
  - 特征设计困难：简单特征区分度不够，复杂特征会导致稀疏和空间爆炸
  - 语义理解较浅：受制于特征设计和浅层模型，难以实现对输入文本的深度理解

深度学习时代的NLP

- 任务根据输入数据结构和输出数据结构，归类到少数几种神经网络类型
  - 嵌入层：输入符号序列中，每个符号都对应着一个可学习的向量表示
  - 编码器/解码器：分别是处理任务输入的模块，和生成任务输出的模块
- 常见神经网络类型
  - RNN类：作为编码器处理序列的输入数据，作为解码器生成序列的输出数据
  - GNN类：作为编码器处理图状的输入数据，通常不作为解码器
  - Transformer：可认为是特殊的GNN，考虑所有符号之间的信息交互
- 主要缺点
  - 需要大量标注数据：每种任务需要大量标注语料去训练，甚至比浅层模型需要的更多

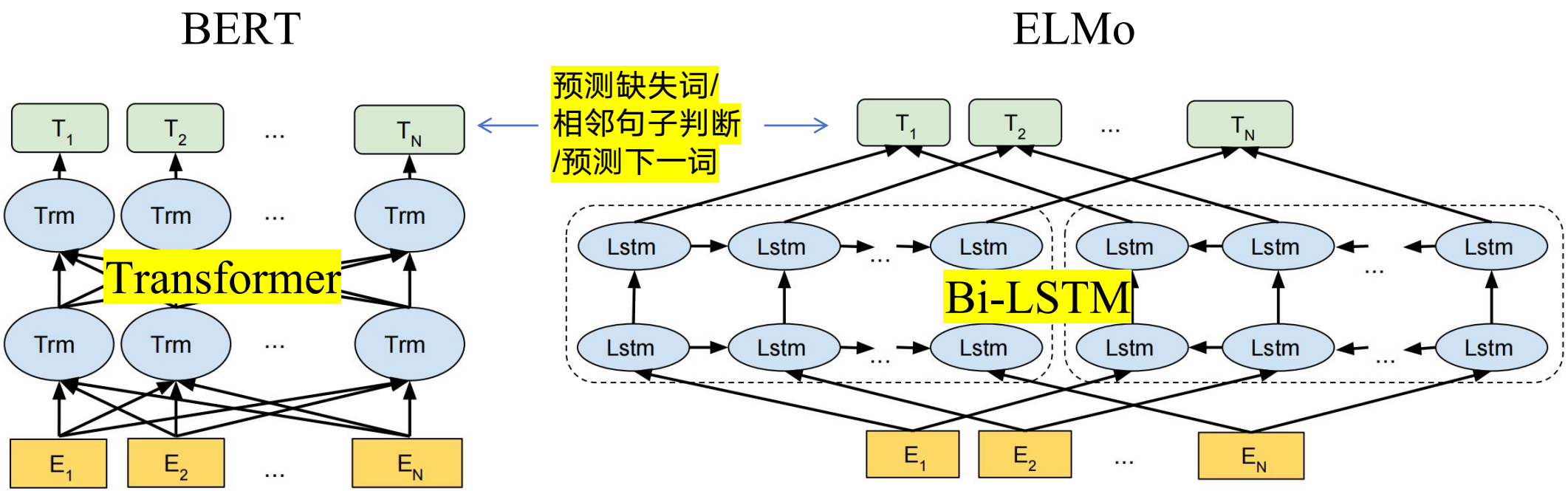
目录

基于简单预训练的NLP

- 通过海量文本数据中的词语共现关系，可以学到词语的通用语义表示
  - 嵌入层热启动：预训练的词语表示作为嵌入层初始值，显著提升训练效率和效果
- 常见简单预训练技术
  - Word2Vec：通过中心词预测周围的上下文单词，或者通过上下文单词预测中心词
  - Glove：遍历语料库构建词语共现矩阵，基于词语共现矩阵降维生成低维的词向量
  - ELMo：基于双层双向LSTM网络结构，根据词语的上下文动态地生成词向量
- 主要缺点
  - 仍需大量调整网络参数：仅实现嵌入层热启动，无法实现编解码器的迁移学习

