机器学习与数据挖掘

Machine Learning & Data Mining

权小军 教授

中山大学数据科学与计算机学院

quanxj3@mail.sysu.edu.cn

Preface

Preface

2018年BERT诞生，横扫11大NLP任务

SQuAD1.1 Leaderboard

Since the release of SQuAD1.0, the community has made rapid progress, with the best models now rivaling human performance on the task. Here are the ExactMatch (EM) and F1 scores evaluated on the test set of v1.1.

Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar et al. '16)	82.304	91.221
1 Oct 05, 2018	BERT (ensemble) *Google A.I.*	**87.433**	**93.160**
2 Oct 05, 2018	BERT (single model) *Google A.I.*	85.083	91.835
2 Sep 09, 2018	nlnet (ensemble) *Microsoft Research Asia*	85.356	91.202
2 Sep 26, 2018	nlnet (ensemble) *Microsoft Research Asia*	85.954	91.677

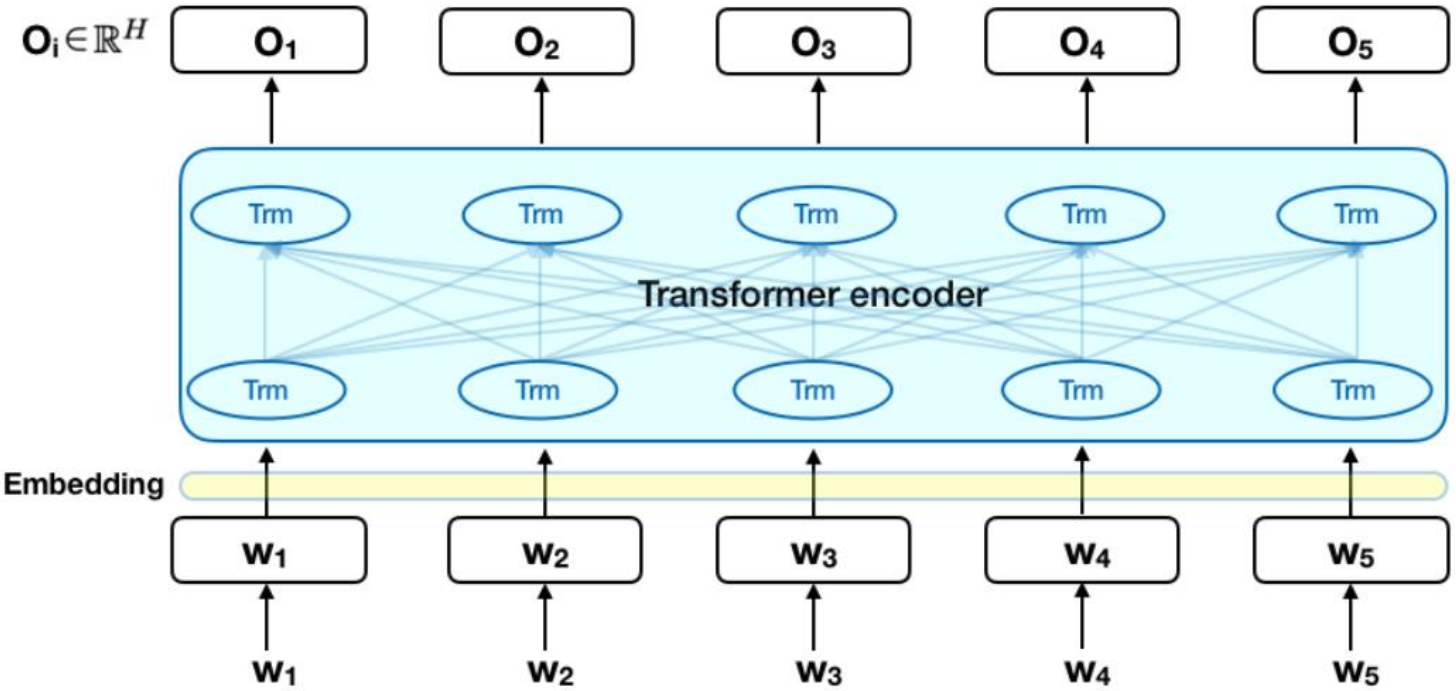
System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Average** -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

System	Dev	Test
ESIM+GloVe	51.9	52.7
ESIM+ELMo	59.1	59.2
OpenAI GPT	-	78.0
$\text{BERT}_{\text{BASE}}$	81.6	-
$\text{BERT}_{\text{LARGE}}$	**86.6**	**86.3**
Human (expert)†	-	85.0
Human (5 annotations)†	-	88.0

