



NLP预训练模型介绍 专题

导师： 向右

比赛相关

成绩

排名	参与者	组织	score	f1	p	r
1	 kaihui1095279440978	楷汇	0.7654	0.7654	0.7189	0.8183
2 ^{↑5}	 蚁族数据	T	0.7640	0.7640	0.7247	0.8079

比赛相关

干货

目前baseline，一折（非全部数据），线上 $F1=0.7483$ ， $P=0.7214$ ， $R=0.7772$ 。全数据 $F1=0.754$ ，线上0.764为交叉验证+参数调优结果（暂不提供，大家可跟随我思路去尝试，后续提供），整体代码提供如下设计：

1. 简单定义即可修改网络结构：

支持采用原生bert最后一层 或 最后多层进行融合，也可自行设计；
支持修改bert+不同网络结构（BILSTM、CNN）进行encoding，也可自行尝试新的结构。

2. 严格按照构建验证集方式，记录实验结果：

支持模型训练过程中保存每个epoch下验证集对应的准确率、召回率、F1值，用于挑选最优模型
支持设置交叉验证，同步记录实验结果。

3. 训练样本目标构造方式上，采用IOBS方式，如想修改设计思路，可自行修改，其他代码复用。

4. 如果没有其他设计思路，可以利用该整合版本代码跑不同的实验结果，进行模型融合。

比赛相关

干货

后续上分点:

1. 调参, lr | batch_size | dropout | bert最后一层, 还是最后多层 | bert+cnn? 抑或 bert+rnn?
2. 提高recall, 从目前线上来看, 召回偏低, 对于解码部分比较严格, 会丢失一部分预测结果, 可想办法尽可能控制准确率提高召回。如果没有好的思路, 可采用模型融合的方式进行召回补充。
3. 构造训练样本方式上, 由于训练样本很多偏长 (大于512), 可以尝试CNN卷积划窗的形式。以某个特殊符号进行切分, 设定窗口大小。
4. 模型融合 (很重要的上分点) | PS: 多实验, 多记录过程, 实时保存最好的模型文件, 最后进行模型融合。
5. 半监督迁移学习 【网上找公开的医学相关数据, 最好数据分布差异小】



比赛相关

干货





比赛相关

后面课为何要听?

- 比赛固然重要，但是基础知识，他是个好东西！！！！
- 告诉你如何做 半监督迁移学习？
- 作为面试官，遇到过很多比赛选手，理论知识匮乏，无法解释清楚某些步骤操作原理以及算法背后原理.....

很重要 很重要 很重要！！！！

目录

1/ 迁移学习

2/ Transformer

3/ 预训练模型

4/ 作业

5/ 互动时间

1、迁移学习

transfer learning



迁移学习

定义

通过减少源域到目标域的分布差异，进行知识迁移，从而实现数据标注工作。



迁移学习

定义的理解

传统深度学习，一般来说训练集与测试集同分布：
$$\min \frac{1}{n} \sum_{i=1}^n L(x_i, y_i, \theta)$$

迁移学习里两个重要的概念

域 (Domain)

