

# Google BERT 模型解析及实验探索

追一科技 · 潘晟锋

# BERT是什么

- BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding
- BERT: **B**idirectional **E**ncoder **R**epresentations from **T**ransformers.
- BERT是
  - 预训练
  - 双向的编码表征
  - 深度transformer结构
  - 以语言模型为训练目标

# 大纲

- Part 1 语言模型简介
  - 文本表征
  - 语言模型
  - 深度学习与语言模型
- Part 2 BERT解析
  - BERT的基本结构
  - 模型模块详解
  - 源码中的细节
- Part 3 应用
  - 如何使用预训练模型
  - 实验心得

# 文本表征

- 文本表征是文本的“数字形式”表达：
  - 模型的运算需要数字
  - 模型的运算是连续的
    - 对于 $x, y \in (0, 1)$ , 我们总可以找到介于 $x$ 与 $y$ 之间的值;
- 自然语言是离散的
  - 对于 $v \in \{\text{开心, 很开心, 非常开心}\}$ , 我们无法定义任意两个词之间的词;
- 文本表征
  - one-hot, tf-idf...
  - word2vec, doc2vec, glove, fasttext...
  - ELMo, GPT, BERT...

# 词向量

- 分布式假设
  - 相同上下文语境的词有相似的含义
- 词向量的弊端
  - 每个词一个固定表征
- 结合上下文的动态表征
  - ELMo, GPT, BERT等基于语言模型的表征

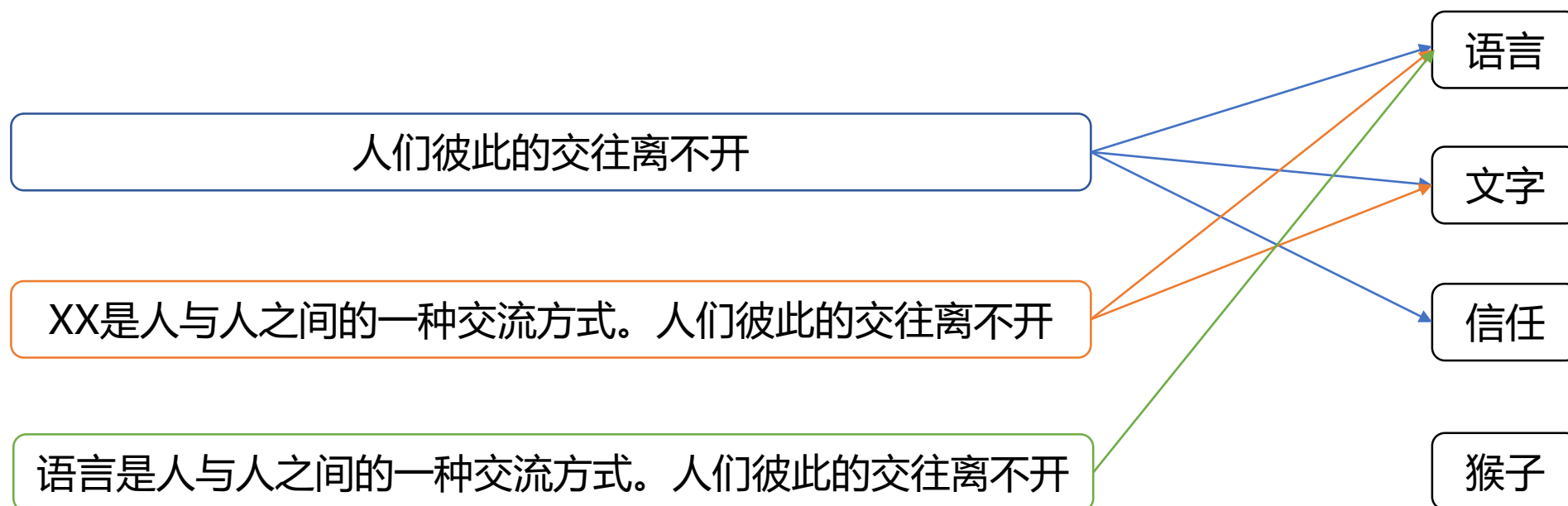
美国总统**特朗普**决定在墨西哥边境修建隔离墙

听说美国的那个**川普**总统要在墨西哥边上修个墙

同学你的**川普**挺带感啊

# 语言模型是什么

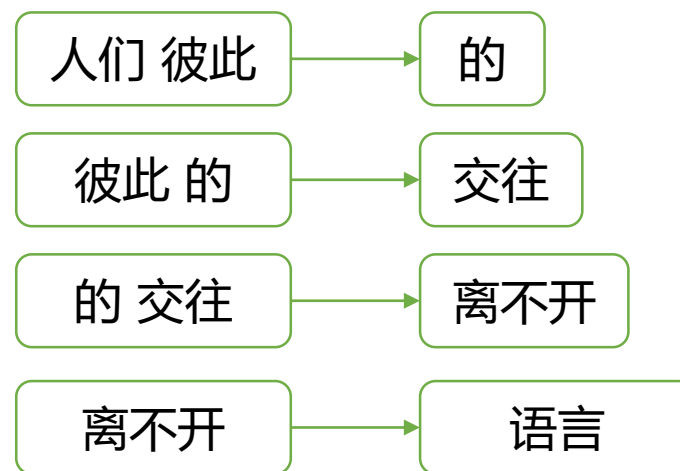
- 语言模型本质上是概率模型；
  - 在给定前文的基础上，从概率上预测接下来的内容；
  - 它的目标就是学会语言的“套路”；
  - 随着前文越详细，预测的目标越明确；