Preface

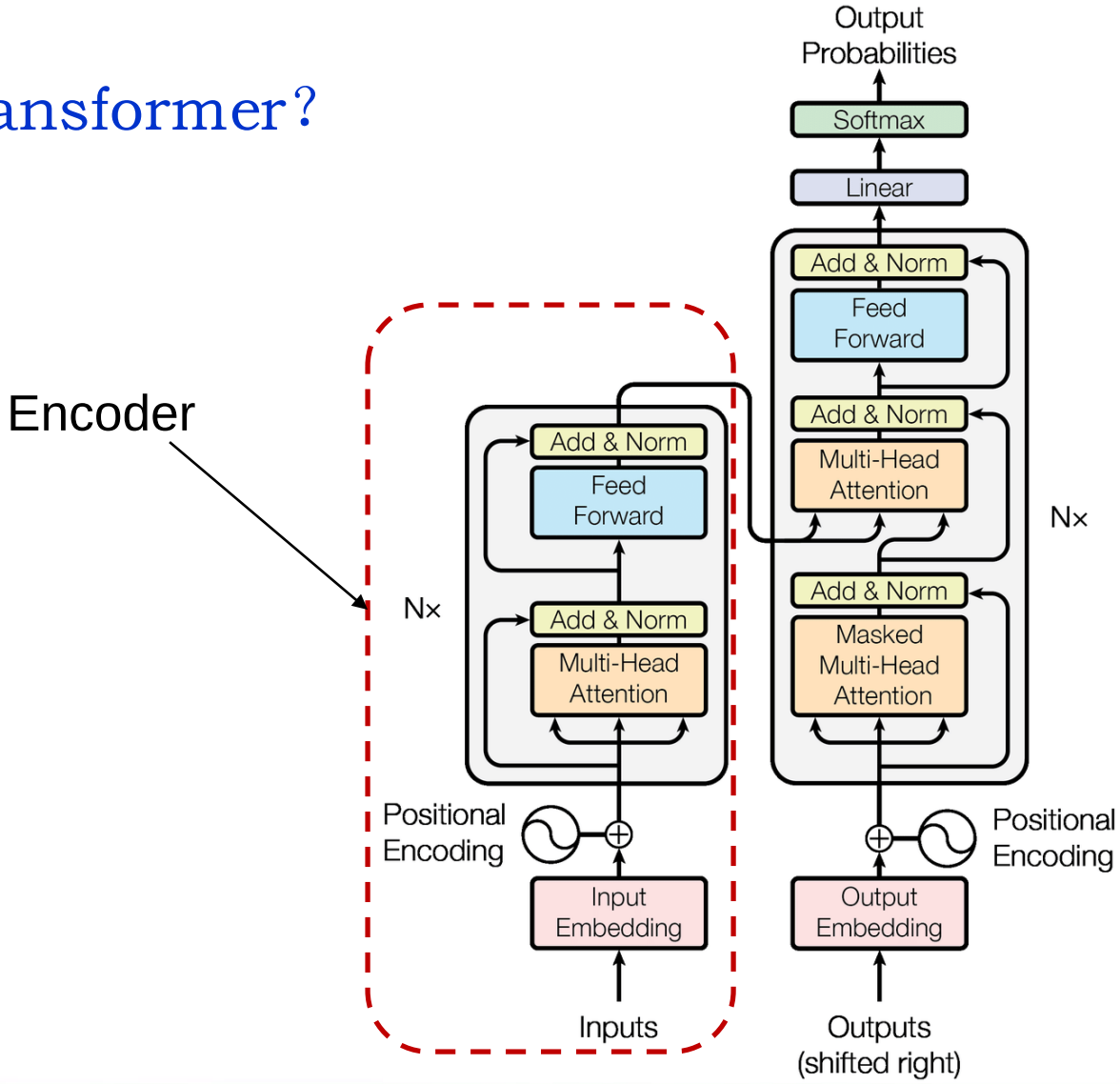
什么是BERT？

BERT: Bidirectional Encoder Representations from Transformers



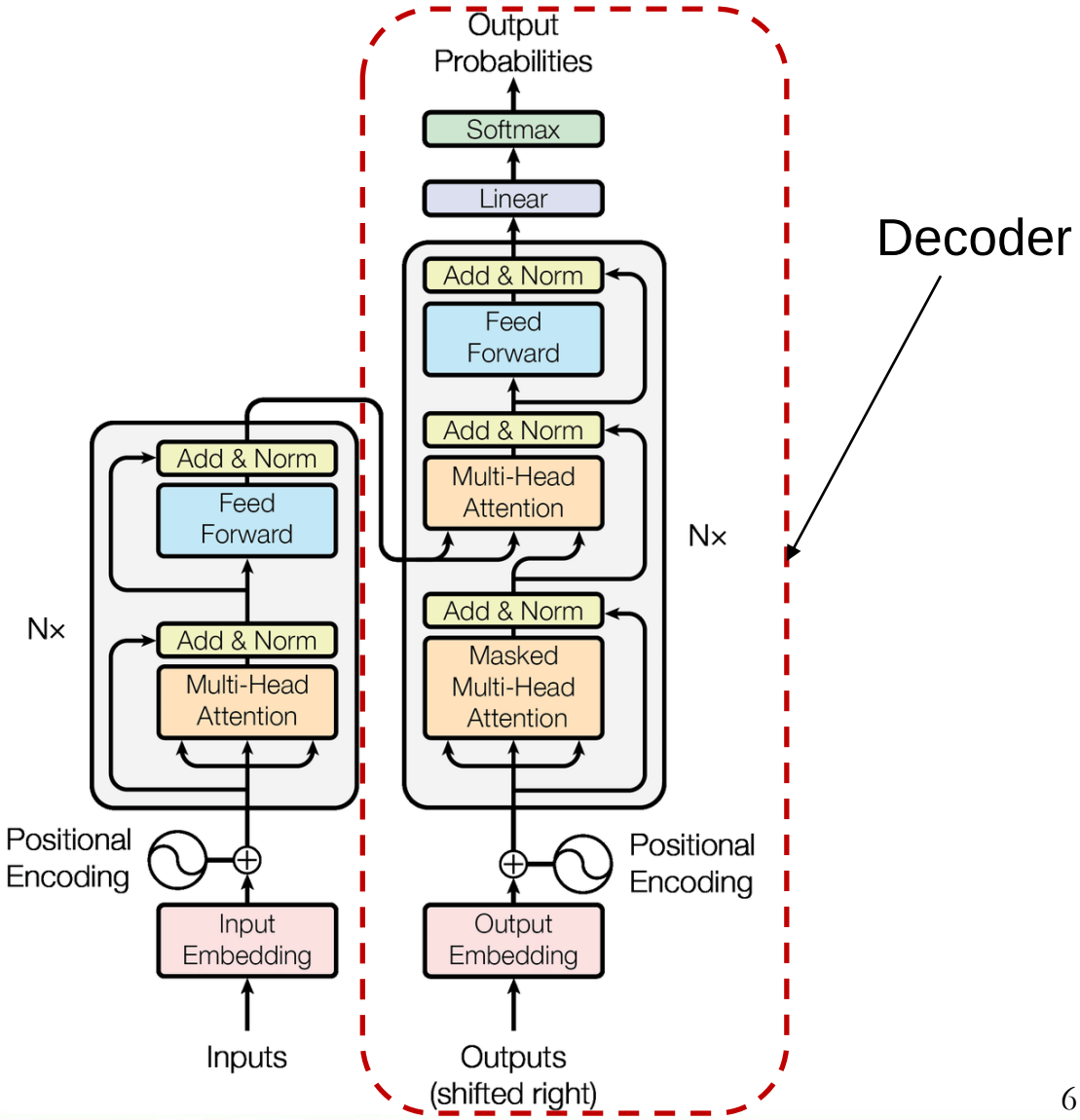
Preface

什么是Transformer？



Preface

什么是Transformer?