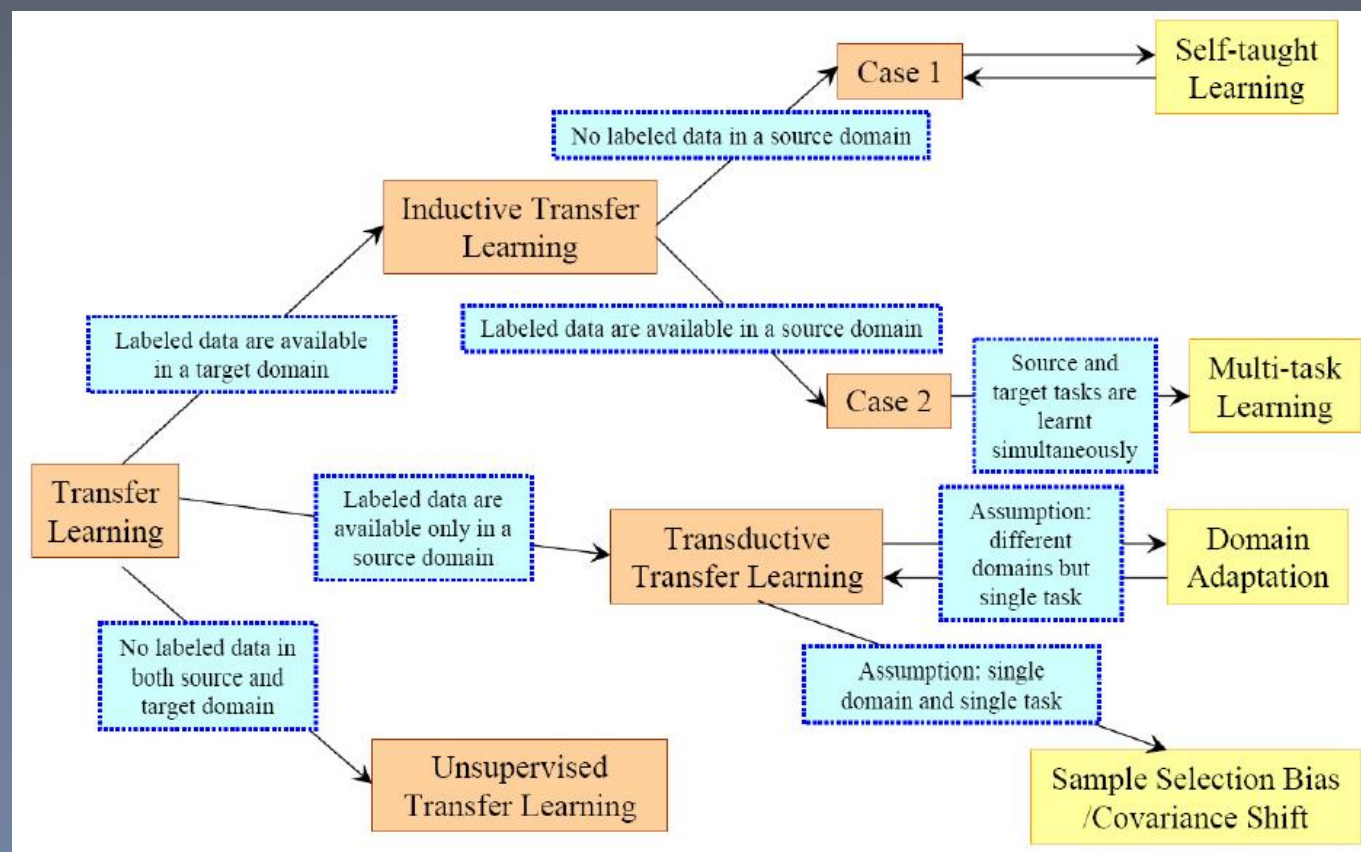
域 可以理解为某个时刻的某个特定领域，比如书本评论和电视剧评论可以看作是两个不同的domain。域本身分为source domain和target domian。

任务 (Task)

任务 就是要做的事情，比如情感分析和实体识别就是两个不同的task。对于source domain和target domian的任务，也不一定一致。

迁移学习

综述





迁移学习

综述

归纳式迁移学习 (*Inductive transfer learning*) :

目标任务不同但是相关, 无论源域和目标域的数据域是相同, 还是不同。

直推式迁移学习 (*Transductive transfer learning*) :