# 传统语言模型的问题

人们彼此的交往离不开语言

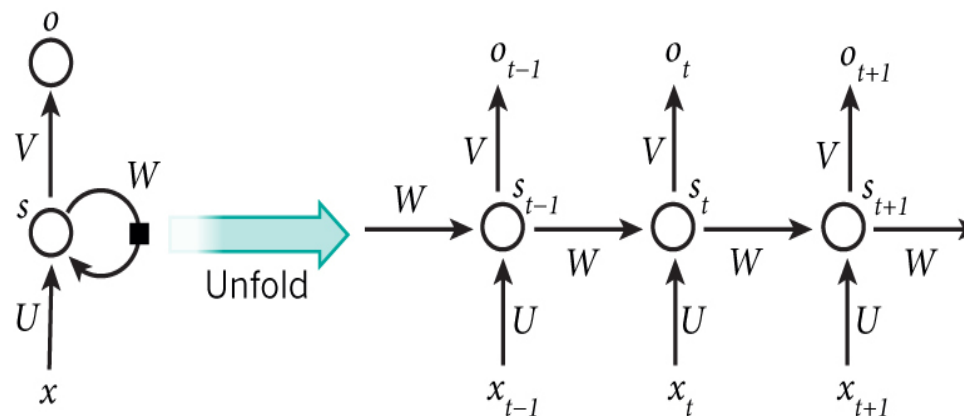
- N元语言模型
  - 过于短程的依赖
- 解决长程依赖的途径



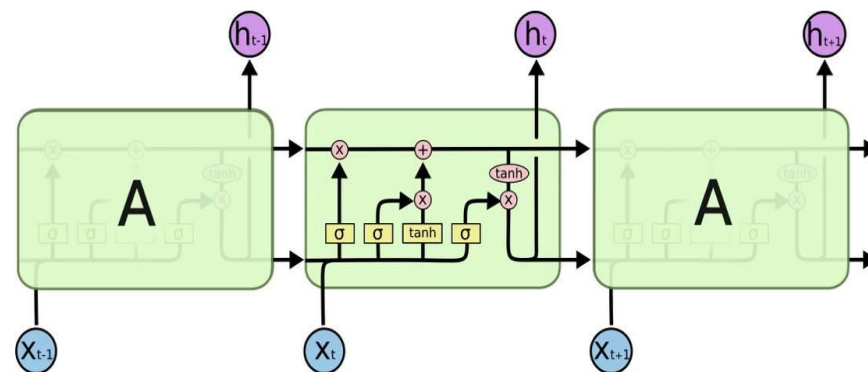
# 传统语言模型的问题

人们彼此的交往离不开语言

- N元语言模型
  - 过于短程的依赖
- 解决长程依赖的途径
  - RNN+LSTM, GRU



Long-Short Term Memory module: LSTM



long-short term memory modules used in an RNN

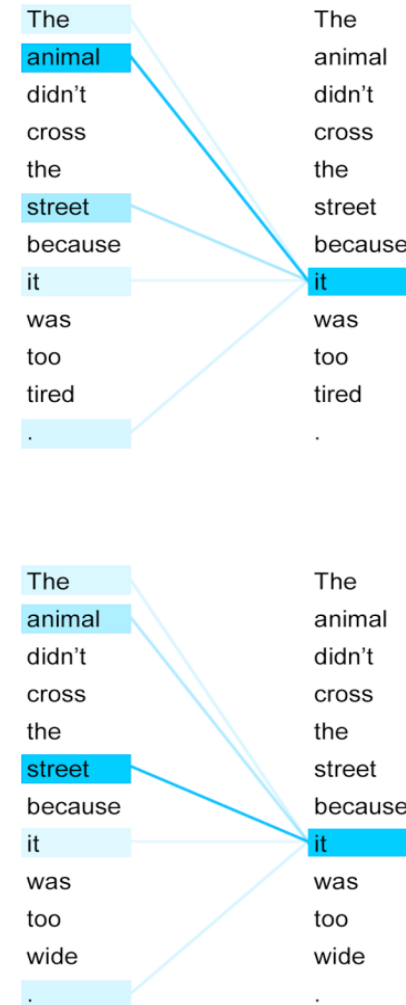
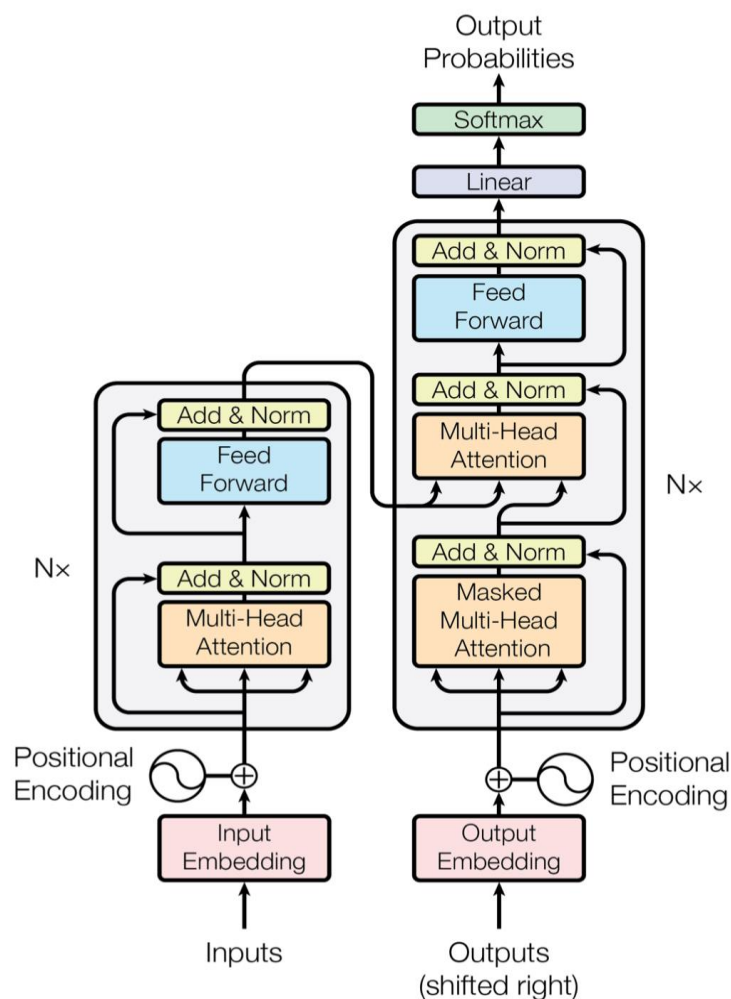




# 传统语言模型的问题

人们彼此的交往离不开语言

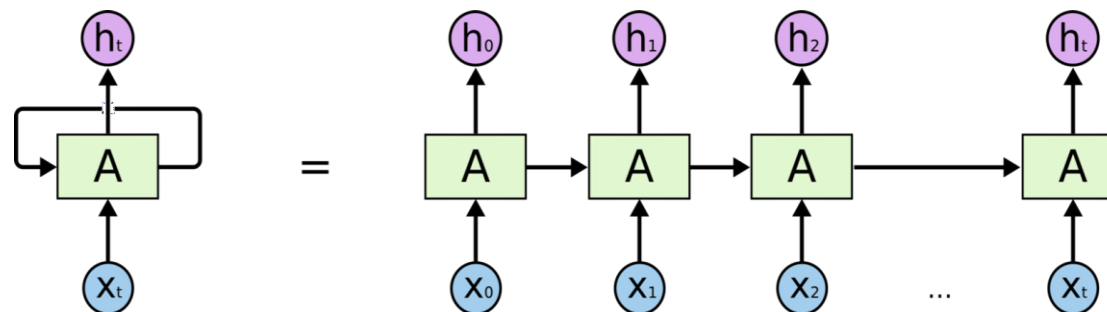
- N元语言模型
  - 过于短程的依赖
- 解决长程依赖的途径
  - RNN+LSTM, GRU
  - Attention



# 神经网络语言模型

人们彼此的交往离不开语言

- 单向RNN



人们

彼此

人们 彼此

的

人们 彼此 的

交往

人们 彼此 的 交往

离不开

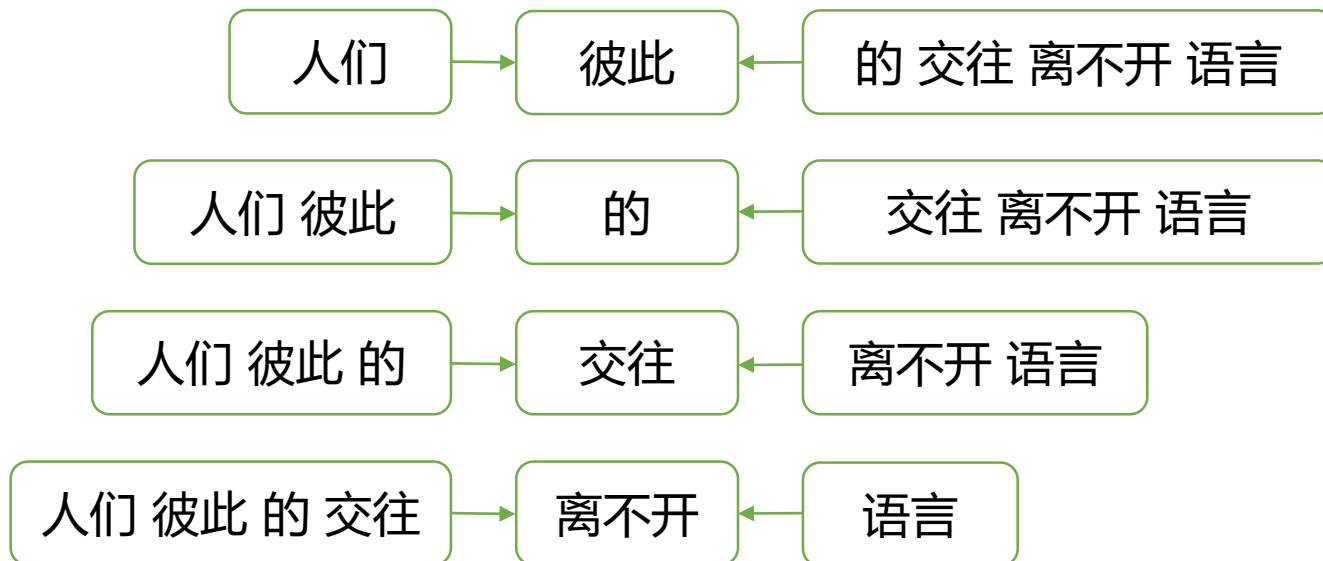
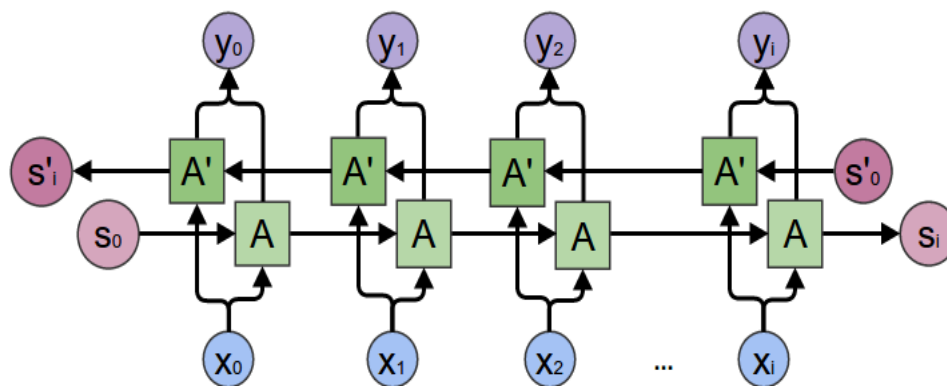
人们 彼此 的 交往 离不开