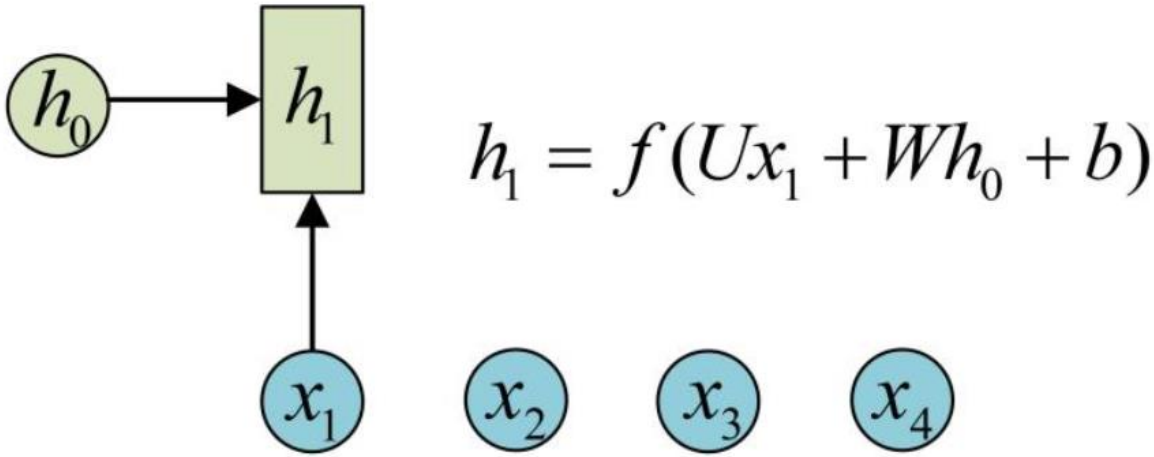


Lecture 12 Transformer

12.1 RNN and Attention

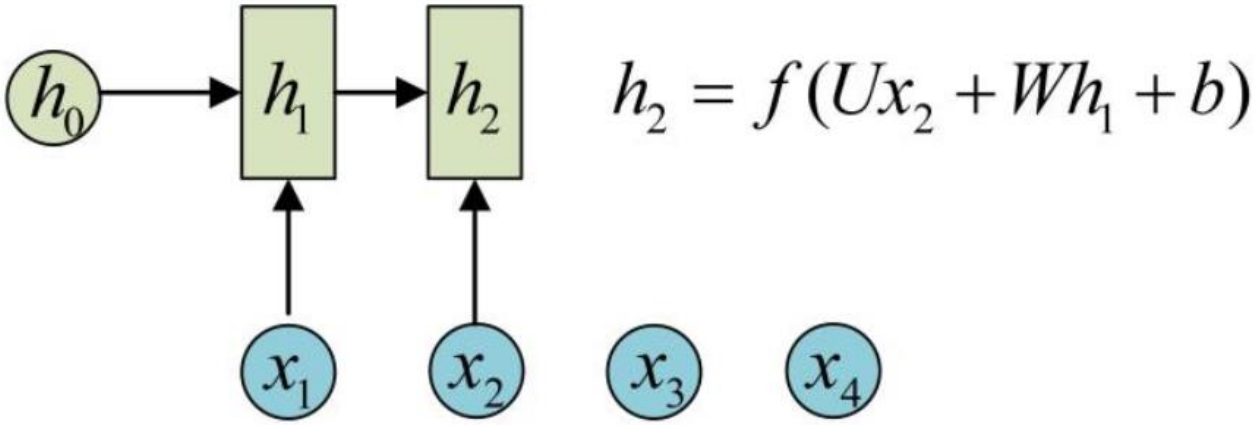
RNN and Attention

- 循环神经网络主要用于处理（变长）序列数据



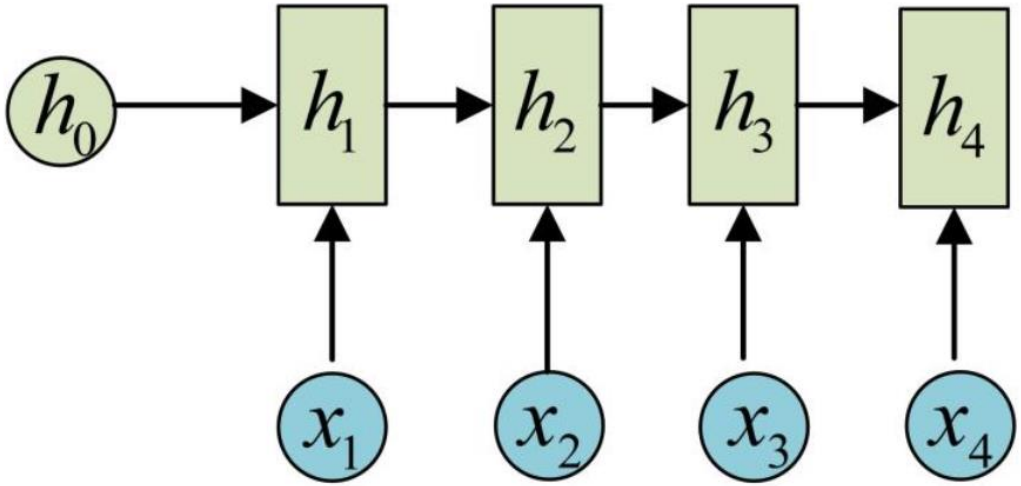
RNN and Attention

- 循环神经网络主要用于处理（变长）序列数据



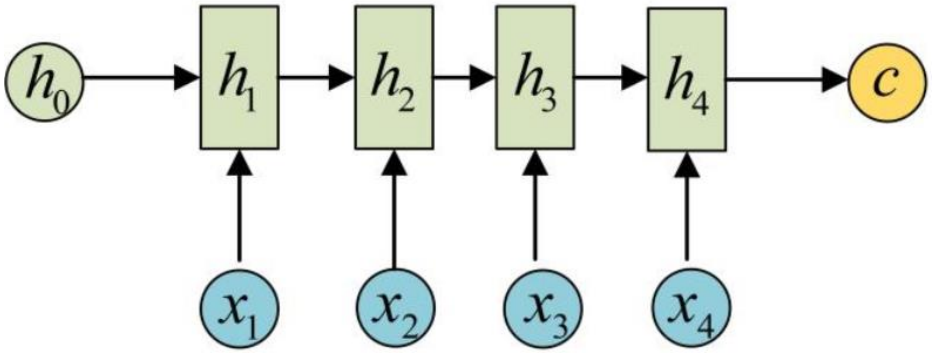
RNN and Attention

- 循环神经网络主要用于处理（变长）序列数据



RNN and Attention

- Encoder Decoder结构——处理输入n输出m的问题



（1）$c = h_4$

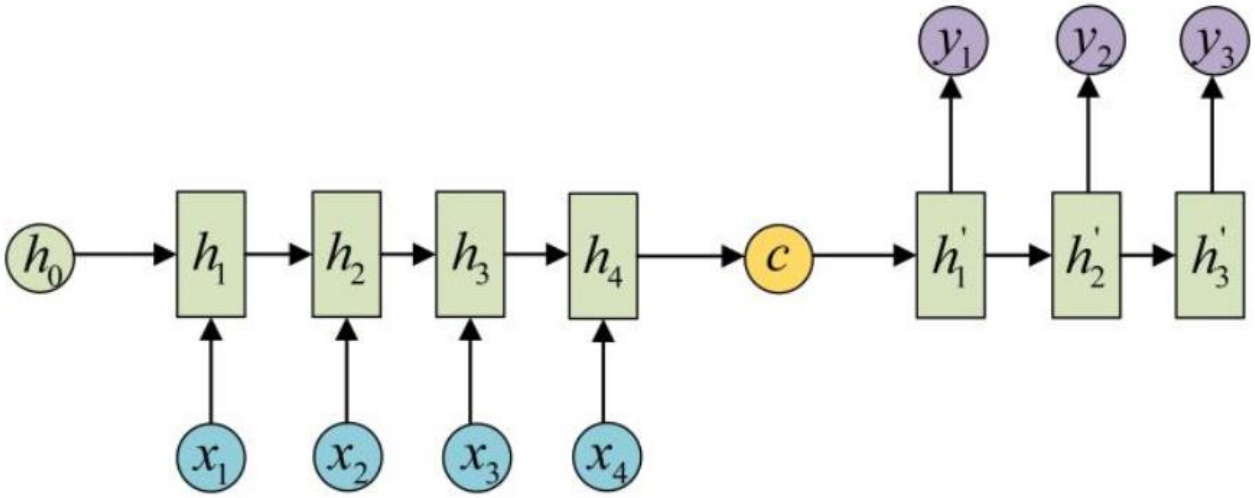
（2）$c = q(h_4)$

（3）$c = q(h_1, h_2, h_3, h_4)$

首先将输入编码为一个context向量c，c的计算可以有很多方法，最简单的是取最后一个隐状态