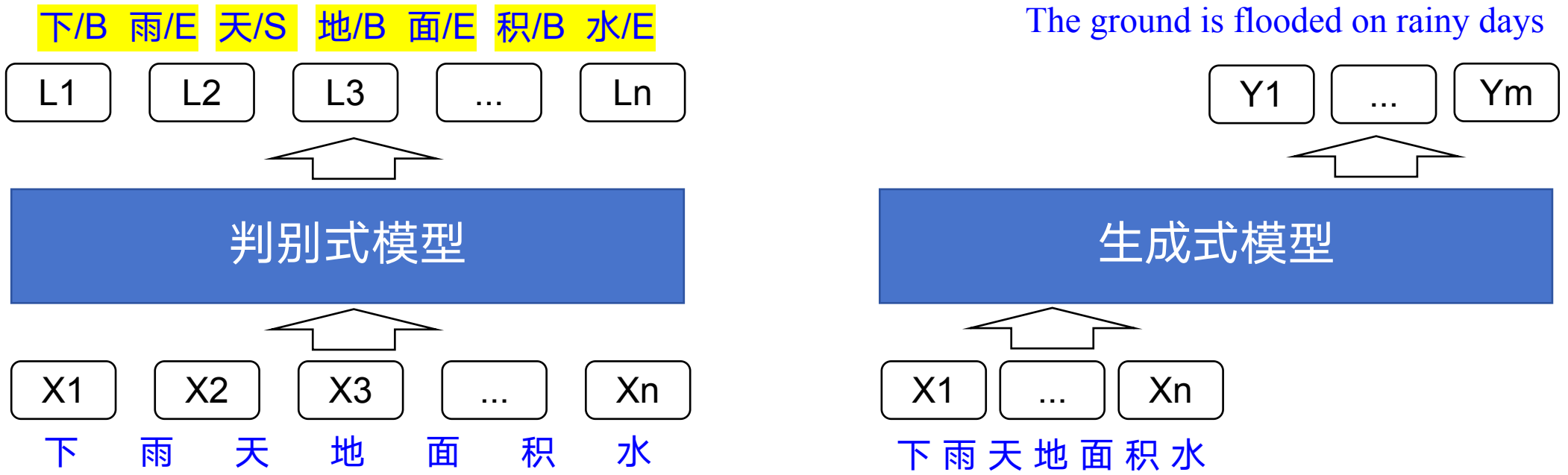
BERT

- 通过相邻句子预测和词语完形填空等任务，自动学习网络的语言理解能力
  - 嵌入层/其他层：分别学到词语语义表示，以及输入文本不同抽象度的语义表示



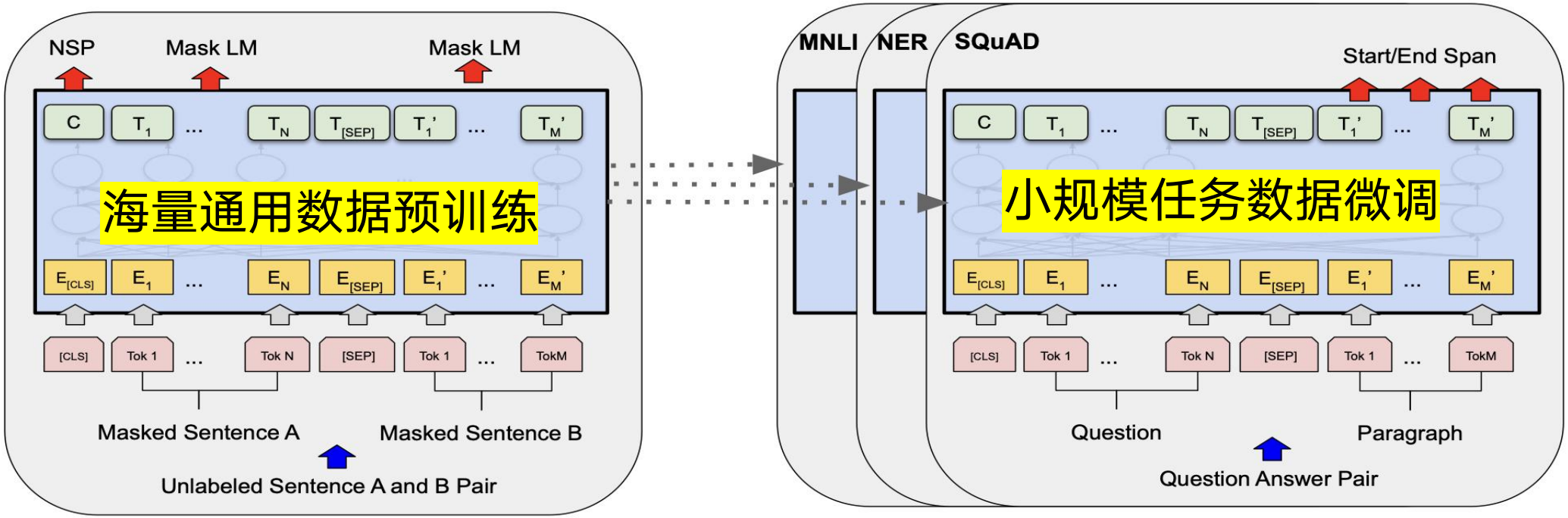
判别式预训练模型

- 判别式模型 vs 生成式模型
  - 判别式：用于“判别”，对输入序列进行统一分类标签或者逐符号标注标签
  - 生成式：用于“生成”，根据输入序列，输出任意长度的符号序列



“大模型+微调”范式

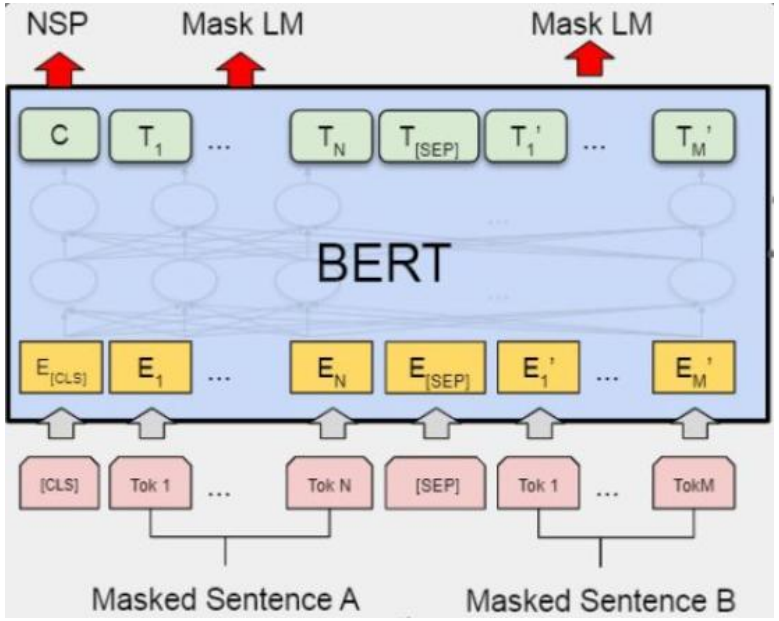
- BERT开创了“大模型+微调”的NLP求解新范式
  - 输出层定制：根据任务的输出，对BERT输出层进行定制，分类头的增删改
  - BERT微调：在BERT基础上用少量标注数据微调，通常即可超越之前的SOTA

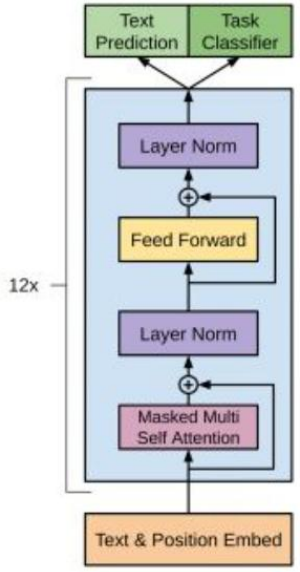


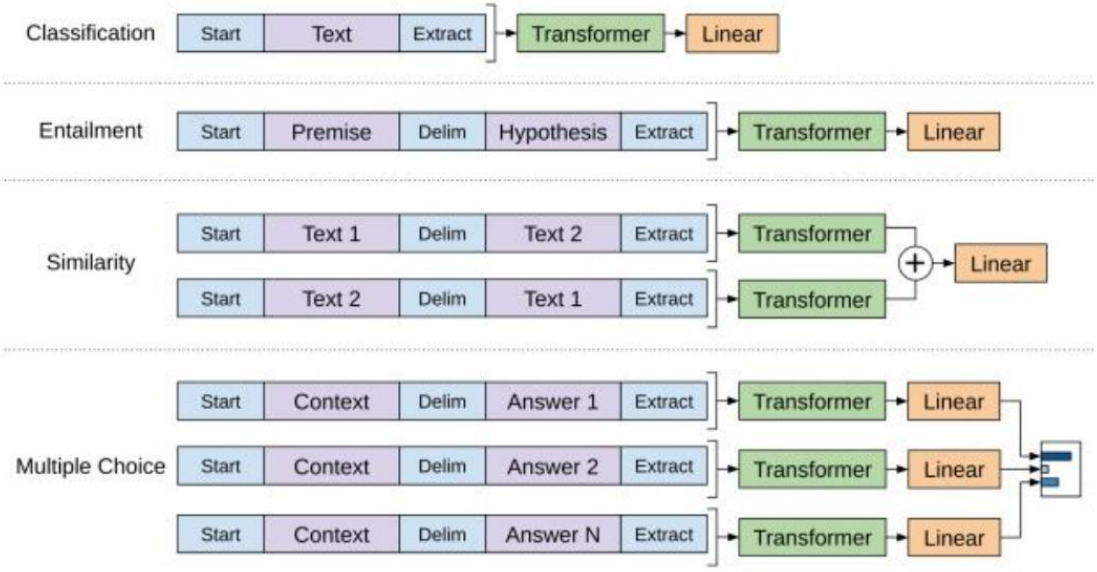
目录

BERT架构

- BERT基于Transformer，采用词语完型填空和相邻句子预测进行大规模预训练，学到多层次的语义表示
  - 语言模型都是通过rnn，lstm来建模，无法并行化，给模型的训练和推理带来了困难
  - Transformer使用了自注意的方式对上下文进行建模，训练可并行，建模效果更好