语言

# 神经网络语言模型

人们彼此的交往离不开语言

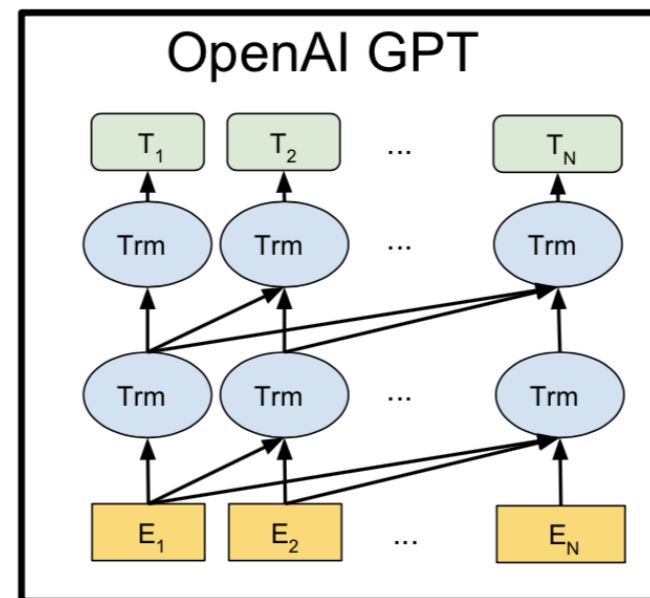
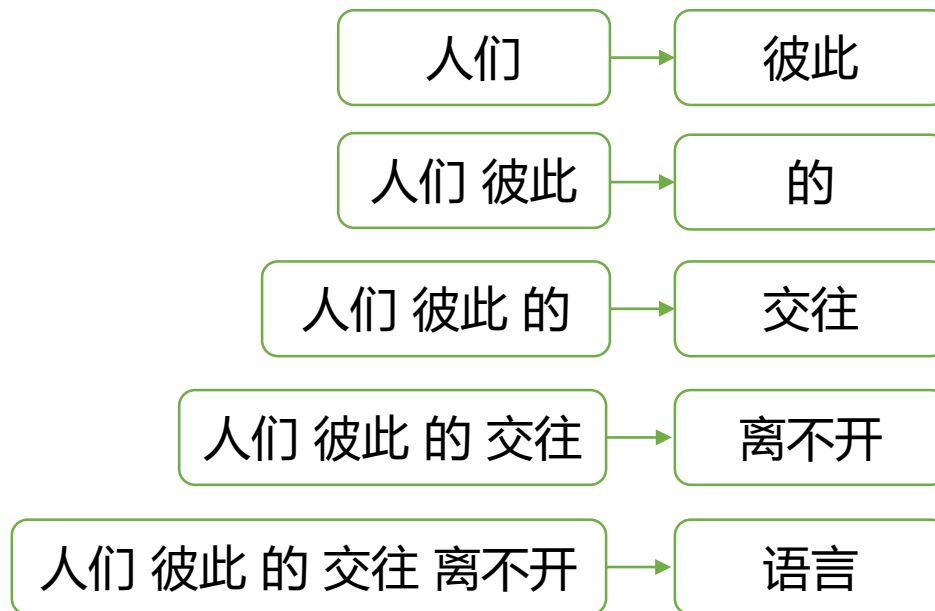
- 单向RNN
- 双向RNN