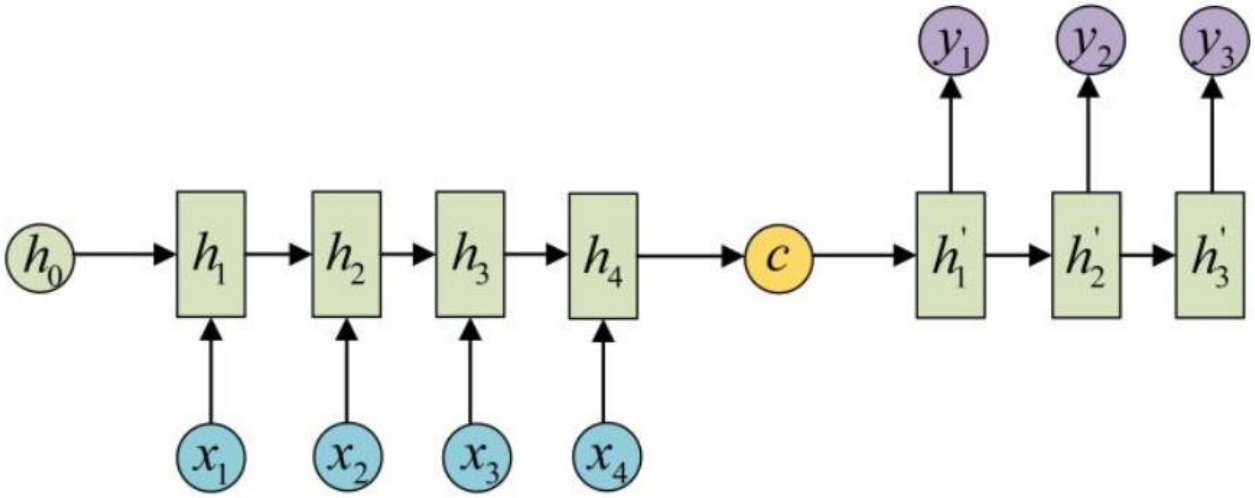
RNN and Attention

- Encoder-Decoder结构



RNN and Attention

- Attention机制

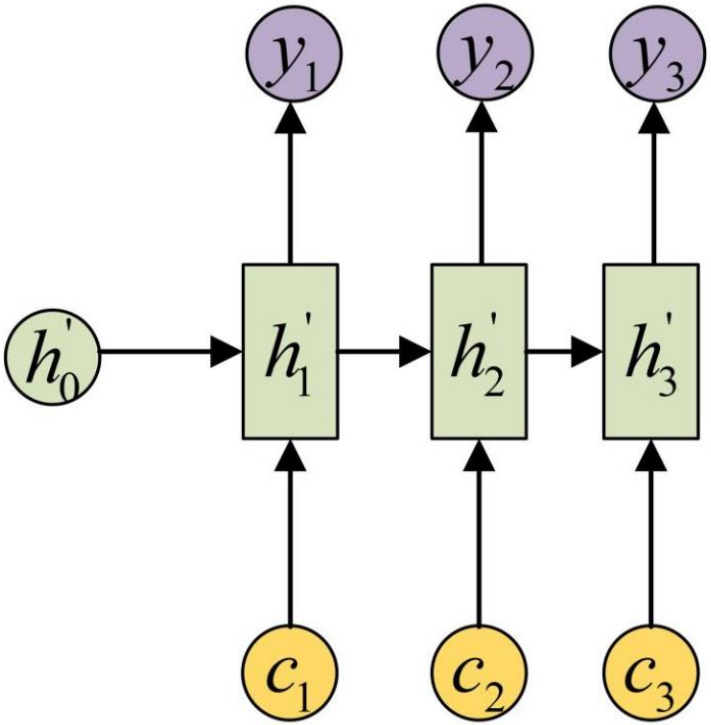


在Encoder-Decoder结构中，Encoder把所有的输入序列都编码成一个统一的语义特征c再解码，因此，c中必须包含原始序列中的所有信息，句子的长度就成了限制模型性能的瓶颈。如机器翻译问题，当要翻译的句子较长时，一个c可能存不下那么多信息，就会造成翻译精度的下降。

RNN and Attention

- Attention机制

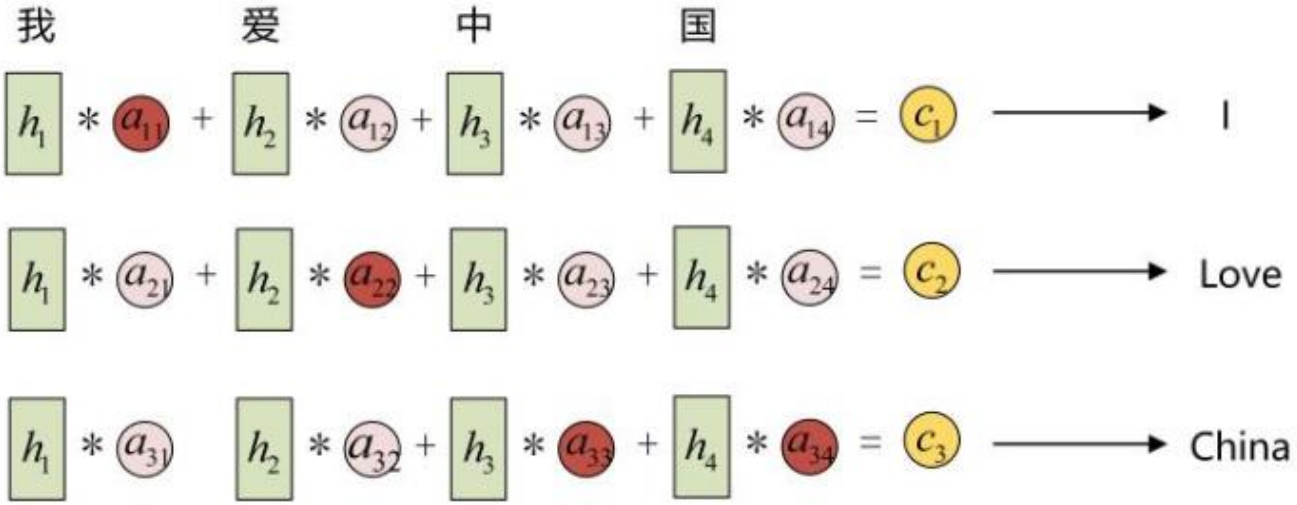
Attention机制通过在每个时间输入不同的c来解决这个问题



RNN and Attention

- Attention机制

每一个c会自动去选取与当前所要输出的y最合适的上下文信息。具体来说，我们用a_{ij}衡量Encoder中第j阶段的h_j和解码时第i阶段的相关性，最终Decoder中第i阶段的输入的上下文信息c_i 就来自于所有h_j 对a_{ij}的加权和。