注意力机制

- 注意力机制的本质是通过注意力进行上下文加权融合
  - Attention机制主要涉及到三个概念：Query、Key和Value
  - Attention机制将目标字作为Query、其上下文的各个字作为Key，并将Query与各个Key的相似性作为权重，融合上下文的所有Value

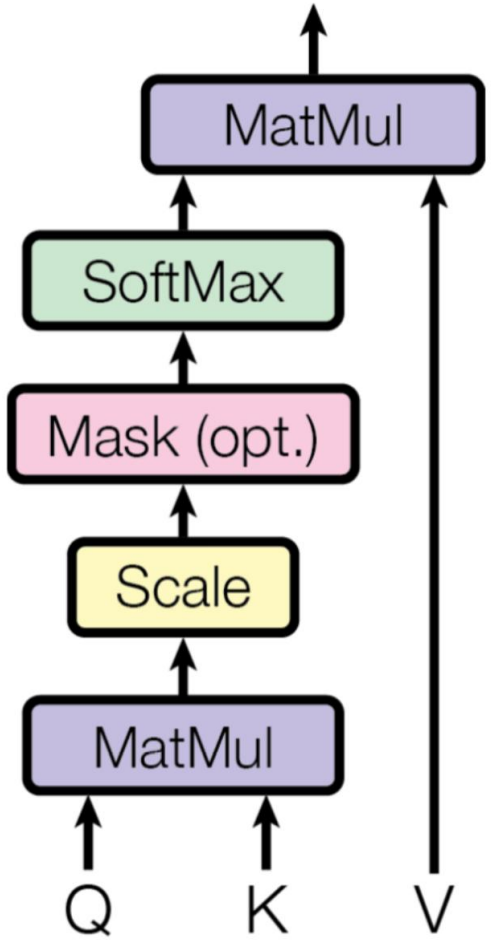
$$Attention(Q, K_i, V_i) = softmax(\frac{Q^T K_i}{\sqrt{d_k}})V_i$$

$$Attention(Q, K, V) = softmax(\frac{Q^T K}{\sqrt{d_k}})V$$

$$Q = \underbrace{\begin{pmatrix} q \\ q \\ q \\ \vdots \\ q \end{pmatrix}}_{d_k} \}m$$

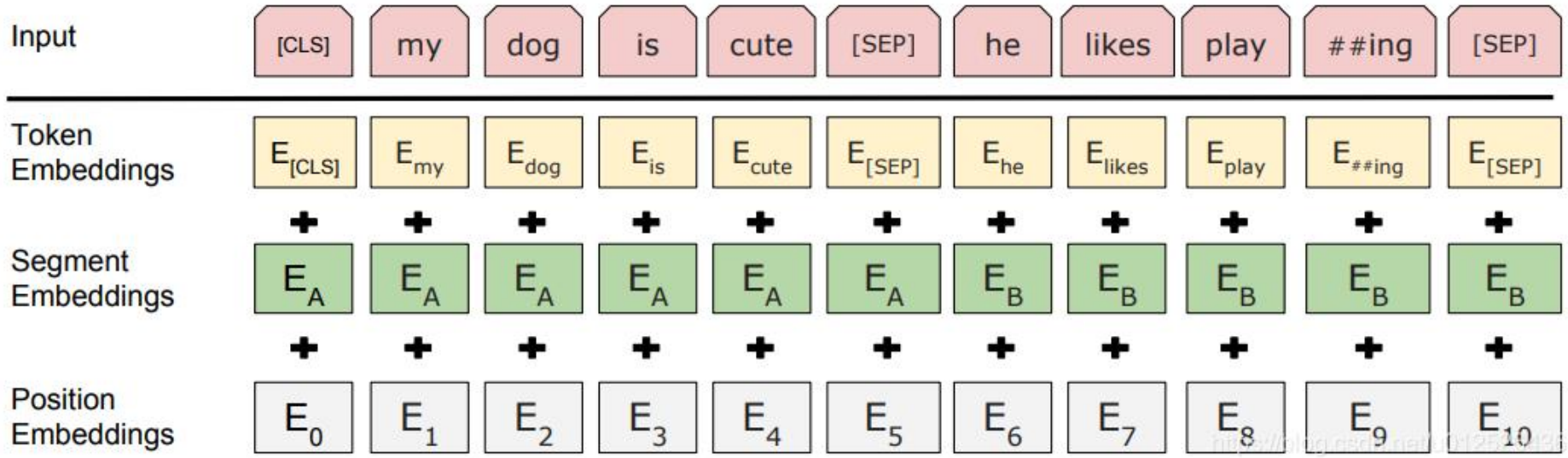
$$K = \underbrace{\begin{pmatrix} k_1 \\ k_2 \\ k_3 \\ \vdots \\ k_m \end{pmatrix}}_{d_k} \}m$$

$$V = \underbrace{\begin{pmatrix} v_1 \\ v_2 \\ v_3 \\ \vdots \\ v_m \end{pmatrix}}_{d_v} \}m$$