# 神经网络语言模型

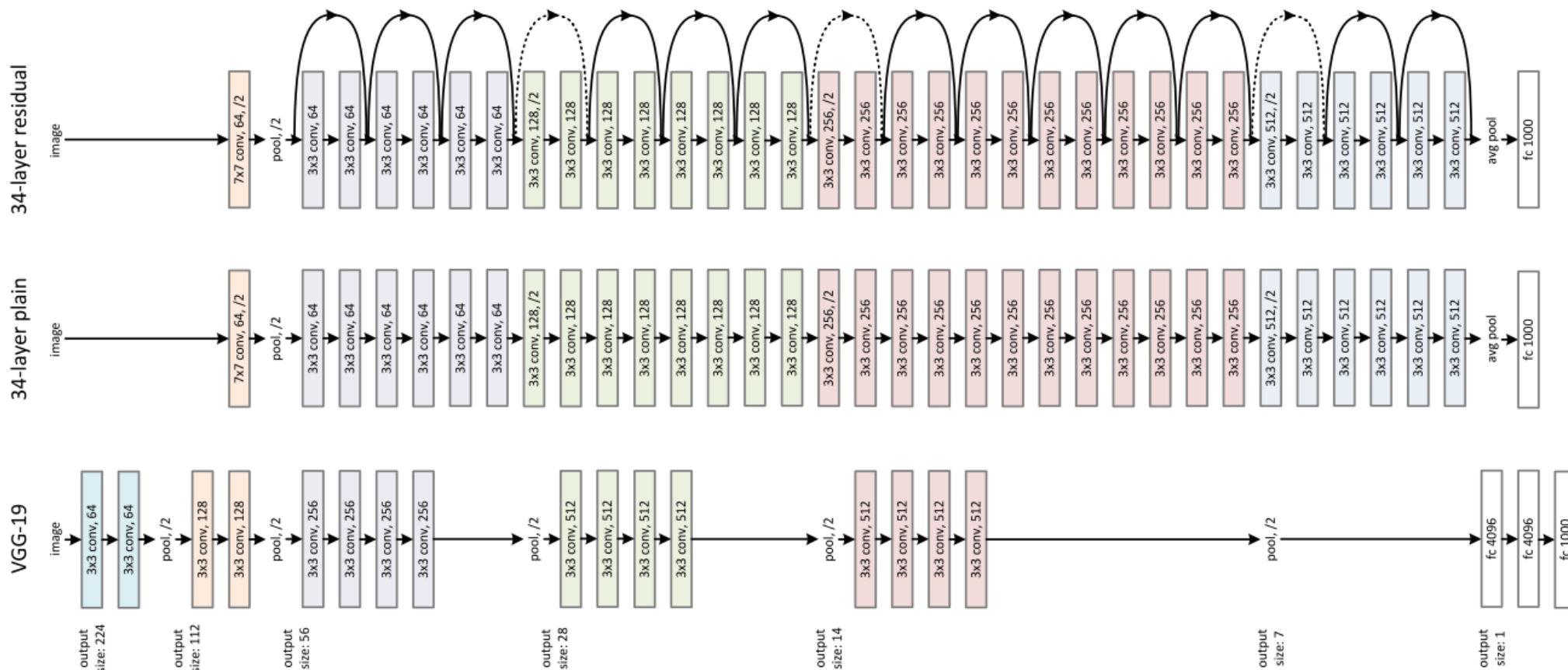
人们彼此的交往离不开语言

- 单向RNN
- 双向RNN
- 单向Attention



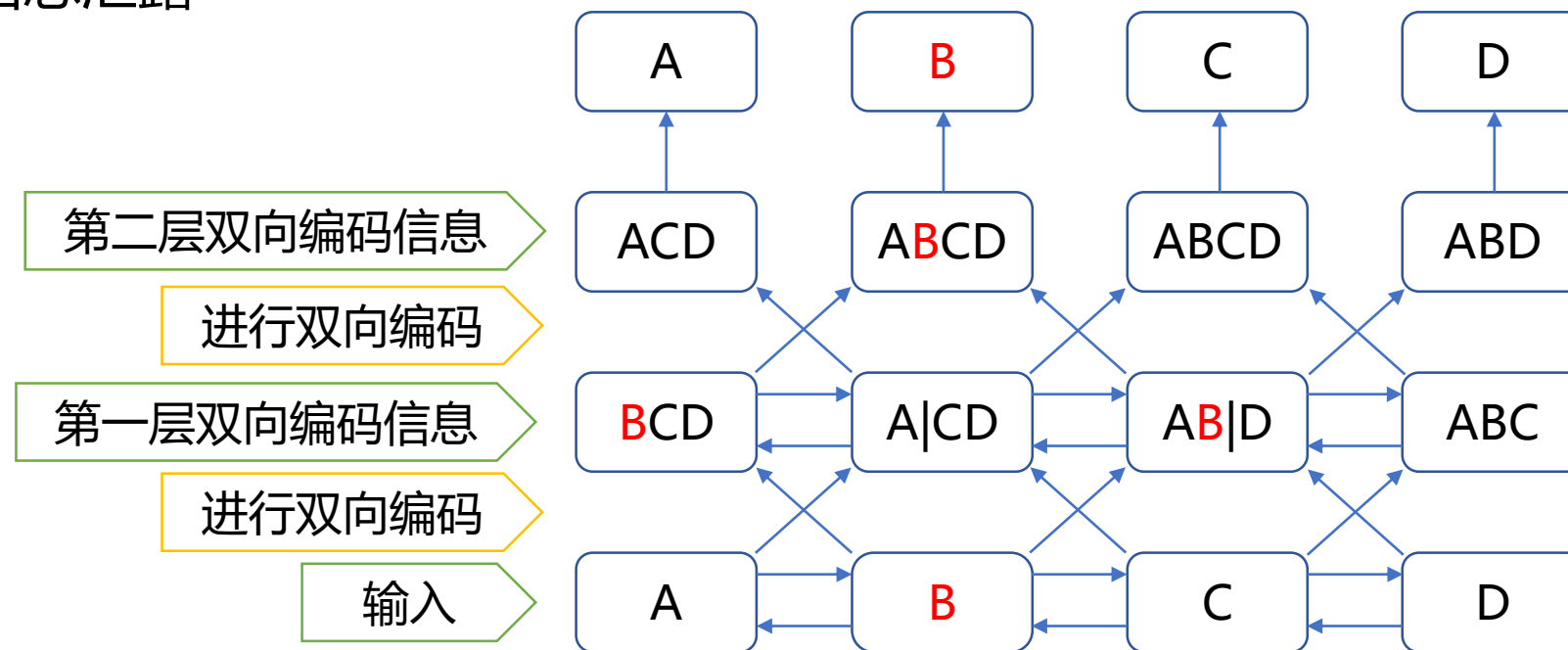
# 双向编码与网络深度的冲突

- 加深网络的层数可以带来更好的效果



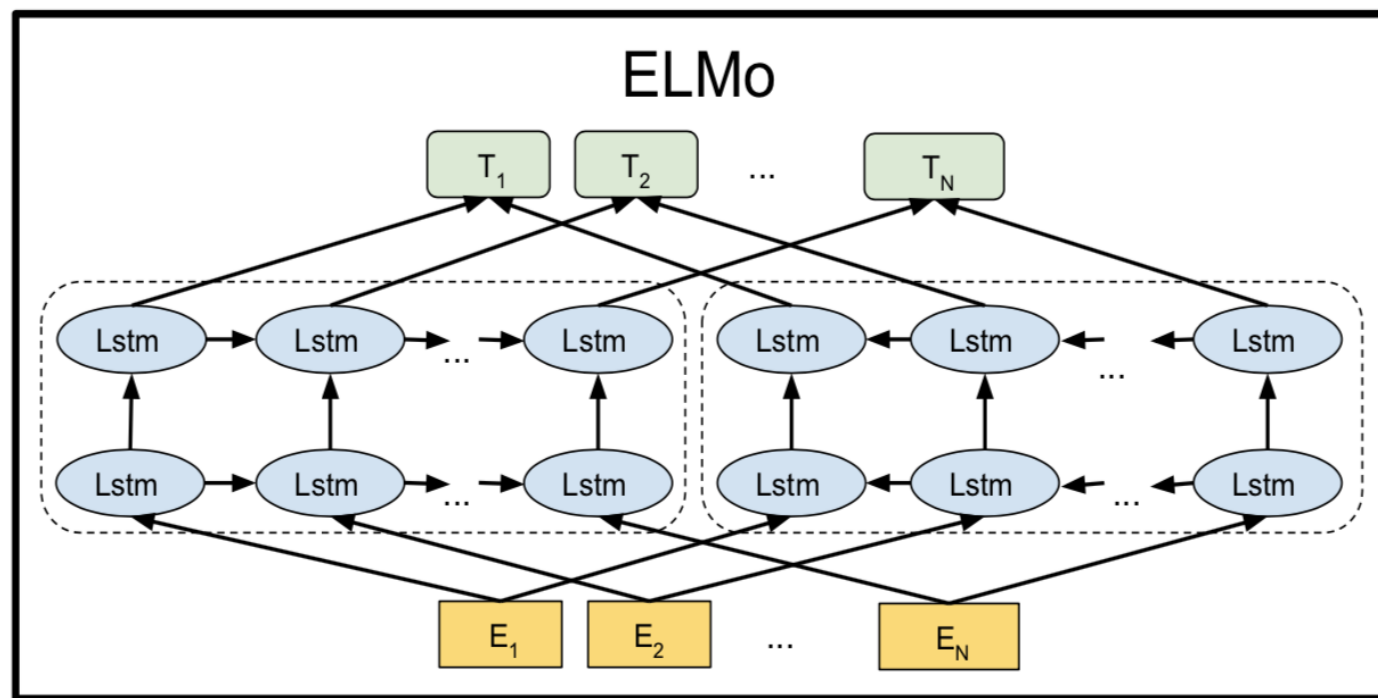
# 双向编码与网络深度的冲突

- 加深网络的层数可以带来更好的效果
- 深度的增加导致标签信息泄露



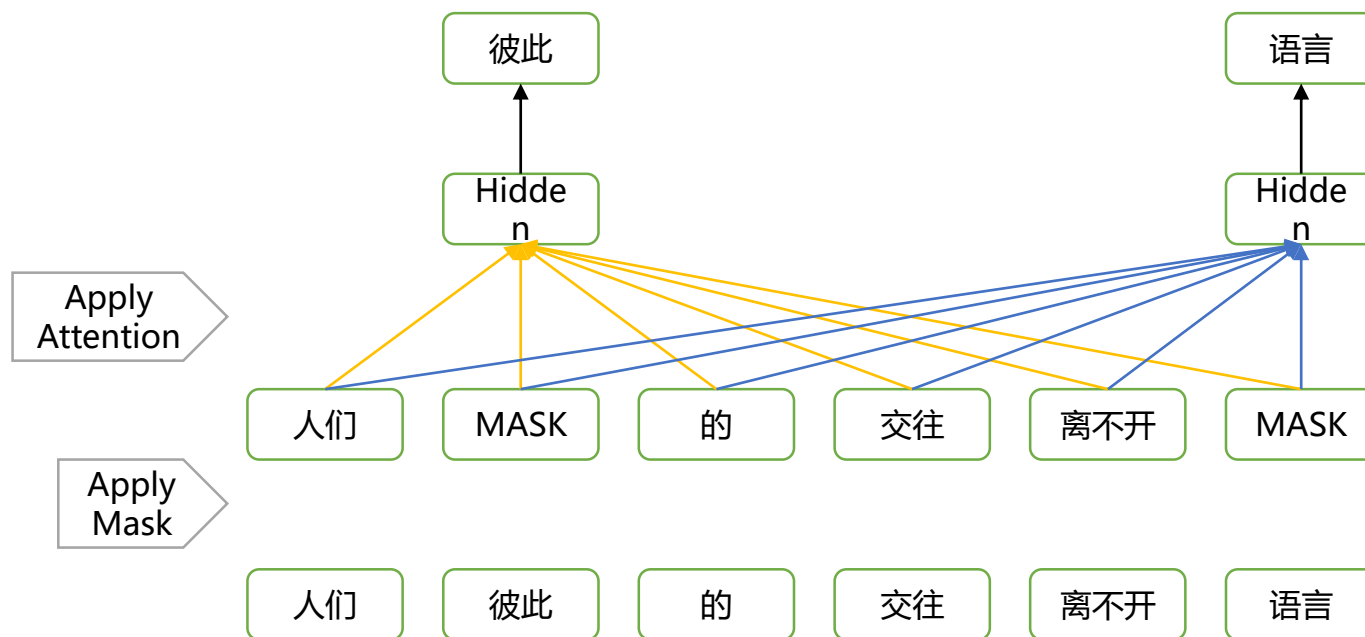
# 双向编码与网络深度的冲突

- 加深网络的层数可以带来更好的效果
- 深度的增加导致标签信息泄露
- 解决方式
  - 多层单向Rnn, 独立建模(ELMo)



# 双向编码与网络深度的冲突

- 加深网络的层数可以带来更好的效果
- 深度的增加导致标签信息泄露
- 解决方式
  - 多层单向RNN, 独立建模(ELMo)
  - Mask LM (BERT)





# 深度学习的模型可以从文本中学到什么？

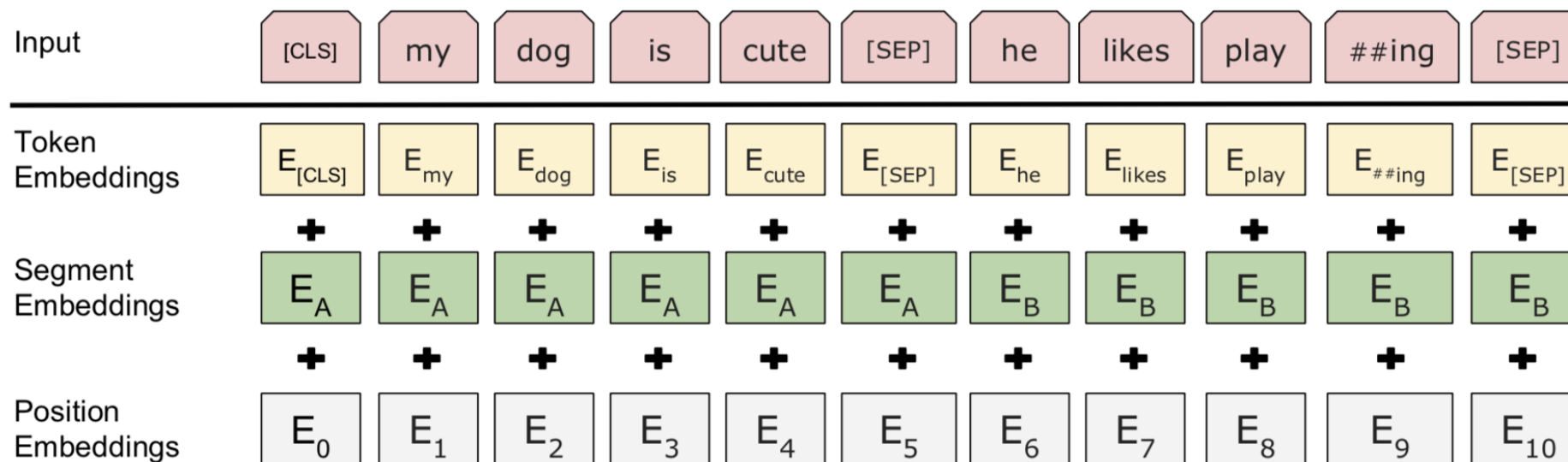
- 根据不同的语料以及任务，深度学习模型可以从数据中学到不同的内容：
  - 通用文本：句法结构、pos、词共现，层级关系等。
    - (Blevins et al., ACL 2018; Zhang & Bowman, 2018)
    - (Peters et al., NAACL 2018; EMNLP 2018)
  - 诗句：格律、节奏
    - (Lau et al., ACL 2018)
  - 结构化文本：格式、规则
    - (Yin and Neubig, ACL 2017)