12.2 Transformer

Transformer

- 循环神经网络的缺点

  1、循环结构难以并行化

  2、难以捕捉长期依赖

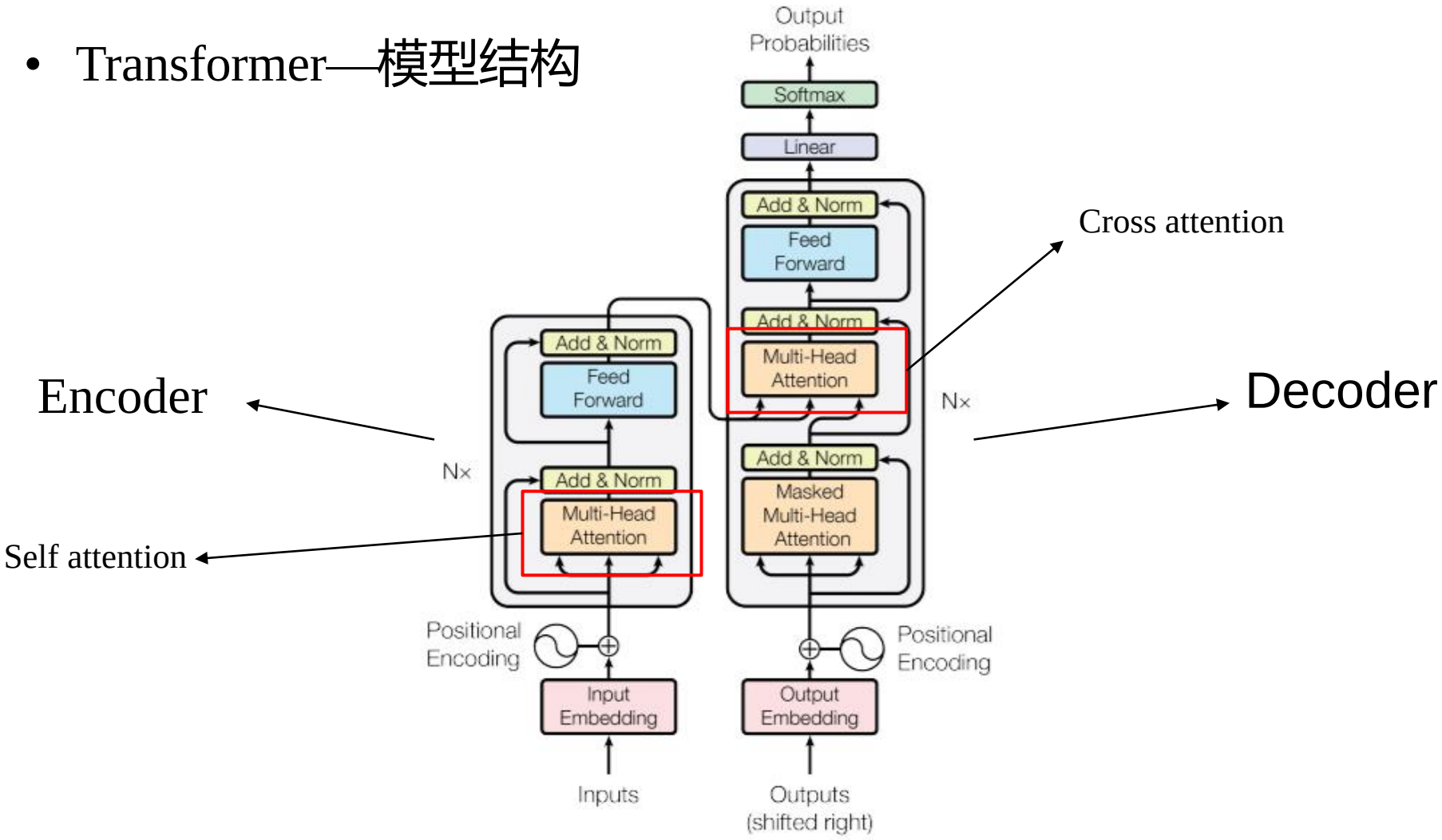
Transformer

- Transformer—完全基于注意力机制

Transformer

Vaswani et al., Attention Is All You Need. NIPS 2017

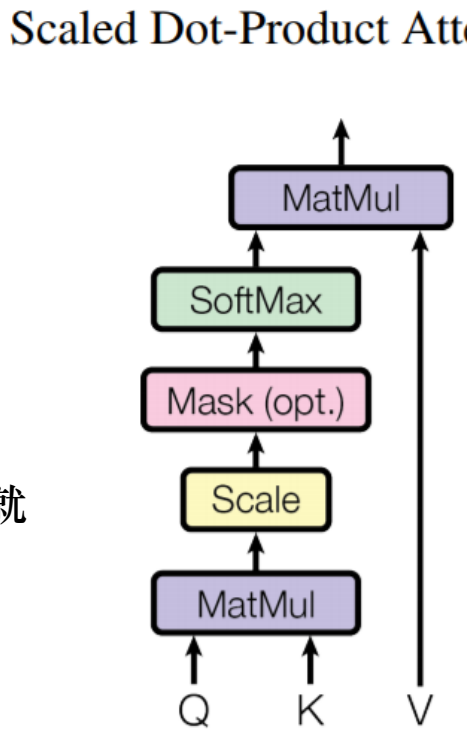
- Transformer—模型结构



Self attention

Transformer

- Transformer——Attention模块



当Q，K，V来自同一个序列时就是self attention，当来自两个不同的序列时就是cross attention

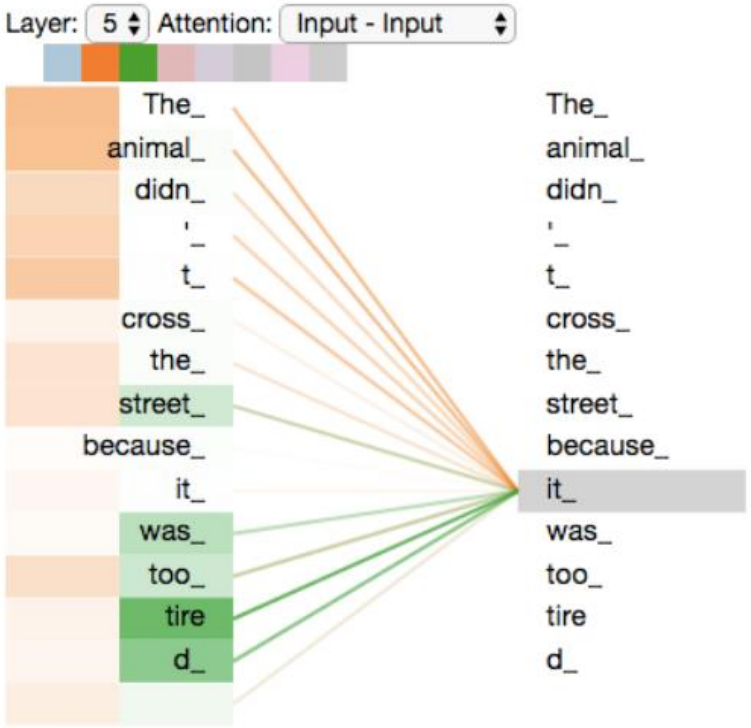
能够完全并行

Attention权重，在自中自注意力中就是输入token之间的相关性

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

Transformer

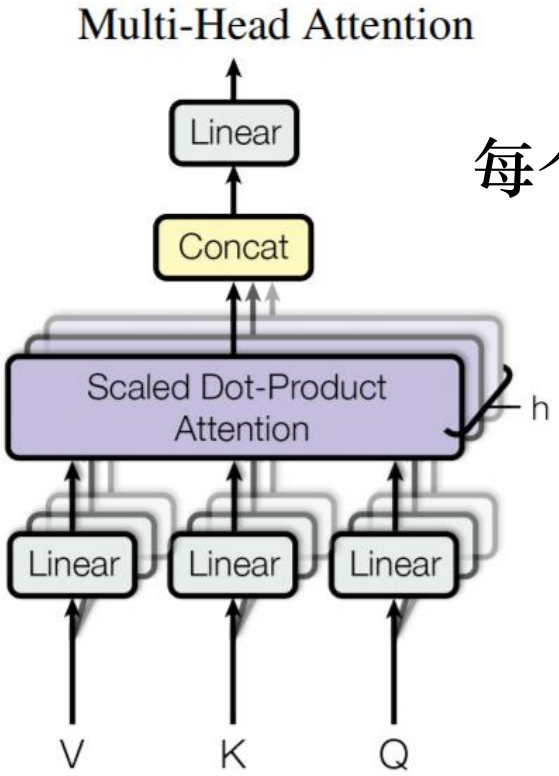
- Transformer——Attention模块



多头自注意力可视化

Transformer

- Transformer——MultiHead机制



每个head可以学习不同的知识

$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$

$$\text{where head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Transformer

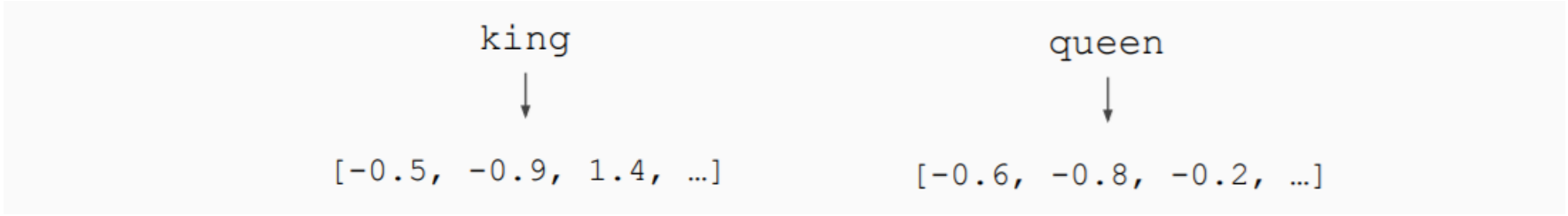
- 机器翻译上取得sota的结果

Model	BLEU		Training Cost (FLOPs)	
	EN-DE	EN-FR	EN-DE	EN-FR
ByteNet [18]	23.75			
Deep-Att + PosUnk [39]		39.2		$1.0 \cdot 10^{20}$
GNMT + RL [38]	24.6	39.92	$2.3 \cdot 10^{19}$	$1.4 \cdot 10^{20}$
ConvS2S [9]	25.16	40.46	$9.6 \cdot 10^{18}$	$1.5 \cdot 10^{20}$
MoE [32]	26.03	40.56	$2.0 \cdot 10^{19}$	$1.2 \cdot 10^{20}$
Deep-Att + PosUnk Ensemble [39]		40.4		$8.0 \cdot 10^{20}$
GNMT + RL Ensemble [38]	26.30	41.16	$1.8 \cdot 10^{20}$	$1.1 \cdot 10^{21}$
ConvS2S Ensemble [9]	26.36	**41.29**	$7.7 \cdot 10^{19}$	$1.2 \cdot 10^{21}$
Transformer (base model)	27.3	38.1	**$3.3 \cdot 10^{18}$**	
Transformer (big)	**28.4**	**41.8**	$2.3 \cdot 10^{19}$	