目标任务相同, 但是在目标数据域中没有 (或者含有少量) 可获得的、带标记的数据, 然而在源数据域中有许多可获得的带标记的数据。

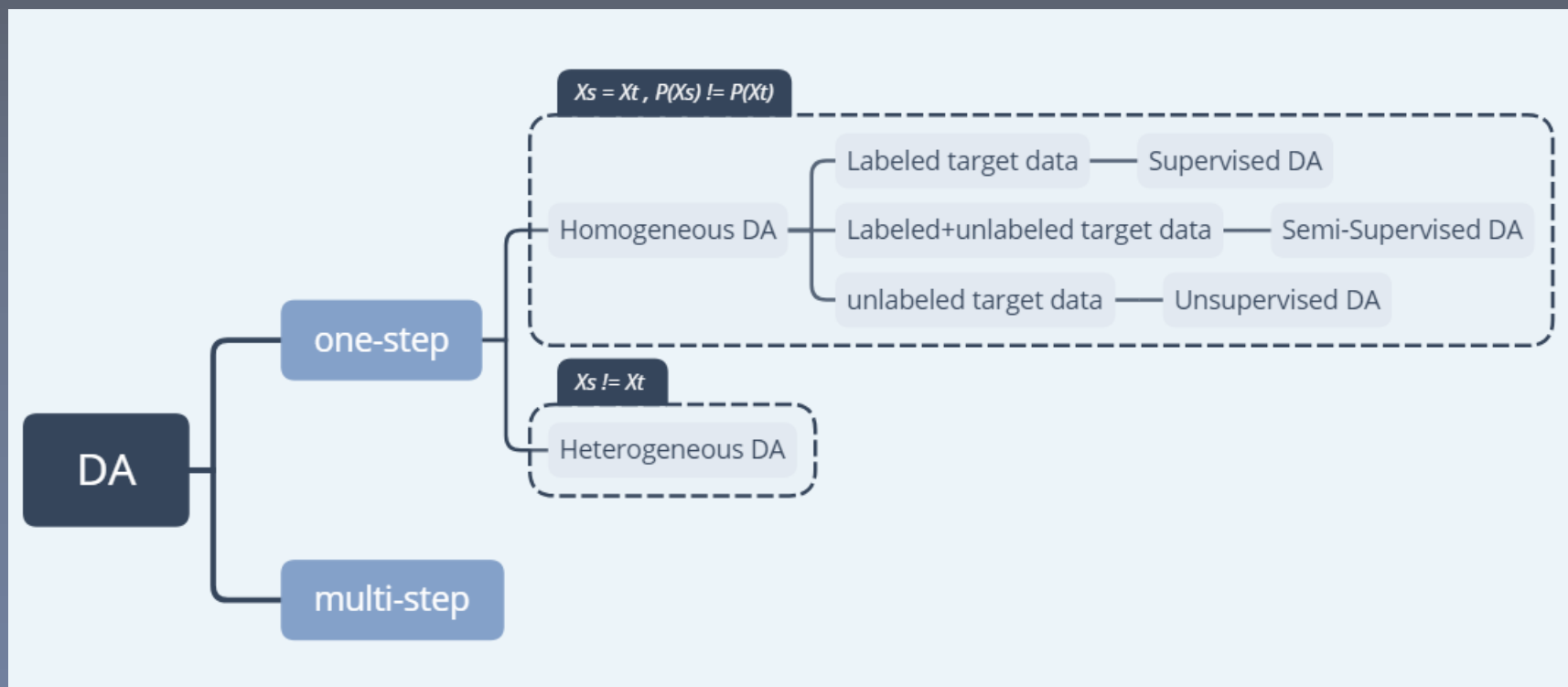
无监督迁移学习 (*Unsupervised Transfer Learning*) :

在源域和目标域中都没有带标签的数据, 关注与目标任务上的聚类、降维和密度估计。



迁移学习

Domain adaptation



Domain adaptation

Feature adaptation

$$\min \frac{1}{n} \sum_{i=1}^n L(\theta(x_i), y_i, \theta)$$

Find a “good” feature representation that reduces difference between the source and the target domains and the error of classification and regression models。

Supervised DA

fine-tune

为什么 fine-tune 有效？

用大型数据集训预练，已经具备了提取浅层基础特征和深层抽象特征的能力。

可以有效避免：

1. 从头开始训练，需要大量数据，浪费计算时间和计算资源；
2. 模型不收敛，参数不够优化，模型泛化能力差。

Semi-Supervised DA

实现方式

$$\begin{aligned} L &= \min \left(\frac{1}{n} \sum_{i=1}^n W_1 * L(\theta(x_i), y_i, \theta) + \frac{1}{m} \sum_{i=1}^m W_2 * L(\theta(x_i), y_i, \theta) \right) \\ &= \min \frac{1}{n+m} \sum_{i=1}^{n+m} W_i * L(\theta(x_i), y_i, \theta) \end{aligned}$$