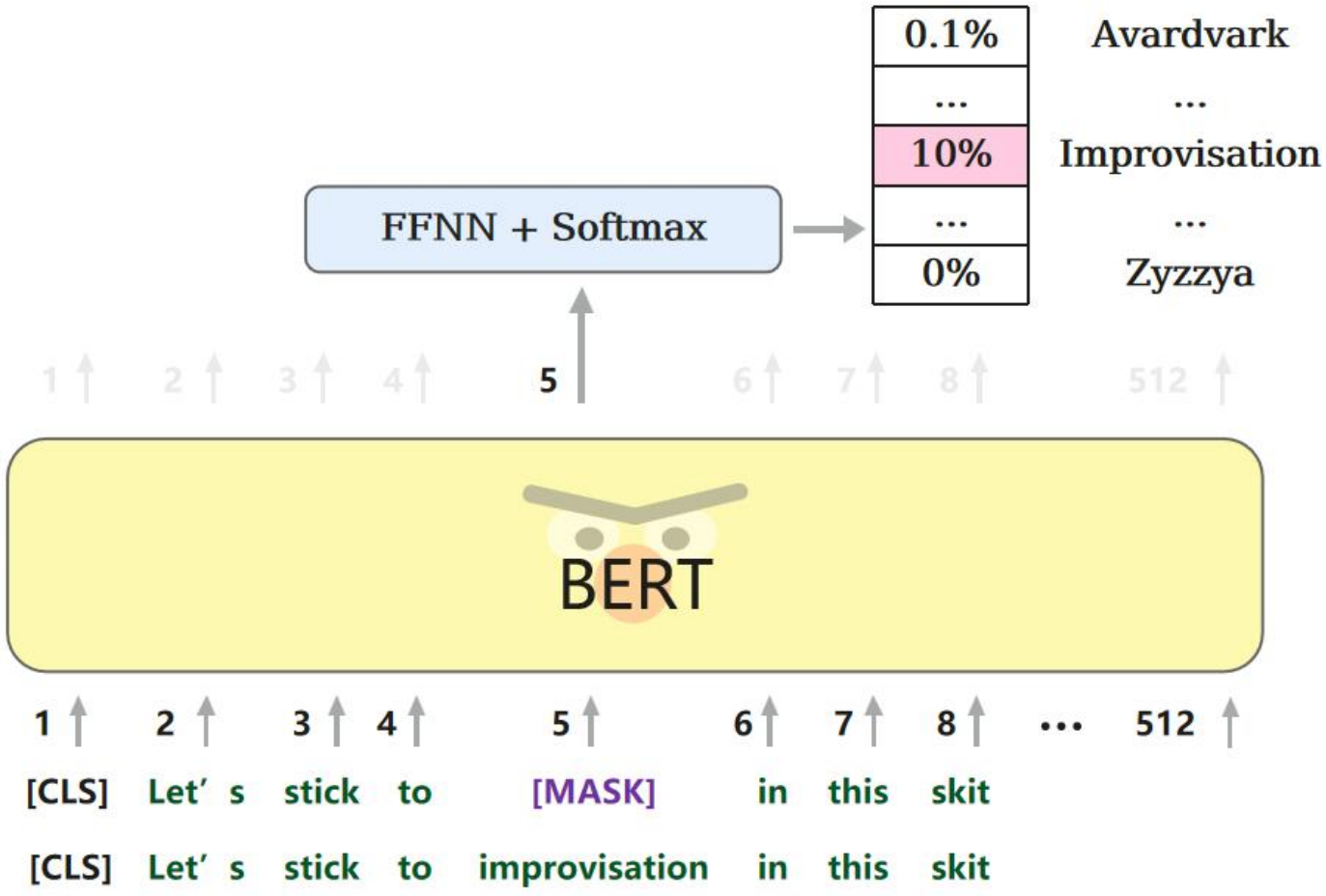
输入层

- 输入包括三个部分，分别是token嵌入，segment嵌入和position嵌入
  - Token：是词向量，第一个单词是CLS标志，可用于分类任务，对于非分类问题则忽略
  - Segment：用来区别两种句子，因为不光做LM还要做以两个句子为输入的分类任务
  - Position：和之前Transformer的位置向量不一样，不是三角函数而是学习出来的



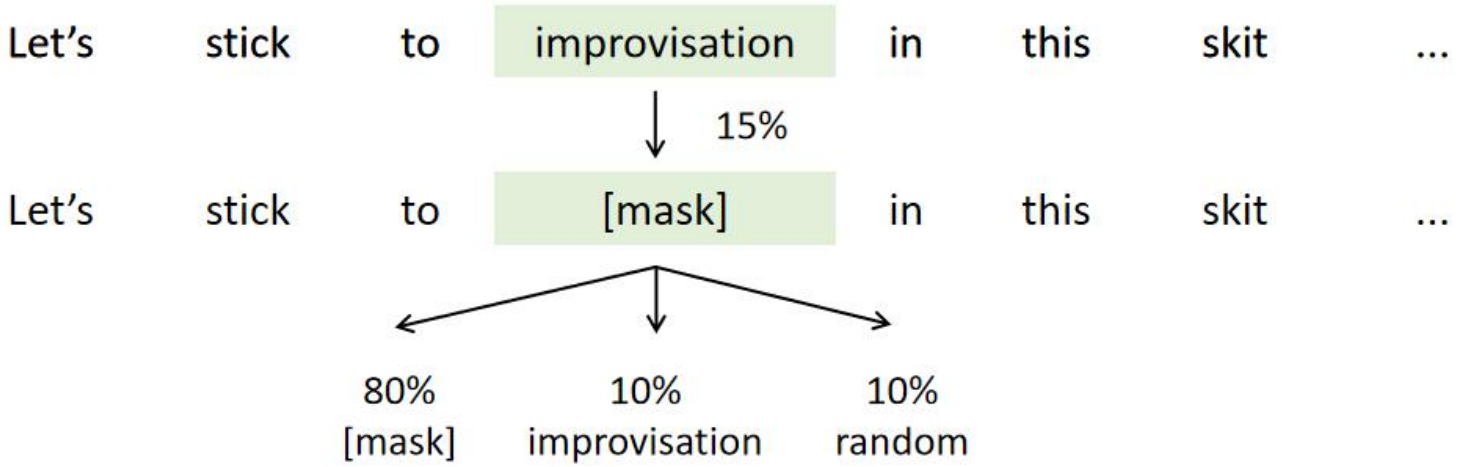
BERT训练：MLM

- **Masked Language Model（MLM）**
  - 类似于完形填空，从输入中遮蔽掉一些单词，然后通过上下文预测该单词
  - 完型填空任务是可以自动构造的，因此这类标注信息可以大规模生成