# 大纲

- Part 1 语言模型简介
  - 文本表征
  - 语言模型
  - 深度学习与语言模型
- Part 2 BERT解析
  - BERT的基本结构
  - 模型模块详解
  - 源码中的细节
- Part 3 应用
  - 如何使用预训练模型
  - 实验心得

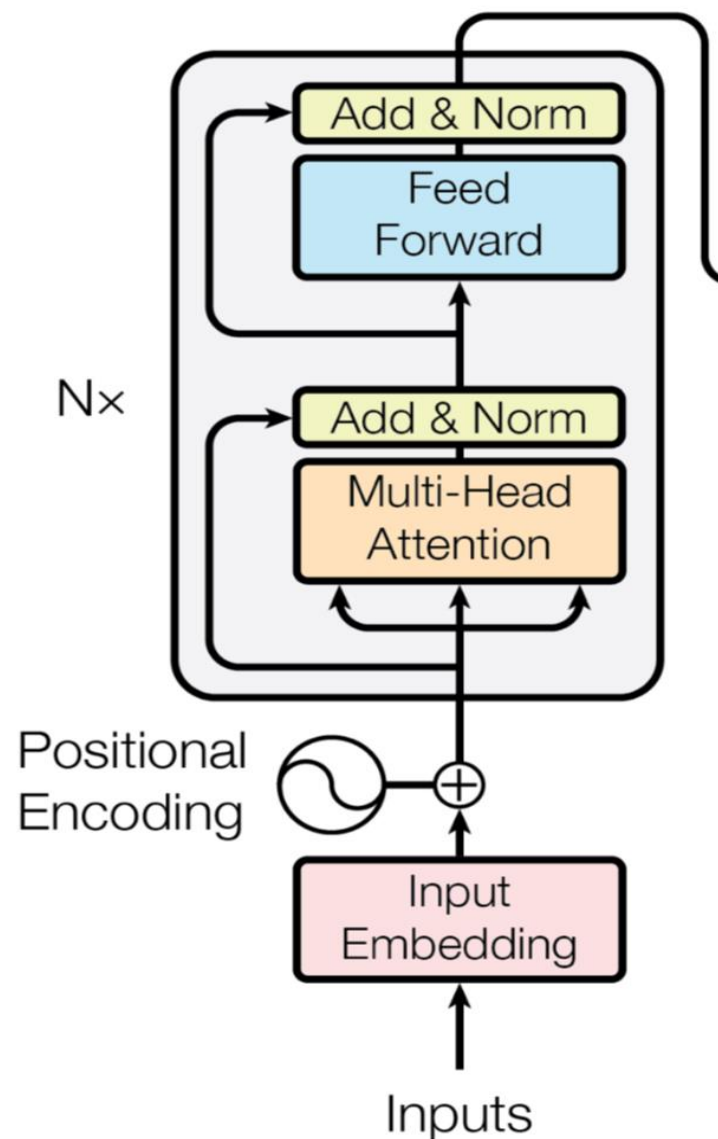
# BERT的整体结构

- Embedding
  - Word embedding
  - Position embedding
  - Type embedding



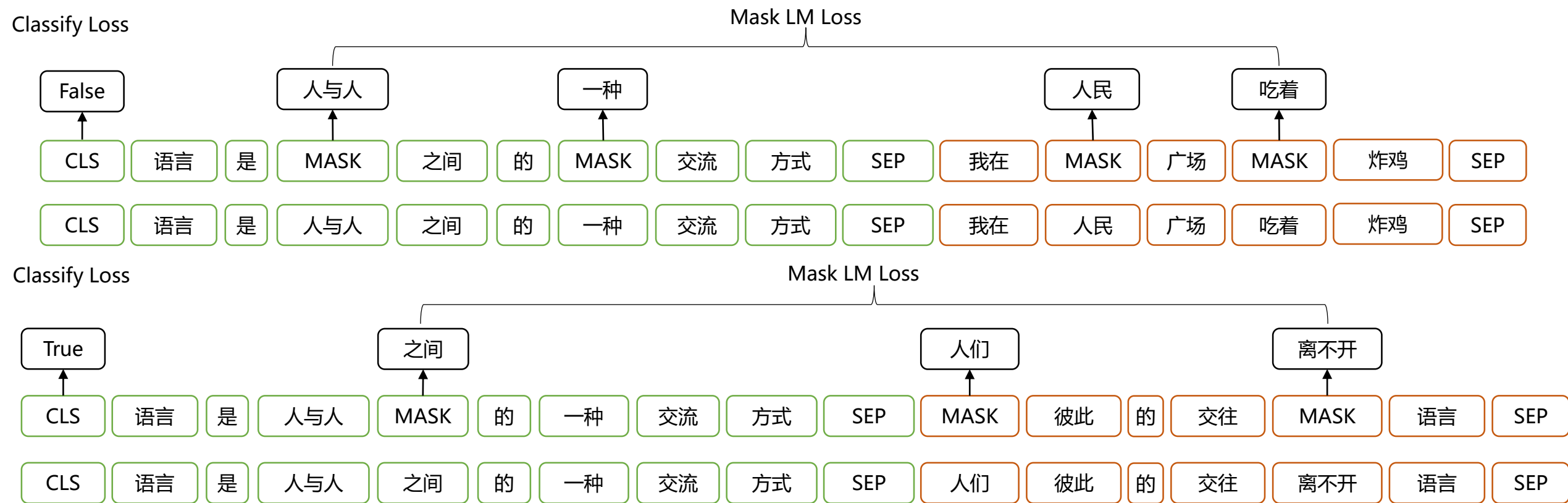
# BERT的整体结构

- Transformer Encoder
  - Multi-Head Attention
  - Feed Forward

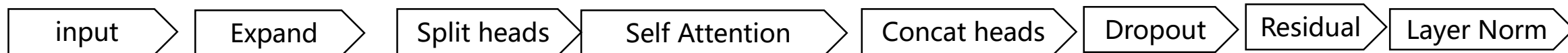
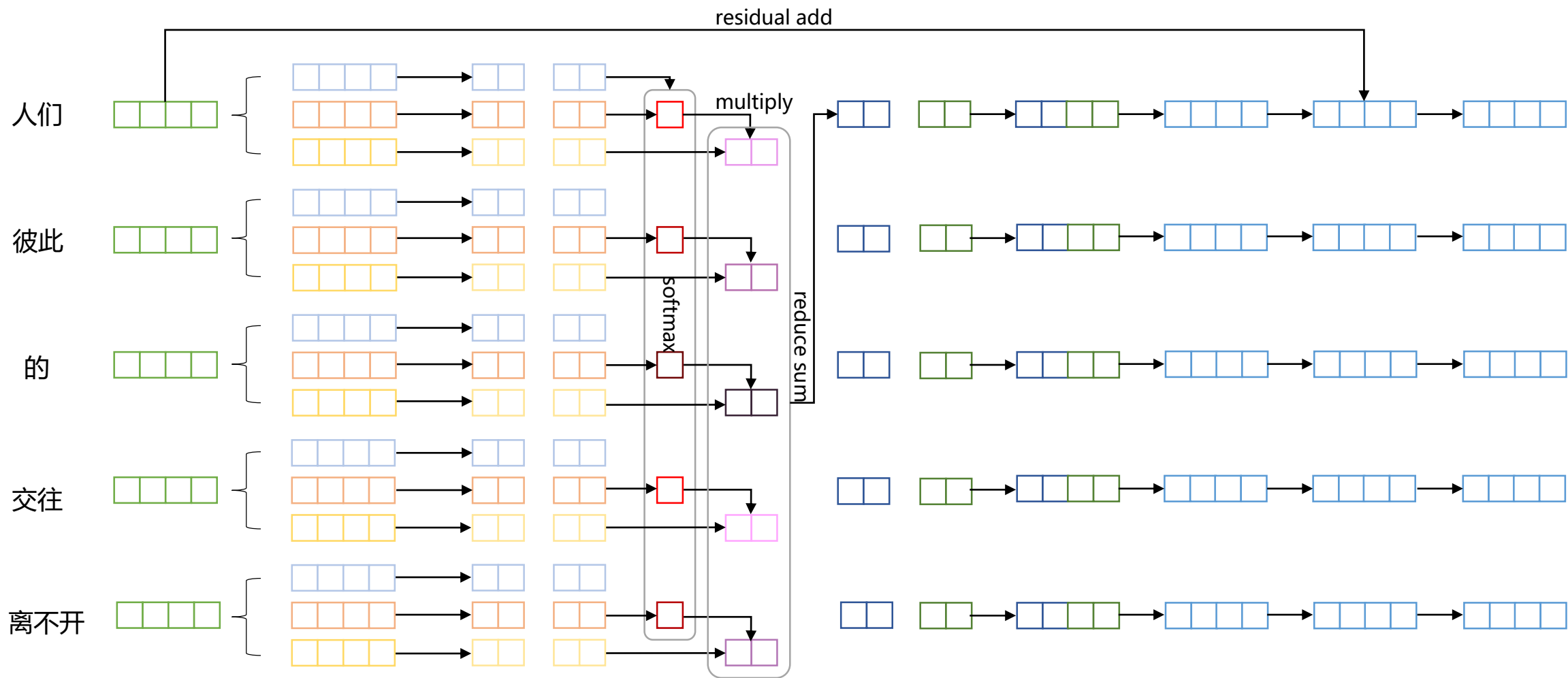