其中 W_i ，是每个样本的权重。有两种设计思路：

1. 固定 W_1 与 W_2 值，经验值，真实标签个数为 n ，未标注样本个数为 m ， W_1 对应真实样本权重， W_2 对应未标注样本权重。

经验值： m/n 在 $[5*W_1/W_2, 10*W_1/W_2]$

2. 探索一下线上与线下的标签分布，根据标签分布进行 W 权重的调整（扰动）。

Semi-Supervised DA

相关论文推荐

Deep Domain Confusion: Maximizing for Domain Invariance

2、Transformer

总览

Transformer框架

Transformer

- Left: Encoder
- Right: Decoder

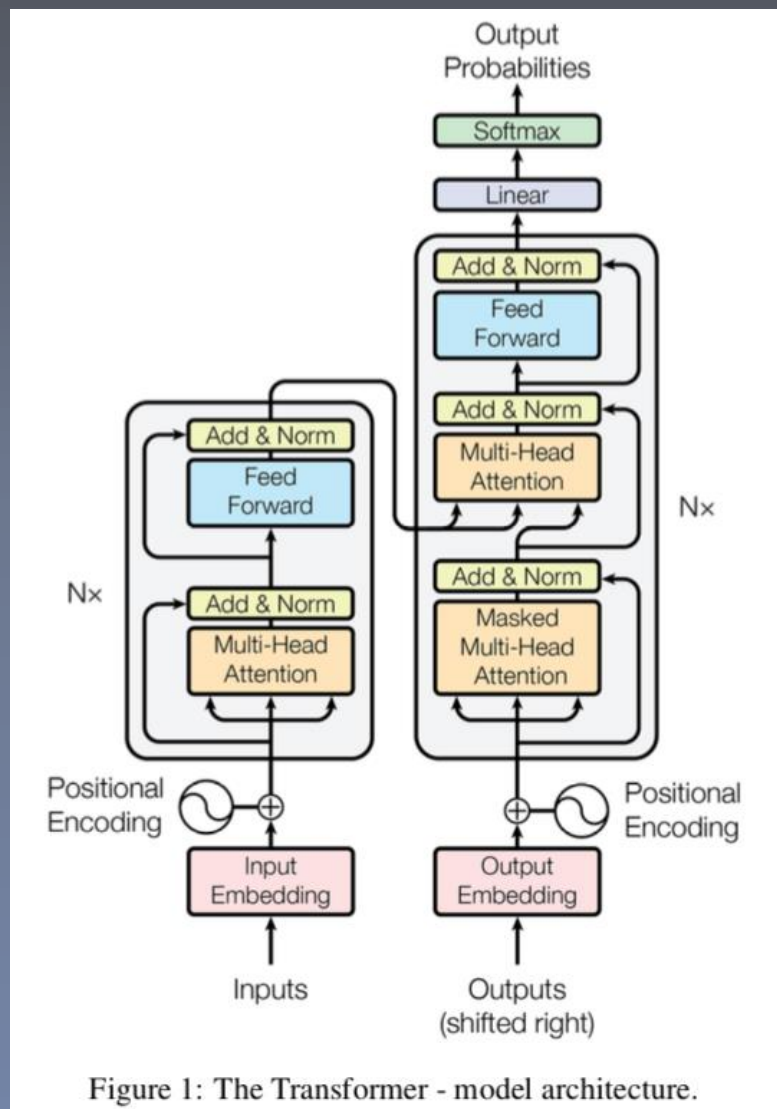


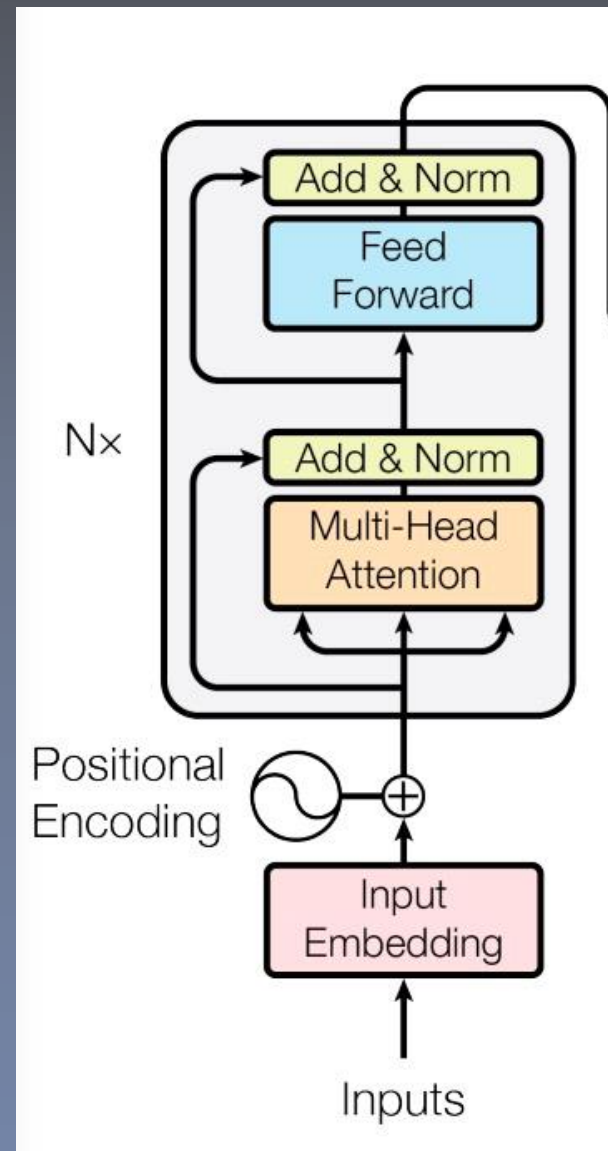
Figure 1: The Transformer - model architecture.