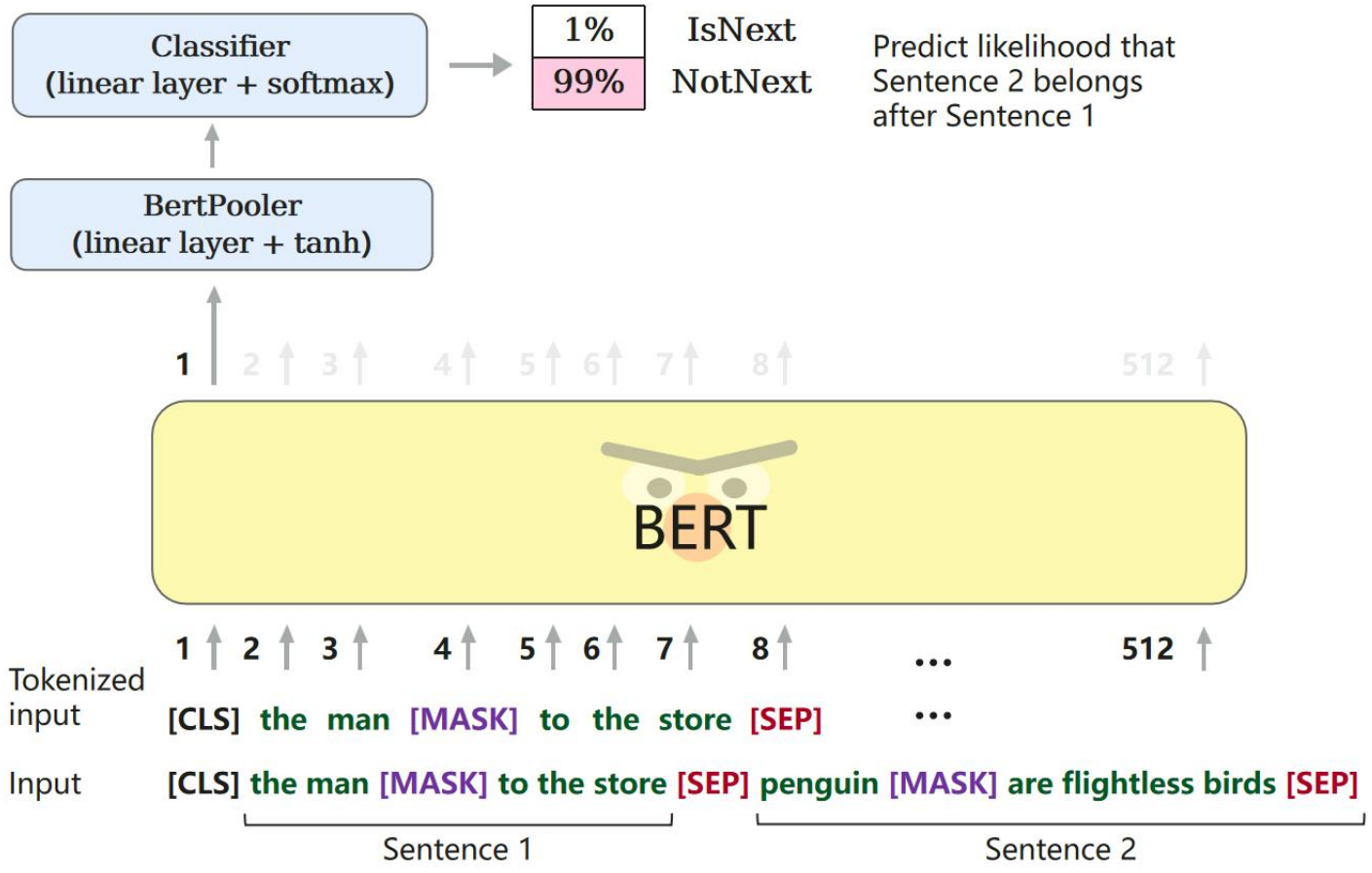
BERT训练：MLM

- BERT随机遮盖15%的输入，在这15%中进一步细分不同的替换方法
  - 80%的概率替换为[MASK]
  - 10%的概率替换为文本中的随机词
  - 10%的概率保持为原始形态



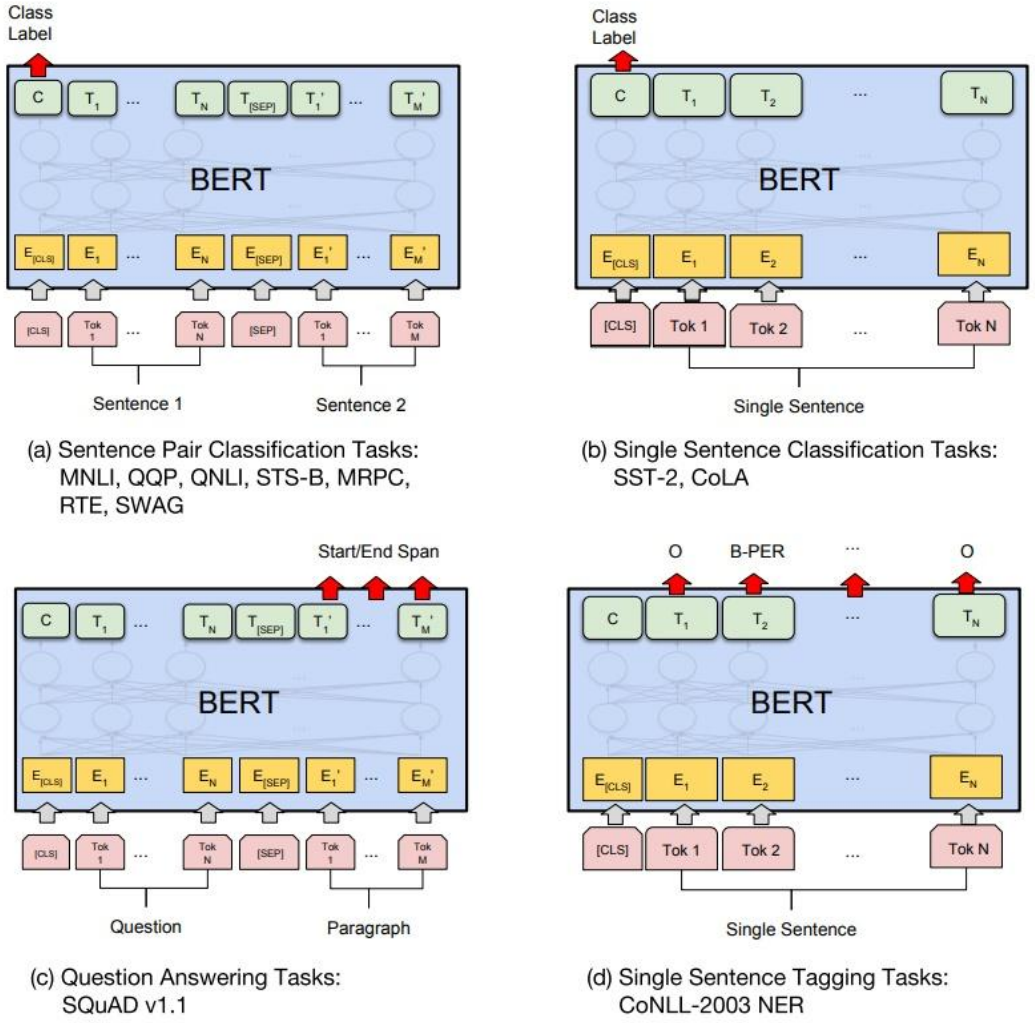
BERT训练：NSP

- **Next Sentence Prediction（NSP）**
  - 是一个针对句子对的分类问题判断一组句子中，句子B是否为句子A的下一句
  - 自然语言文本中天然蕴含着句子邻接信息，该标注可以大规模自动生成