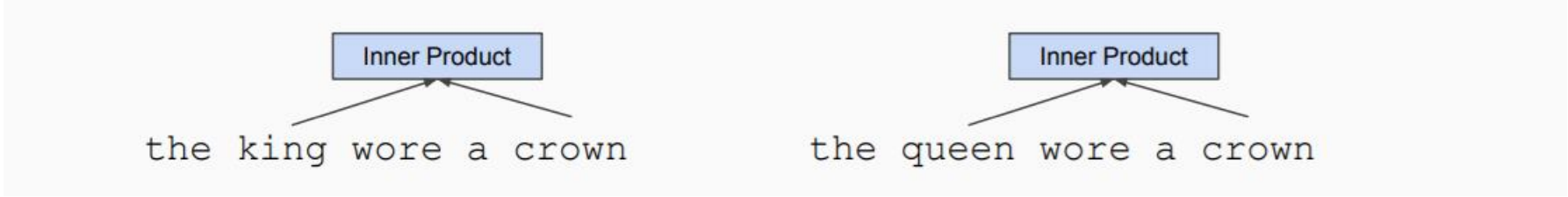
12.3 BERT

BERT

- 背景：词向量



词向量通常在语料上根据共现关系预训练得到



BERT

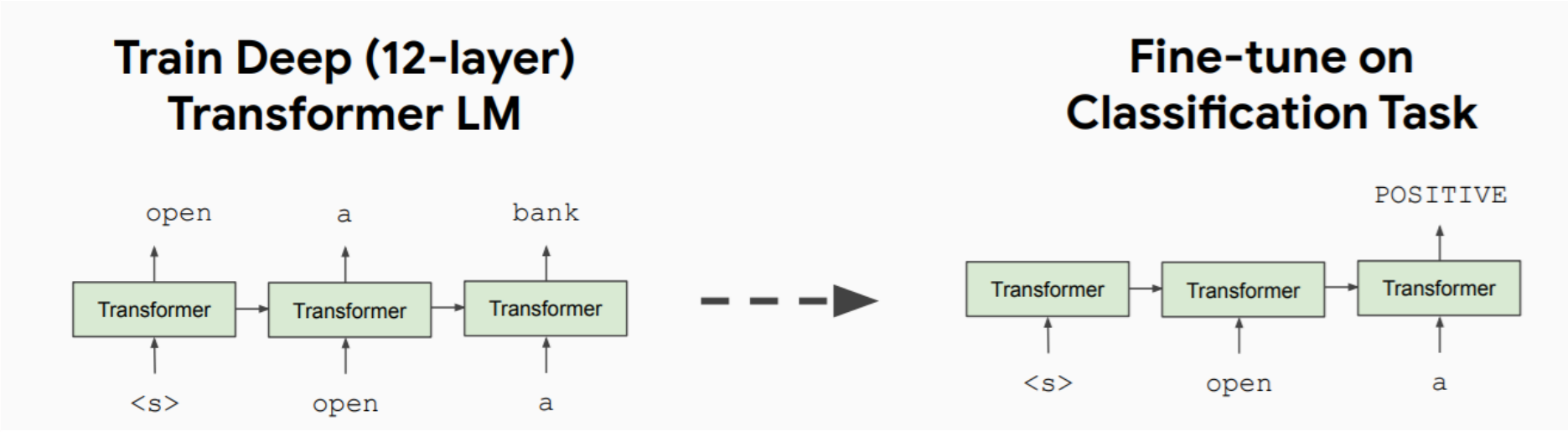
- 背景

词向量是上下文无关的，无法解决歧义

解决方案：训练上下文相关的表示

BERT

- GPT： Generative Pre-training Transformer



BERT

- GPT： Generative Pre-training Transformer

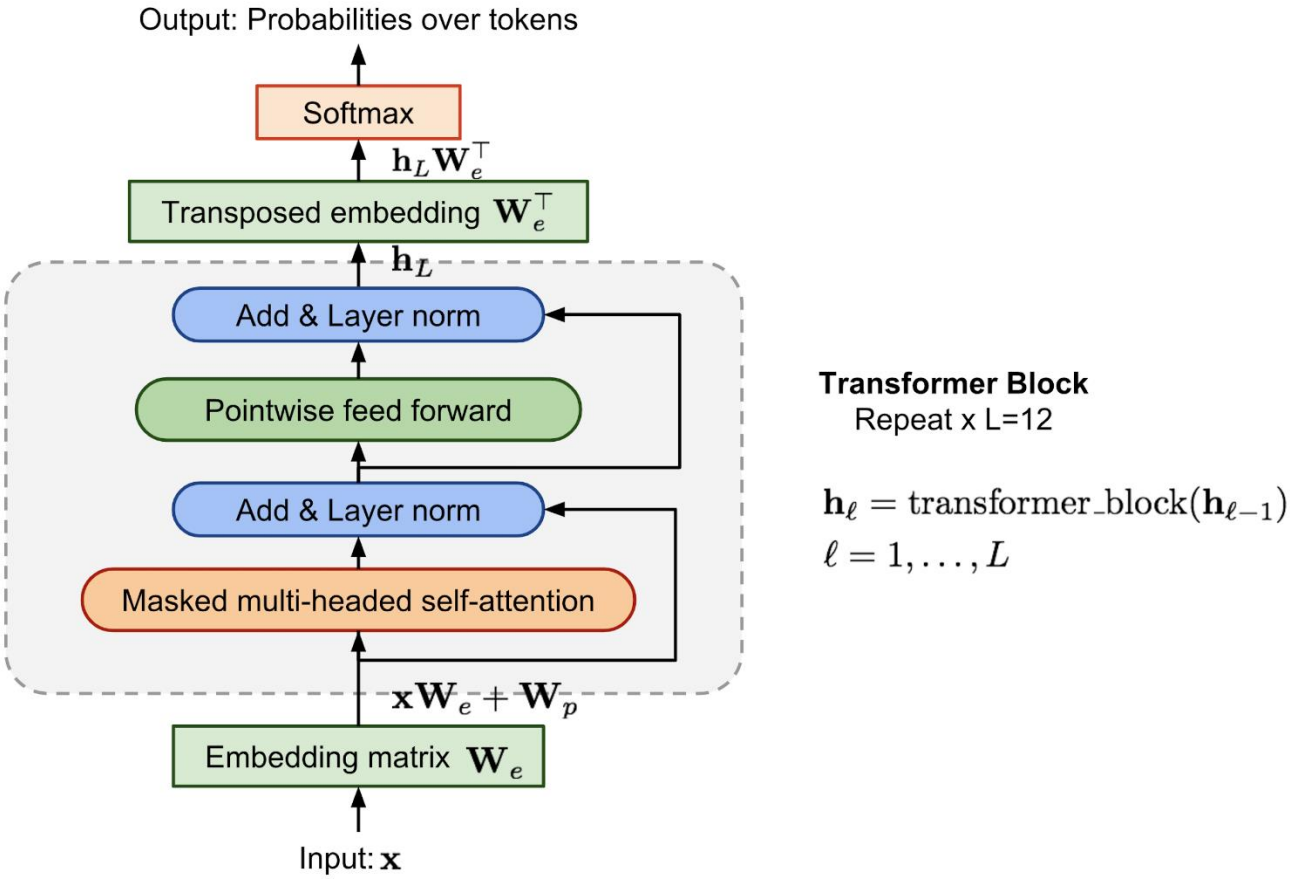
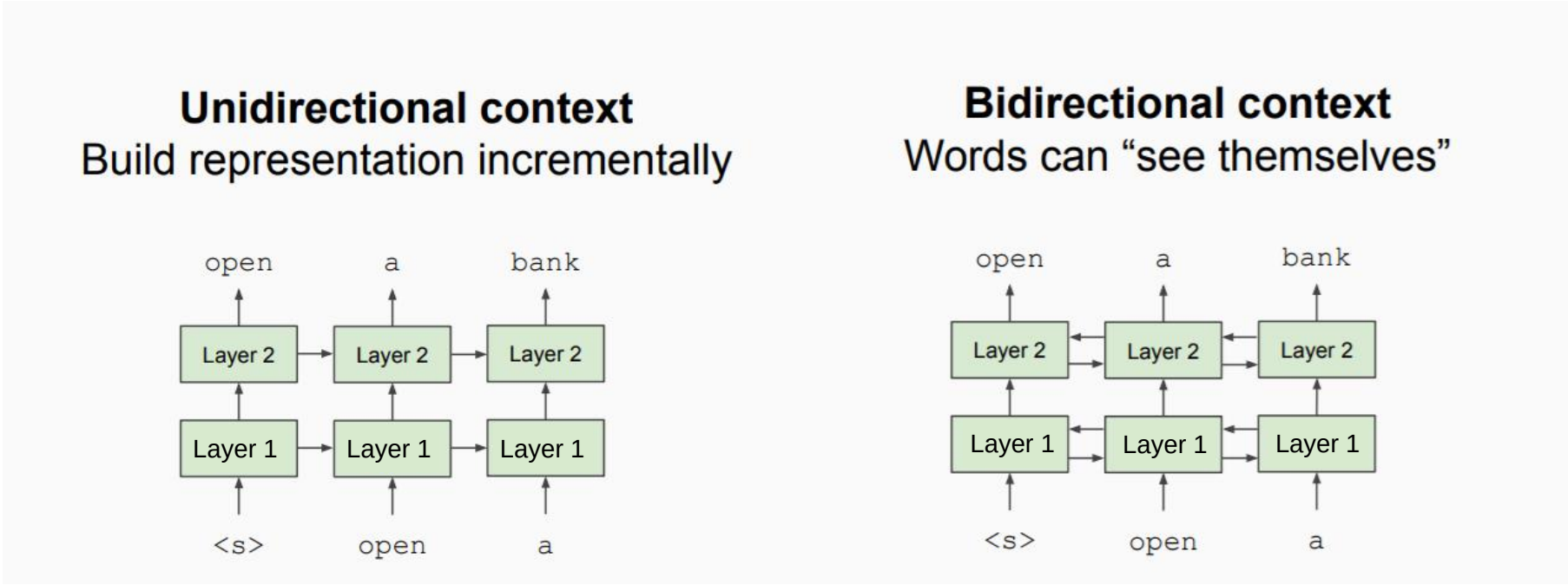