# BERT的整体结构

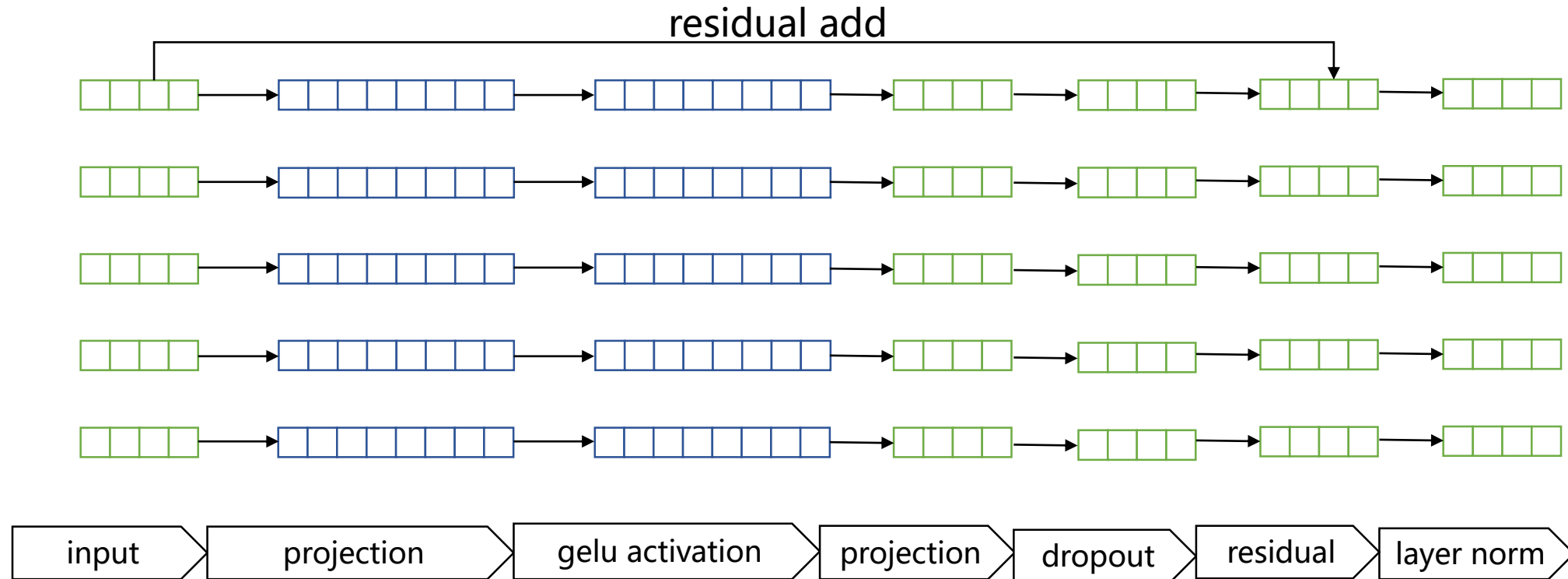
- Losses
  - Mask LM
  - Next sentence prediction