编码器

Encoder: 6 identical layers

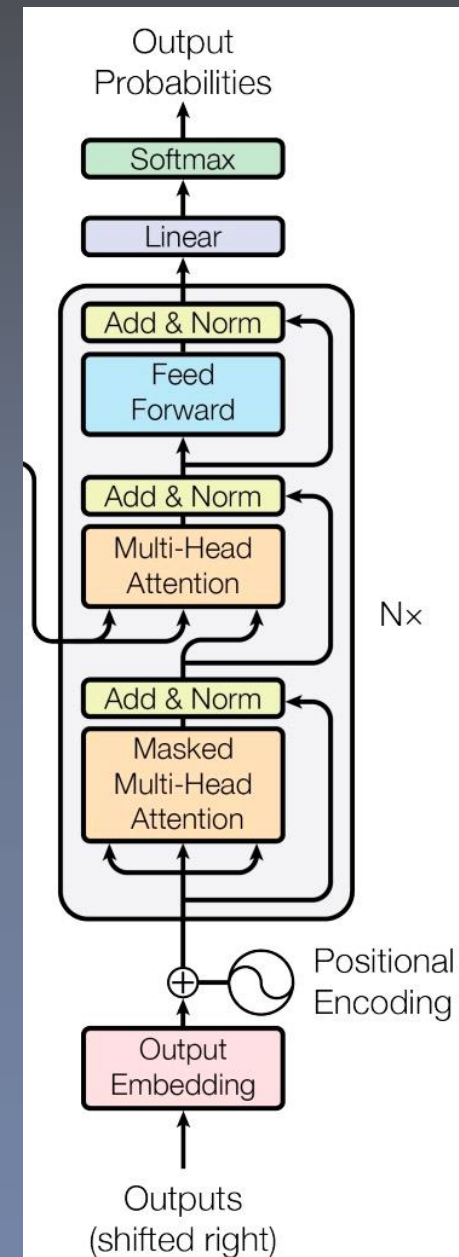
- **Layer:**
 1. multi-head self-attention
 2. fully connected feed-forward network
- **Residual Connection:**
 $x + \text{Sublayer}(x)$
- **LayerNorm($x + \text{Sublayer}(x)$)**
- $d_{\text{model}} = 512$



解码器

Decoder: 6 identical layers

- **Layer:**
 1. **masked multi-head self-attention**
 2. **multi-head attention**
 3. **fully connected feed-forward network**
- **Residual Connection**
 $x + \text{Sublayer}(x)$
- **LayerNorm($x + \text{Sublayer}(x)$)**
- $d_{\text{model}} = 512$



Self-attention

Self-attention

注意力权重函数

Attention Score Functions

- q is the query and k is the key
- Multi-layer Perceptron(Bahdanau et al. 2015)

$$a(q, k) = w_2^T \tanh(W_1[q; k])$$

- Bilinear(Luong et al. 2015)

$$a(q, k) = q^T W k$$

注意力权重函数

Attention Score Functions

- q is the query and k is the key
- Dot Product(Luong et al. 2015)

$$a(q, k) = q^T k$$

- No parameters! But requires sizes to be the same.
- Scaled Dot Product(Vaswani et al. 2017)
 - Problem: scale of dot product increases as dimensions get larger
 - Fix: scale by size of the vector