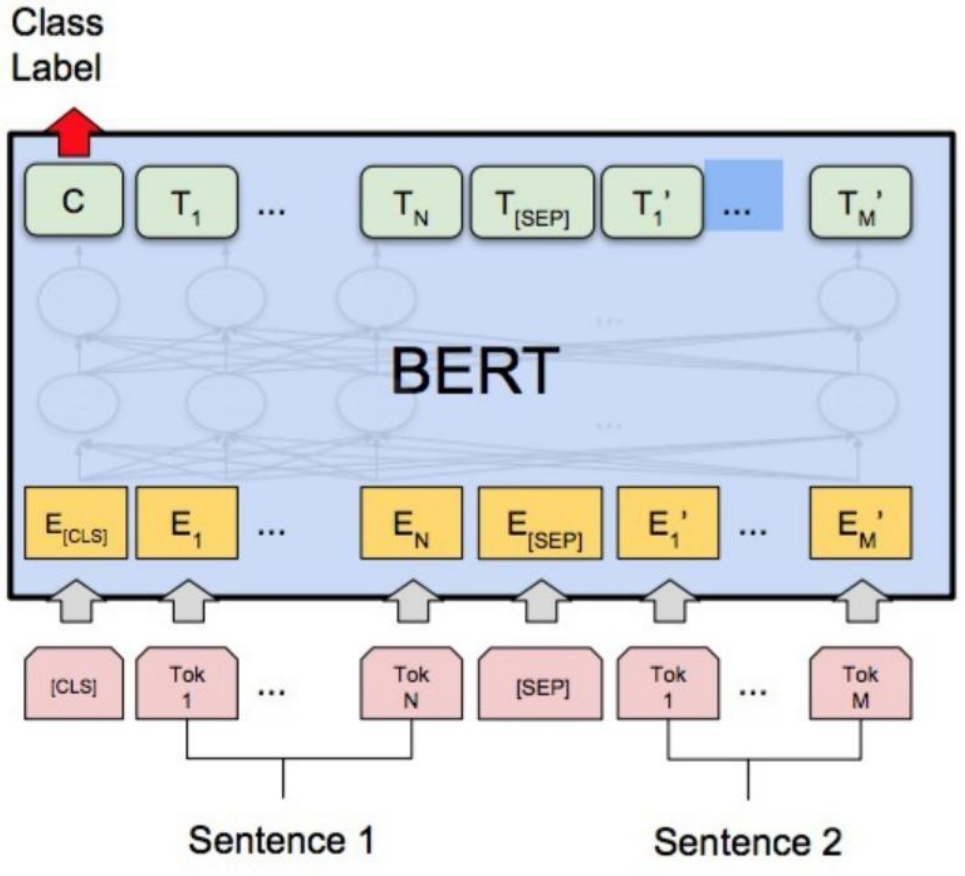
BERT微调

- 在海量无标签语料上训练完BERT之后，便可以将其应用到NLP的各个任务中了
- 微调任务主要包括以下四类
  - 基于句子对的分类任务
  - 基于单个句子的分类任务
  - 问答任务
  - 命名实体识别



微调：句对分类

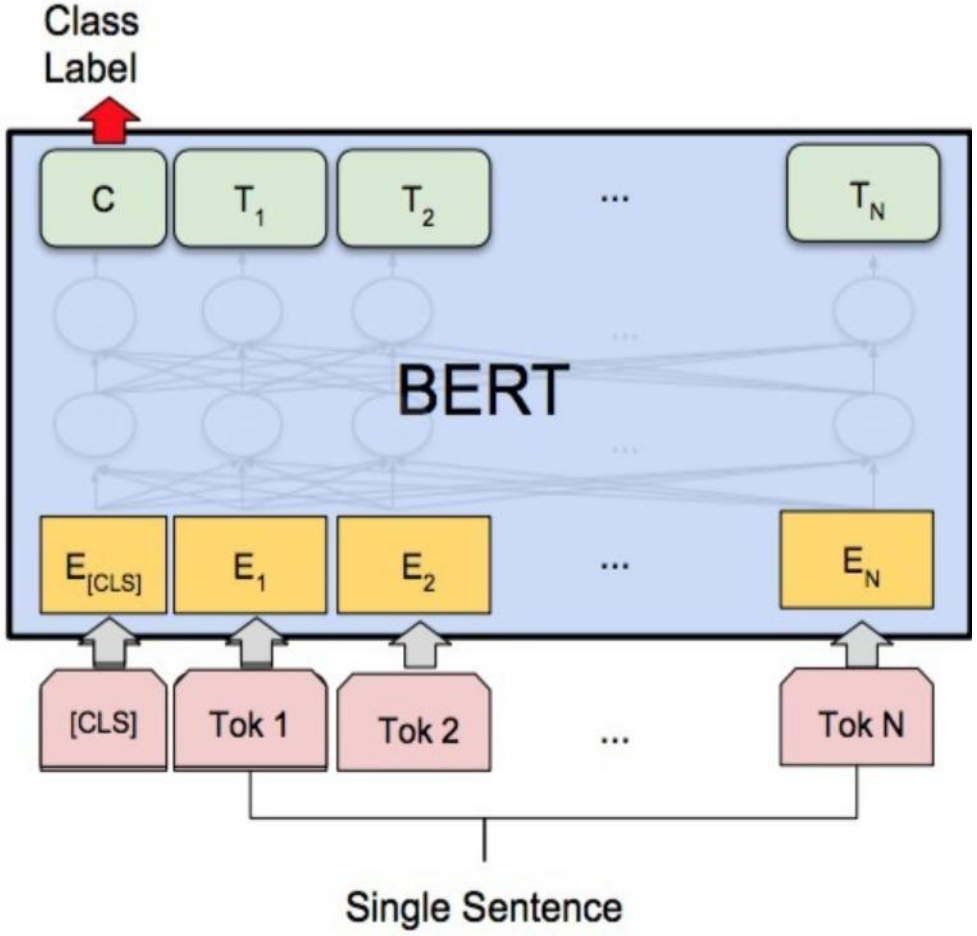
- MNLI
  - 给定一个前提，去推断假设与前提的关系，蕴含关系、矛盾关系以及中立关系
- QQP
  - 判断两个问题句是否表示的是一样的意思
- QNLI
  - 用于判断文本是否包含问题的答案，类似于我们做阅读理解定位问题所在的段落
- ...