# Multi-Head Attention

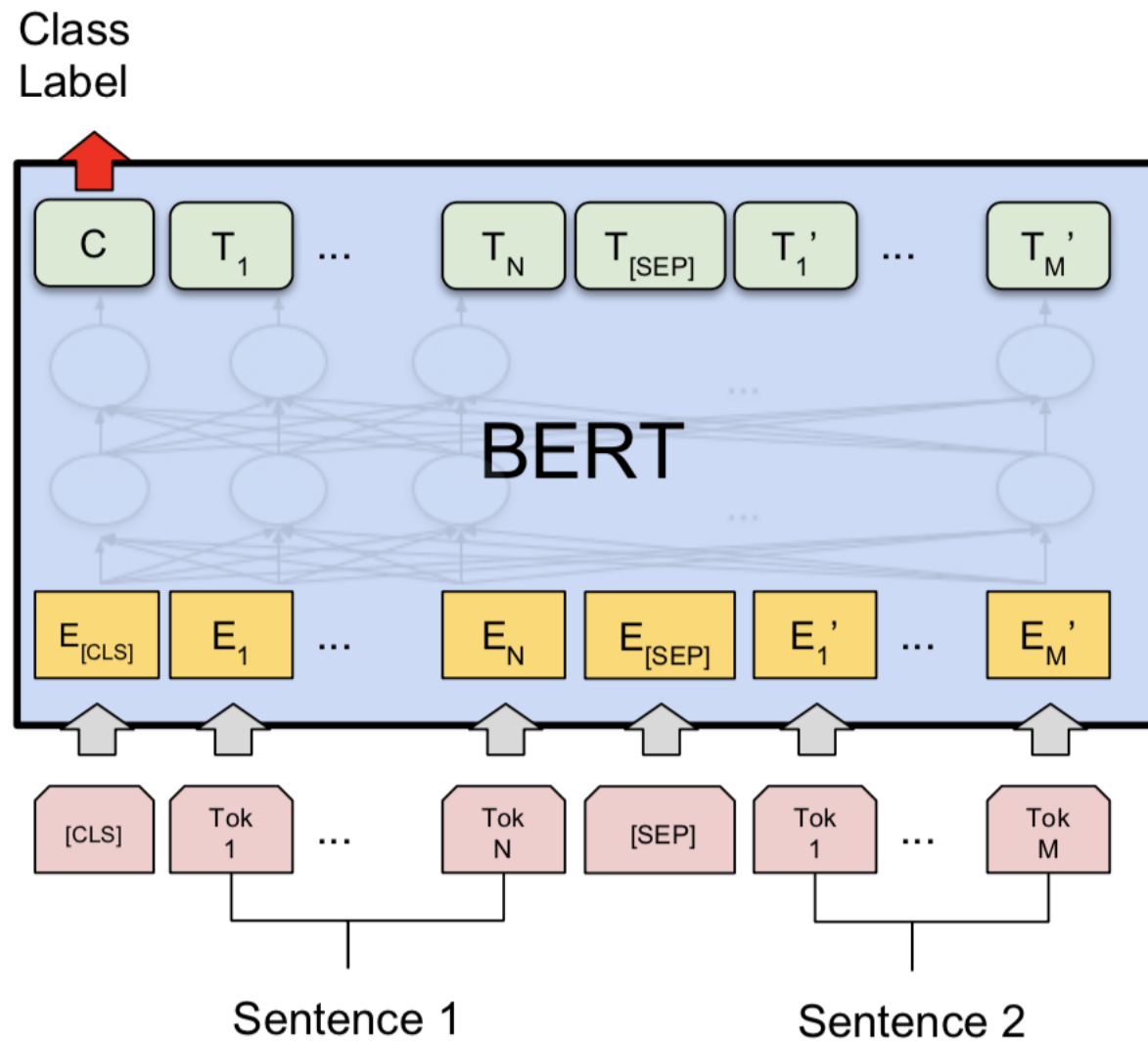


# Feed Forward



# Fine tuning方式

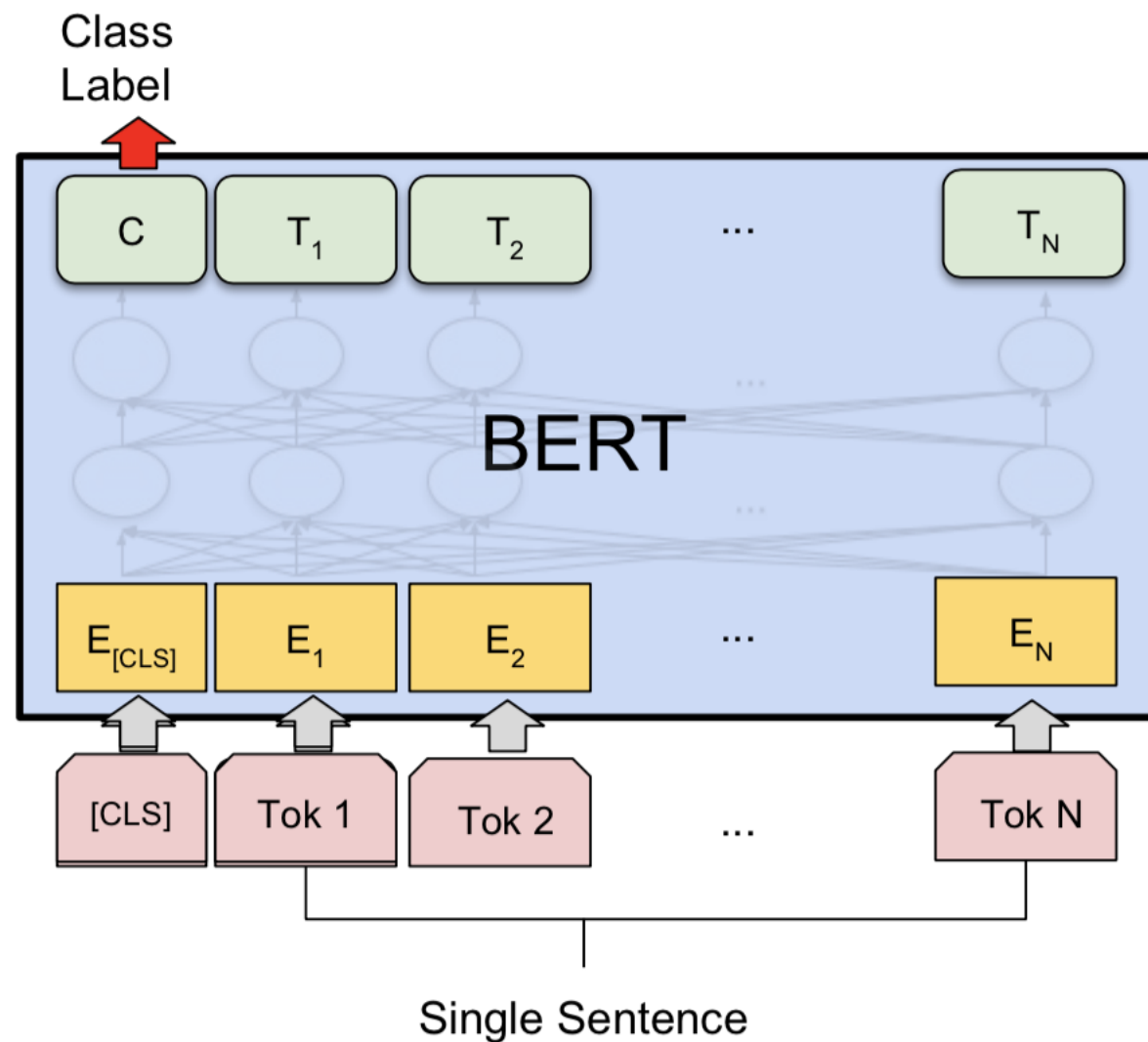
- 文本对分类任务





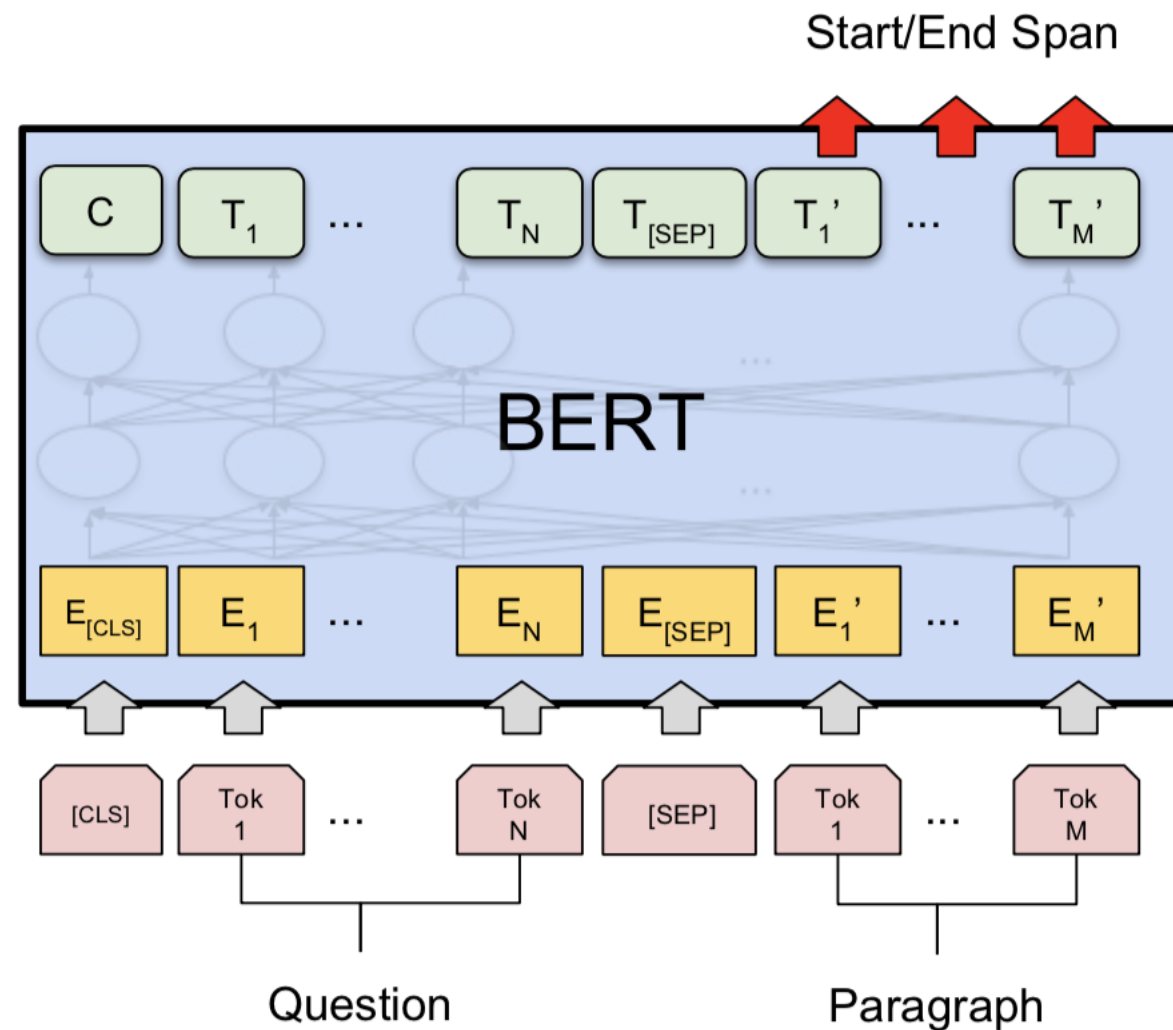
# Fine tuning方式

- 文本对分类任务
- 文本分类任务



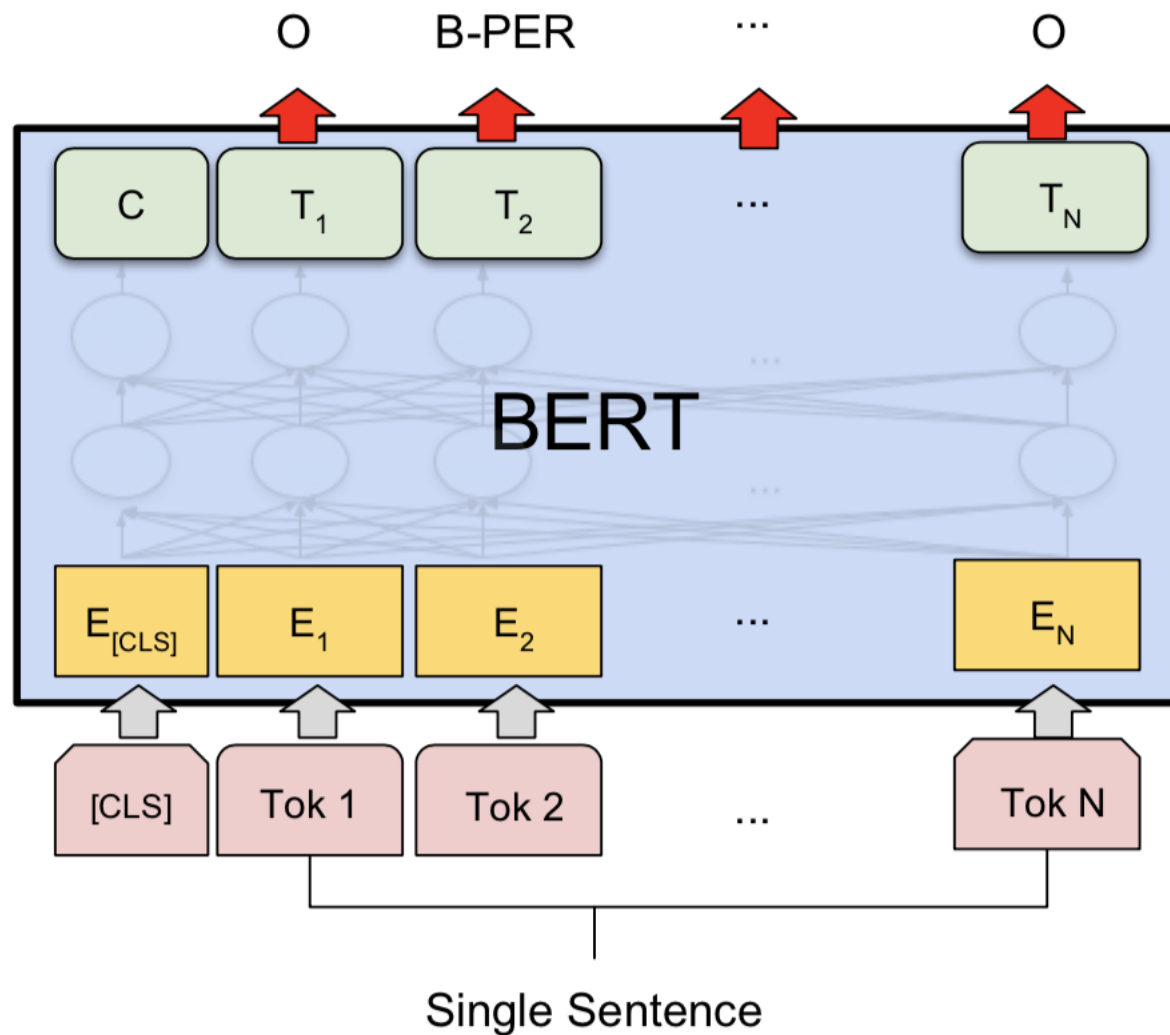
# Fine tuning方式

- 文本对分类任务
- 文本分类任务
- QA任务



# Fine tuning方式

- 文本对分类任务
- 文本分类任务
- QA任务
- 文本Tagging任务



# 一些细节

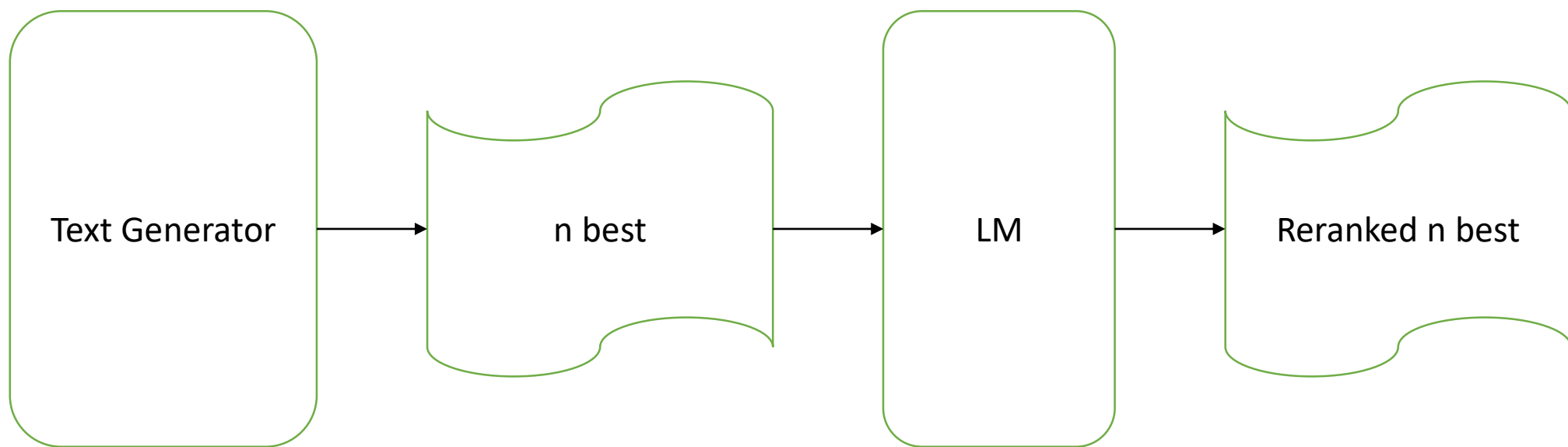
- 预训练数据的构建
  - 输入的两个部分都是包含完整句子
  - 同一句中不会同时Mask同一个词
  - 非连续的句子都是取自不同文档
  - 有一定的概率让输入的有效文本长度小于输入全长（利用padding补齐）
  - 对于超过全长的文本，随机从第一部分部分和第二部分部分的头和尾削减
- 模型
  - Encoder LM任务的输出后经过一个gelu非线性层再进行LM Loss的计算；
  - Classify任务的输出后经过一个tanh非线性层再进行的二分类

# 大纲

- Part 1 语言模型简介
  - 文本表征
  - 语言模型
  - 深度学习与语言模型
- Part 2 BERT解析
  - BERT的基本结构
  - 模型模块详解
  - 源码中的细节
- Part 3 应用
  - 如何使用预训练模型
  - 实验心得

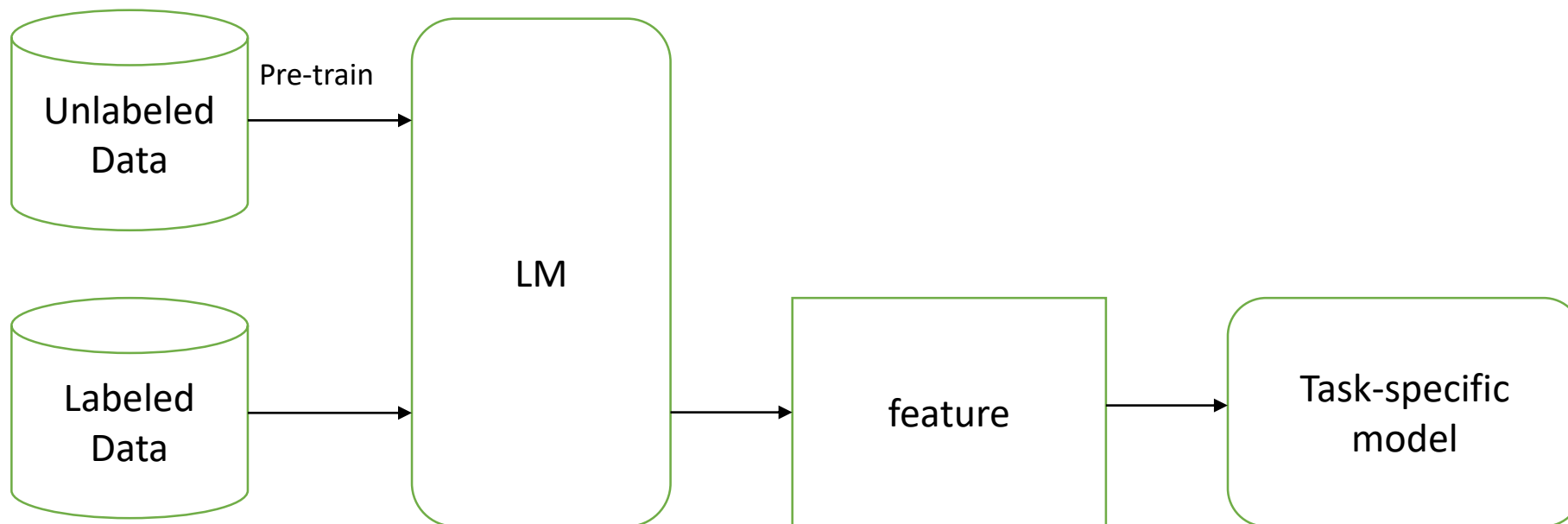
# 预训语言模型探索

- 语言模型可以用来做什么?
  - 重排: ASR, 生成等



# 预训语言模型探索

- 语言模型可以用来做什么?
  - 重排: ASR, 生成等
  - 迁移: ELMo, GPT, BERT



# 论文中结果

Method	MNLI-m	MNLI-mm	SNLI	SciTail	QNLI	RTE
ESIM + ELMo [44] (5x)	-	-	<u>89.3</u>	-	-	-