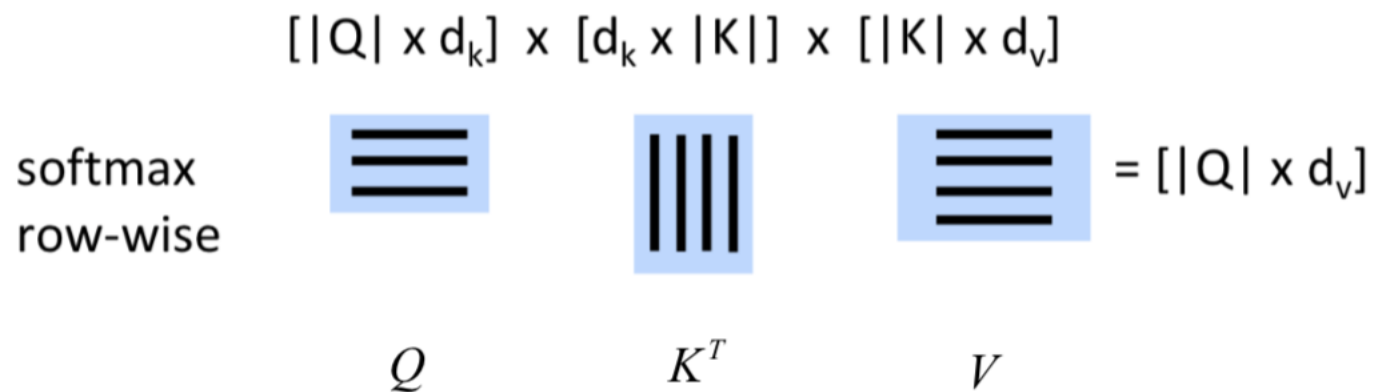
$$a(q, k) = \frac{q^T k}{\sqrt{|k|}}$$



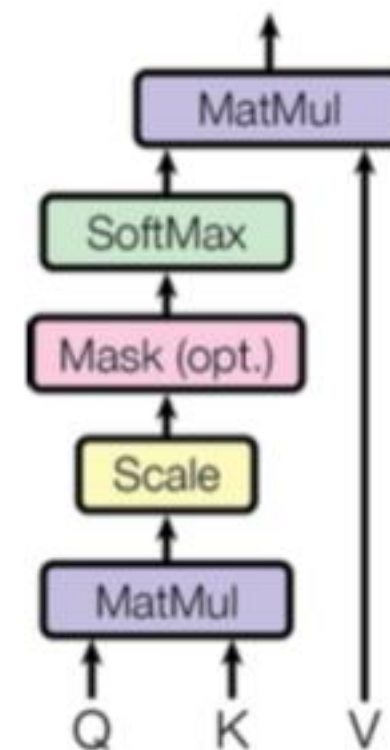
放缩后的点乘注意力

Scaled Dot-product Attention

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

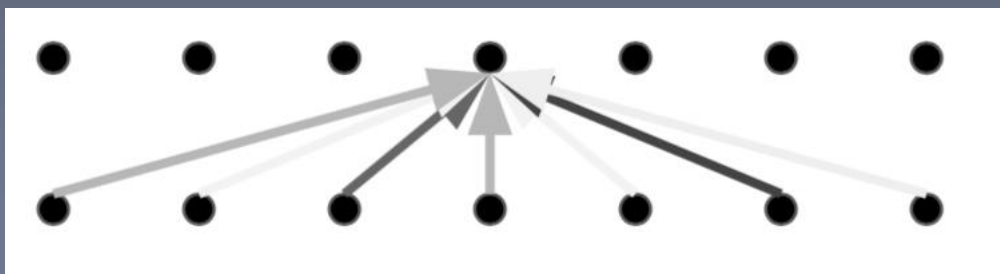


Scaled Dot-Product Attention



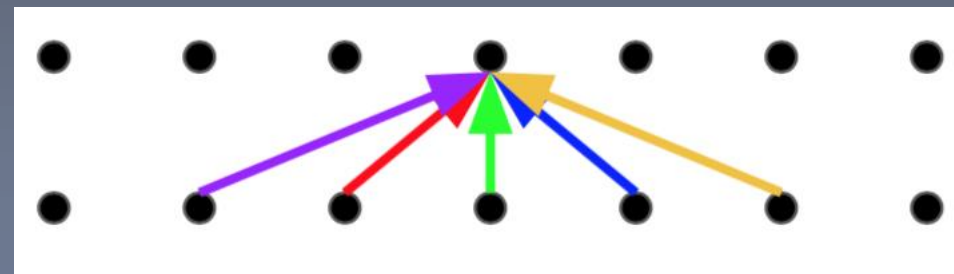
自注意力机制与卷积网络的对比