(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG

微调：句子分类

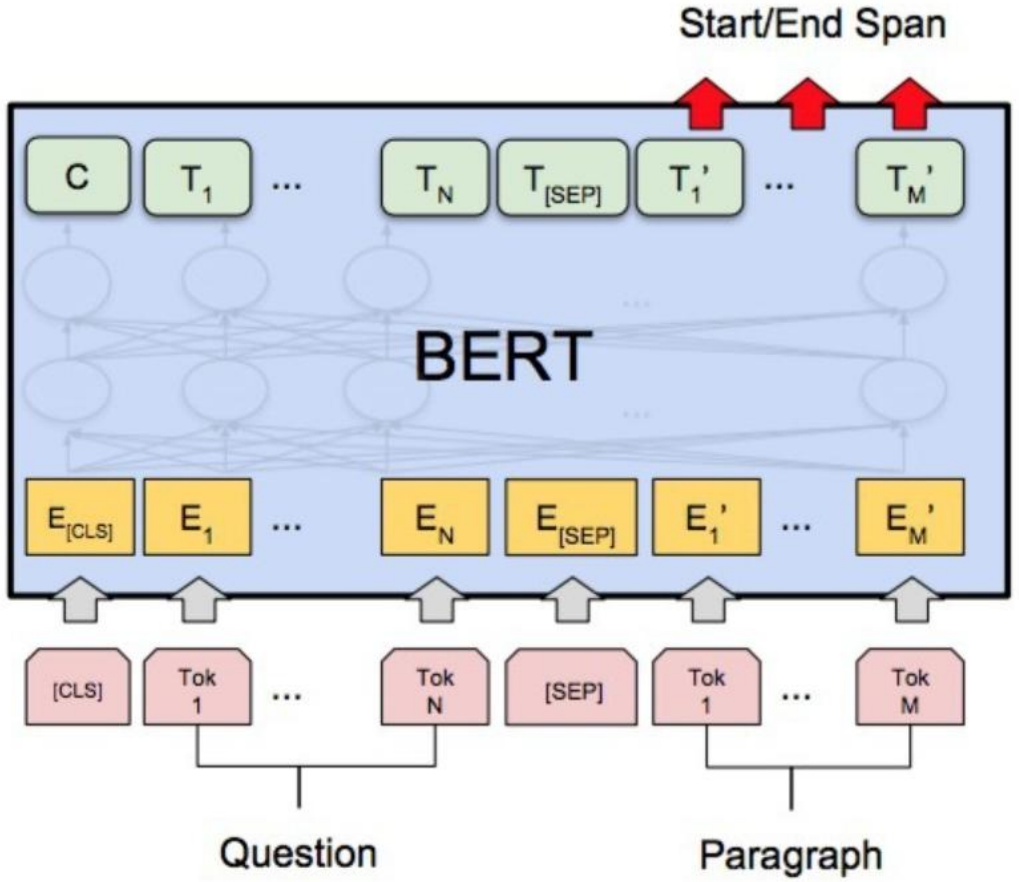
- SST-2
  - 电影评价的情感分析
- CoLA
  - 对一个给定句子，判定其是否语法正确
- GLUE数据集的分类任务
  - MNLI，QQP，QNLI，SST-B，MRPC，RTE，SST-2，CoLA
  - 根据[CLS]生成特征向量，并通过一层全连接进行微调，损失函数根据任务类型自行设计



(b) Single Sentence Classification Tasks: SST-2, CoLA

微调：问答任务

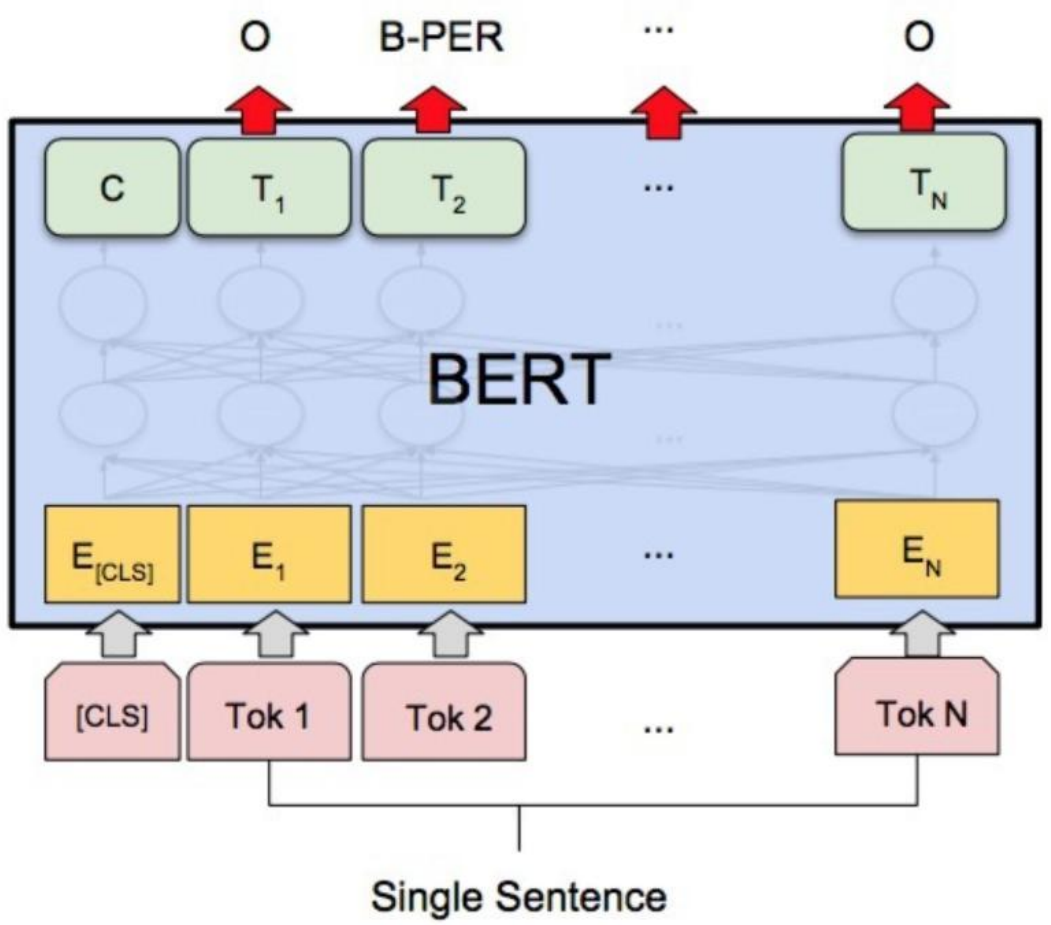
- SQuAD
  - 给定一个句子（通常是一个问题）和一段描述文本，输出这个问题的答案，类似于做阅读理解的简答题
  - 如右图，SQuAD的输入是问题和描述文本的句子对，输出是特征向量，通过在描述文本上接一层激活函数为softmax的全连接来获得输出文本的条件概率