Figure. The transformer decoder model architecture in OpenAI GPT.

BERT

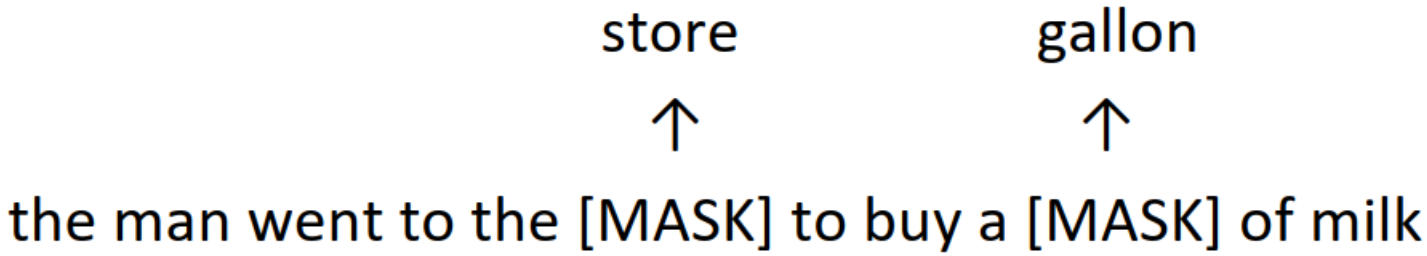
- 动机



单向的语言模型如GPT，无法使用完整的上下文。双向的语言模型在训练时会遇到标签泄露的问题

BERT

- 解决方案：预测目标一



构造完型填空形式的预训练任务：根据上下文预测mask的词

BERT

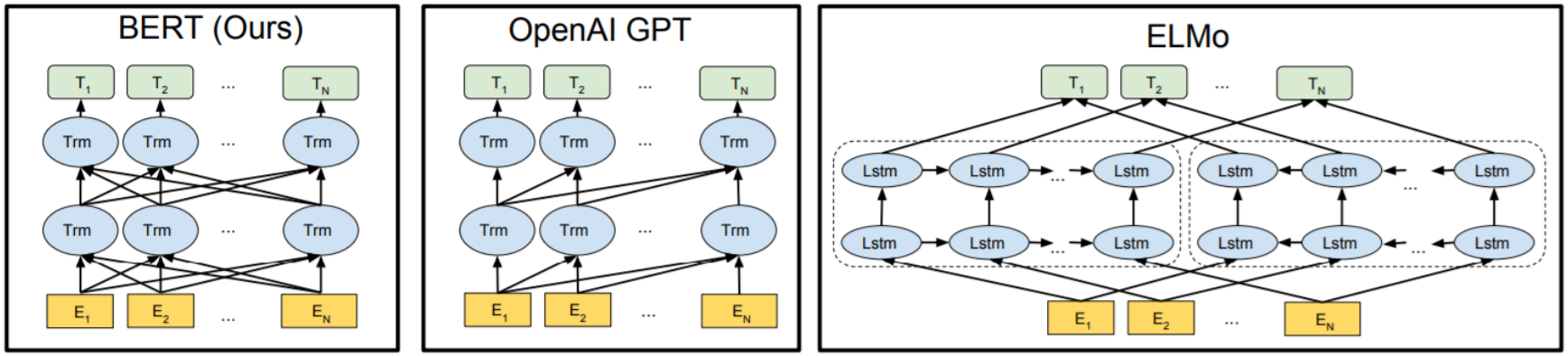
- 解决方案：预测目标二

Sentence A = The man went to the store.
Sentence B = He bought a gallon of milk.
Label = IsNextSentence

Sentence A = The man went to the store.
Sentence B = Penguins are flightless.
Label = NotNextSentence