CAFE [58] (5x)	80.2	79.0	<u>89.3</u>	-	-	-
Stochastic Answer Network [35] (3x)	<u>80.6</u>	<u>80.1</u>	-	-	-	-
CAFE [58]	78.7	77.9	88.5	<u>83.3</u>		
GenSen [64]	71.4	71.3	-	-	<u>82.3</u>	59.2
Multi-task BiLSTM + Attn [64]	72.2	72.1	-	-	82.1	<b>61.7</b>
Finetuned Transformer LM (ours)	<b>82.1</b>	<b>81.4</b>	<b>89.9</b>	<b>88.3</b>	<b>88.1</b>	56.0

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	Average
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.9	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	88.1	91.3	45.4	80.0	82.3	56.0	75.2
BERT <sub>BASE</sub>	84.6/83.4	71.2	90.1	93.5	52.1	85.8	88.9	66.4	79.6
BERT <sub>LARGE</sub>	<b>86.7/85.9</b>	<b>72.1</b>	<b>91.1</b>	<b>94.9</b>	<b>60.5</b>	<b>86.5</b>	<b>89.3</b>	<b>70.1</b>	<b>81.9</b>



# 实验观察

- Elmo
  - 百度百科+中文维基, 18亿tokens
  - batch size: 64, length:128, 256
- GPT
  - 18亿tokens
  - batch size: 24, length:128, 256
- BERT(实验)
  - 1亿tokens
  - batch size: 32, length:512

# 实验观察

- 一些观察
  - 在QA任务上效果明显
  - 在短文本分类上效果不及预期
  - BERT在不按完整句子构建span时不按完整句子构建span
    - 预测是否连续的任务收敛很快
    - 短句很多的时候ppl收敛会慢

# Q&A



追一科技公众号



追一招聘公众号