(c) Question Answering Tasks: SQuAD v1.1

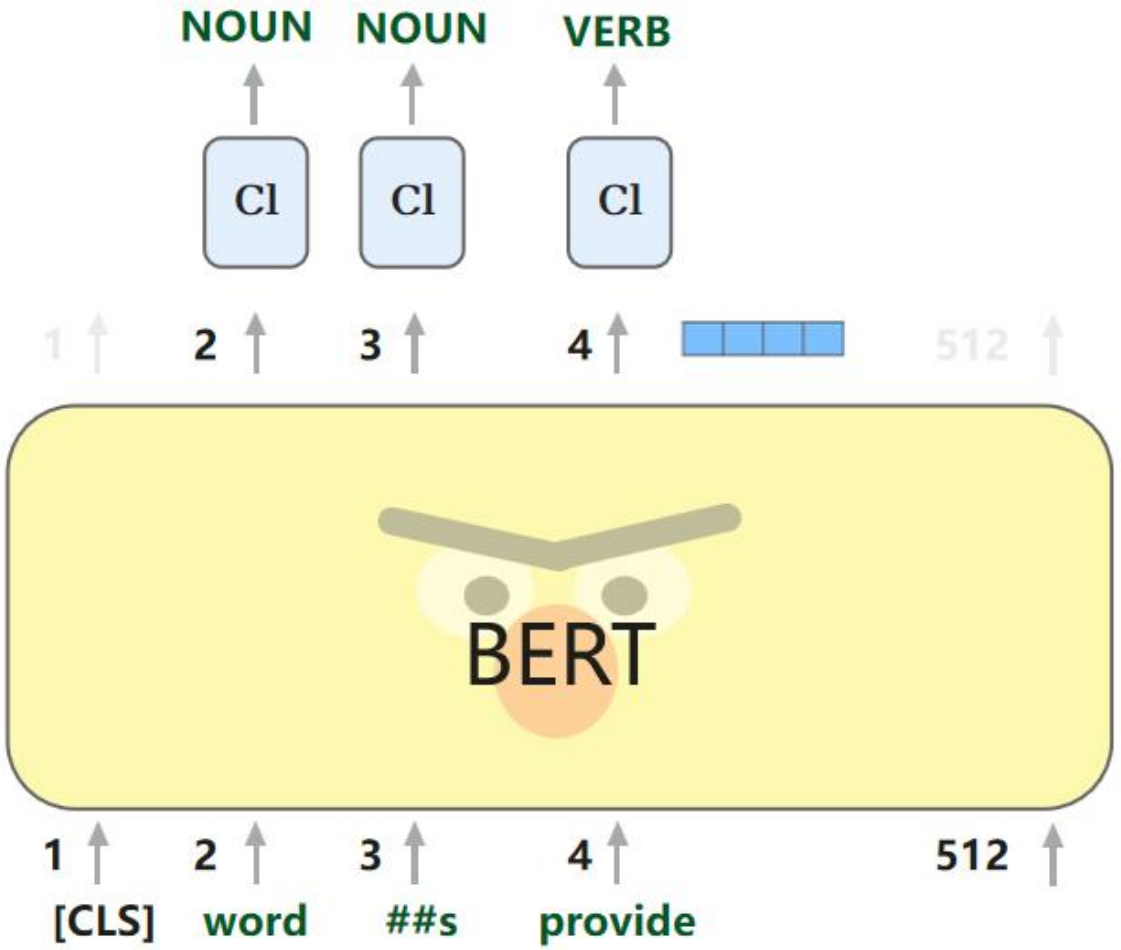
微调：命名实体识别

- CoNLL-2003NER
  - 判断一个句子中的单词是不是人名、地名、机构名等实体或者other（无命名实体）
  - 微调CoNLL-2003NER时将整个句子作为输入，在每个时间片输出一个概率，并通过softmax得到这个Token的实体类别标签



BERT推理

- BERT在推理时并不进行Mask
  - 所有输入Token全部作为输入
- BERT在训练进行Mask的作用
  - 完型填空本身的需要，遮蔽掉该词，然后再预测该词
  - 提高模型的鲁棒性，模拟信息的缺失和噪声，促使模型在有噪声情况下学习



目录

应用场景

- 作为判别式大模型，BERT可用于判别式或可转为判别模式来求解的任务

- 判别式任务
  - 分类：对文本进行分类，对文本对进行分类
  - 序列标注：对文本的每个符号进行分类
- 一些可以转为判别模式求解的任务
  - 句法/语义分析：预测句子中每个词语在句法树/语义树中的位置
  - 对话理解：为序列添加若干个符号，为这些符号打标签（意图、词槽、属性等等）

一般流程

- 任务建模
- 数据准备
- 训练微调
- 测试评估

示例：阅读理解

- 任务定义：给定问题和段落，从段落中识别出回答问题的字符序列
  - 输入：问题，以及包含答案的文本段落
  - 输出：问题的答案在文本段落中的起止位置
- 示例

【输入】
段落：美国的第一任总统是乔治·华盛顿，他于1789年4月30日在纽约联邦大厅宣誓就职。
问题：谁是美国的第一任总统?

【输出】
起止位置：9-14（第一个字符的下标是0）
字符序列：乔治·华盛顿

1. 任务建模

- 将任务建模为针对序列的分类模式，或针对序列元素的标注模式

- 阅读理解任务
  - 将输入段落中的字符，标注好指明答案起止位置的标签

输入问题：谁是美国的第一任总统?
输入段落：美国的第一任总统是乔治·华盛顿，他于...就职。

标注结果：O O O O O O O O O B I I I I E O O O ... O O O

2. 数据准备

- **微调：收集高质量的<输入，输出>数据，用于BERT参数微调**

SQuAD是由斯坦福大学创建的一个广泛使用的机器阅读理解数据集，主要用于训练和评估问答系统。

大规模：SQuAD包含超过10万个问答对，这些问答对来源于Wikipedia文章，覆盖了广泛的主题。

高质量：所有问题和答案都经过了人工标注和验证，确保了数据的准确性和可靠性。

抽取式：SQuAD主要是一个抽取式问答任务，意味着模型需要从给定的文档中提取出正确答案的位置，而不是生成新的句子作为回答。

- **继续训练：如果有大量目标领域文本，也可以基于此进行继续训练**

3. 训练微调

- 对数据进行Tokenization
  - 文本分割成tokens，可以是单词、单词等字符
  - 将tokens映射到词表中的索引，即每个字符对应词表中的一个数字
  - 注意力掩码（attention masks），区分实际内容和填充内容
  - BERT 需要 token_type_ids 来区分不同的句子
- 对Transformer进行训练配置
  - 训练超参数如优化器、迭代轮数等等
  - 成熟的平台会提供丰富、易用的配置功能

4. 测试评估

- **制定评测方法，构造评测工具**

SQuAD的评价指标主要基于精确匹配（Exact Match，简称EM）和部分匹配（Partial Match，简称F1 Score）的度量。

精确匹配（EM）：模型给出的答案与标准答案完全一致时的评价指标。如果模型的答案与参考答案完全相同，则EM得分为1；否则为0。

部分匹配（F1 Score）：通过比较模型答案与参考答案之间的共享词汇来评估答案的相似性。F1 Score是根据模型答案和参考答案之间的匹配度来计算的，综合考虑了精确性（Precision）和召回率（Recall）。

- **构建或者预留测试数据集，用于后续效果评测**

目录

总结

- 第一个产生巨大影响力的判别式大模型
  - 参数大：上亿参数（对比之前的百万参数级别模型）
  - 功能多：任何能够转换为分类或序列标注的任务
- 基于海量文本和自动构造任务学习不同层次的语义知识
  - 完形填空类任务
  - 分类任务如语句关系预测等等
- 相比当今生成式大模型在特定任务仍有优势
  - 所有分类或标注任务仍适用BERT求解

谢谢大家！