为了学习句子间的关系，判断B句子是否为A句子的下一句

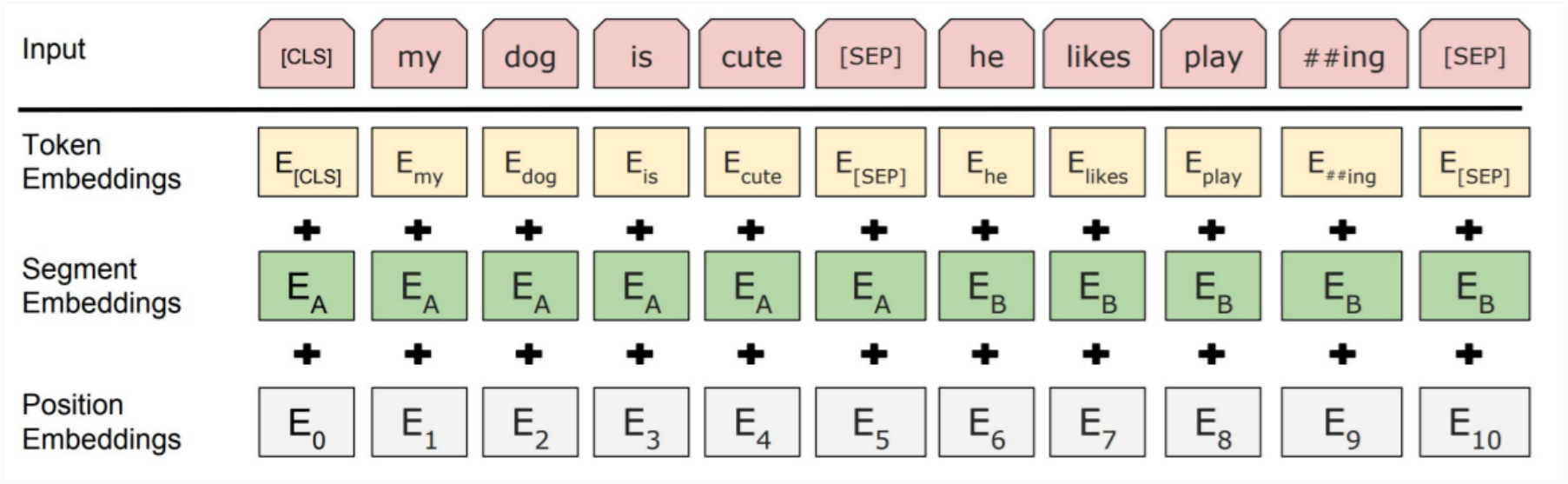
BERT



GPT只能使用单向的信息，ELMo将两个单向LSTM拼接起来，不是真正的双向，只有BERT能够完全使用双向的信息。

BERT

- 输入



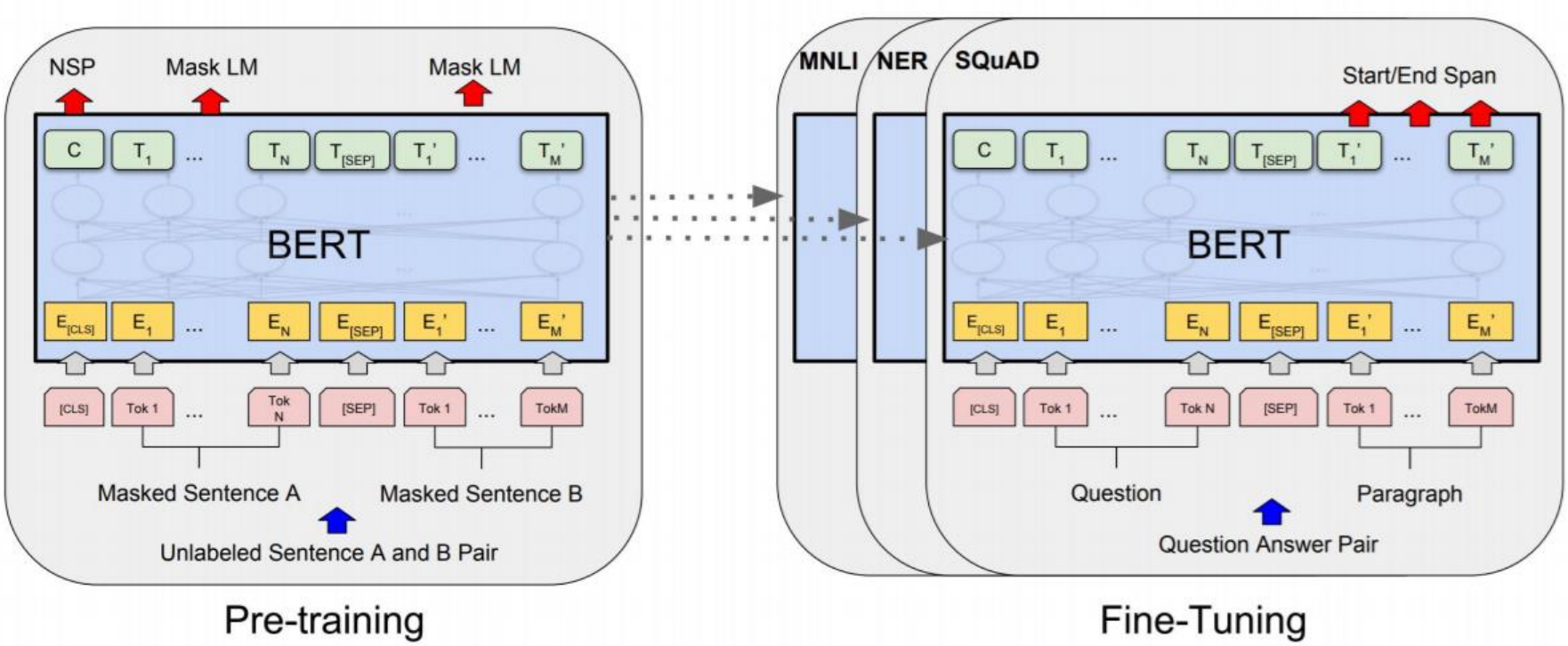
Token Embedding为token的词嵌入。

Segment Embedding为句子类型编码的embedding，为了在下一句子预测任务中分辨A句和B句。

Position Embedding为位置编码的embedding。

BERT

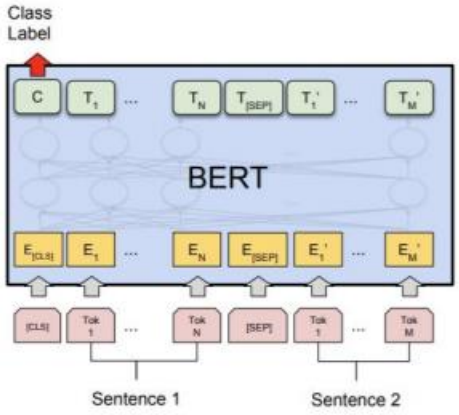
- 预训练——微调


NSP
Mask LM
Mask LM
C
T1
TN
T[SEP]
T1'
TM'
BERT
E[CLS]
E1
EN
E[SEP]
E1'
EM'
[CLS]
Tok 1
Tok N
[SEP]
Tok 1
TokM
Masked Sentence A
Masked Sentence B
Unlabeled Sentence A and B Pair
Pre-training
MNLI
NER
SQuAD
Start/End Span
Tok N
Question
Paragraph
Question Answer Pair
Fine-Tuning

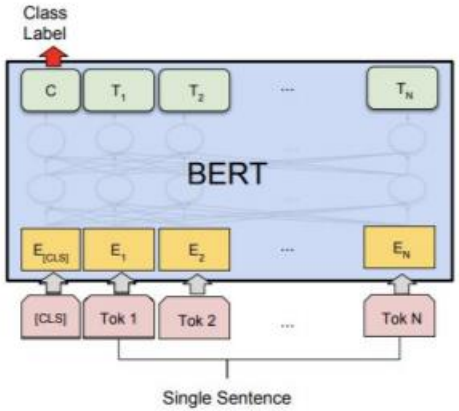
BERT

- 微调下游任务

句子配对任务：使用cls对应隐状态



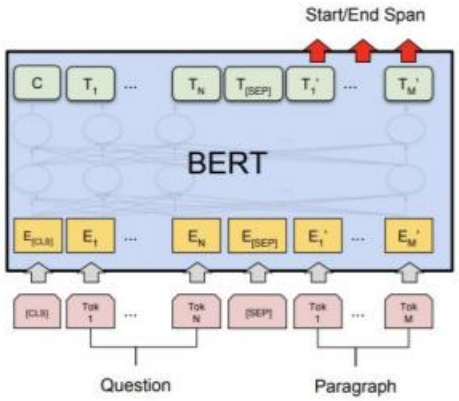
(a) Sentence Pair Classification Tasks: MNLI, QQP, QNLI, STS-B, MRPC, RTE, SWAG



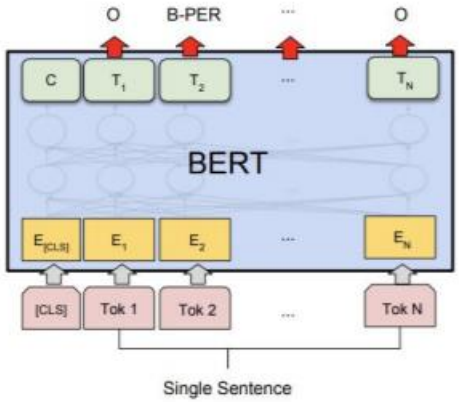
(b) Single Sentence Classification Tasks: SST-2, CoLA

单句分类任务：使用cls对应隐状态

问答任务：将问题和上下文拼接，在上下文中预测答案span的首尾



(c) Question Answering Tasks: SQuAD v1.1



(d) Single Sentence Tagging Tasks: CoNLL-2003 NER

NER任务：使用最后一层的表示预测每个token的类别

BERT

- 性能

System	Dev		Test	
	EM	F1	EM	F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

SQuAD 1.1

Thank you!

权小军 中山大学数据科学与计算机学院