comparison between Self-Attention and Convolution



Self-Attention

variable-sized perceptive field

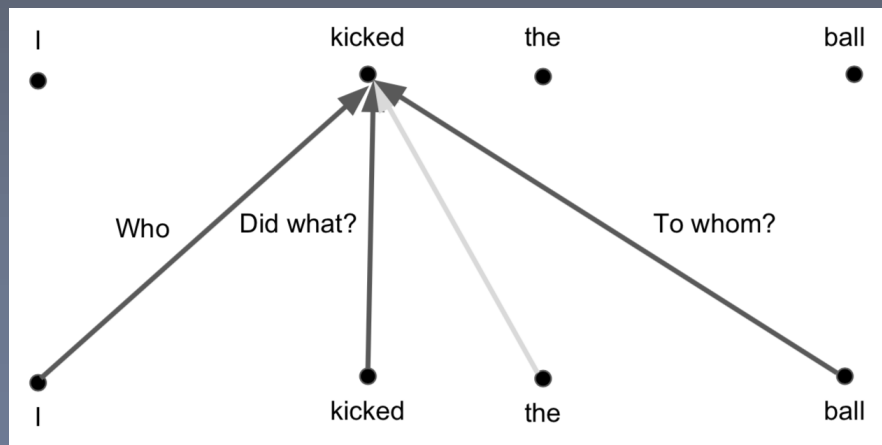


Convolution

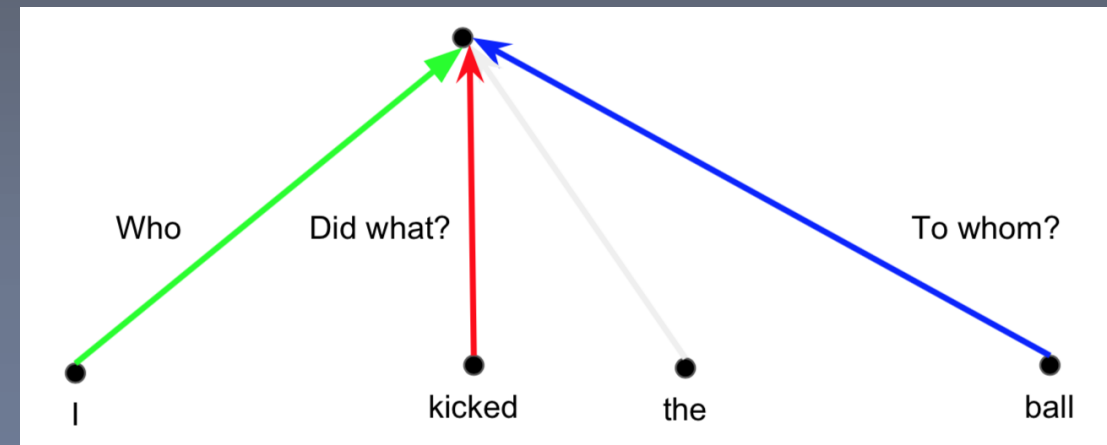
local
dependency

自注意力机制与卷积网络的对比

comparison between Self-Attention and Convolution



Self-Attention

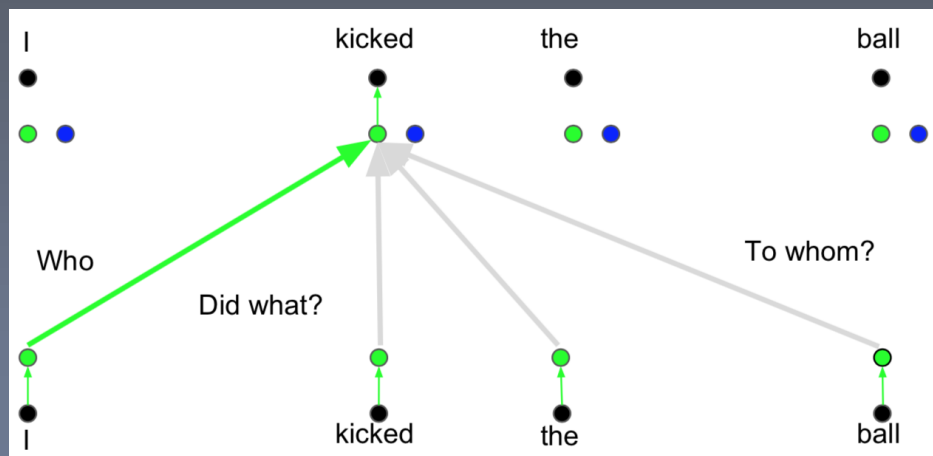


Convolution

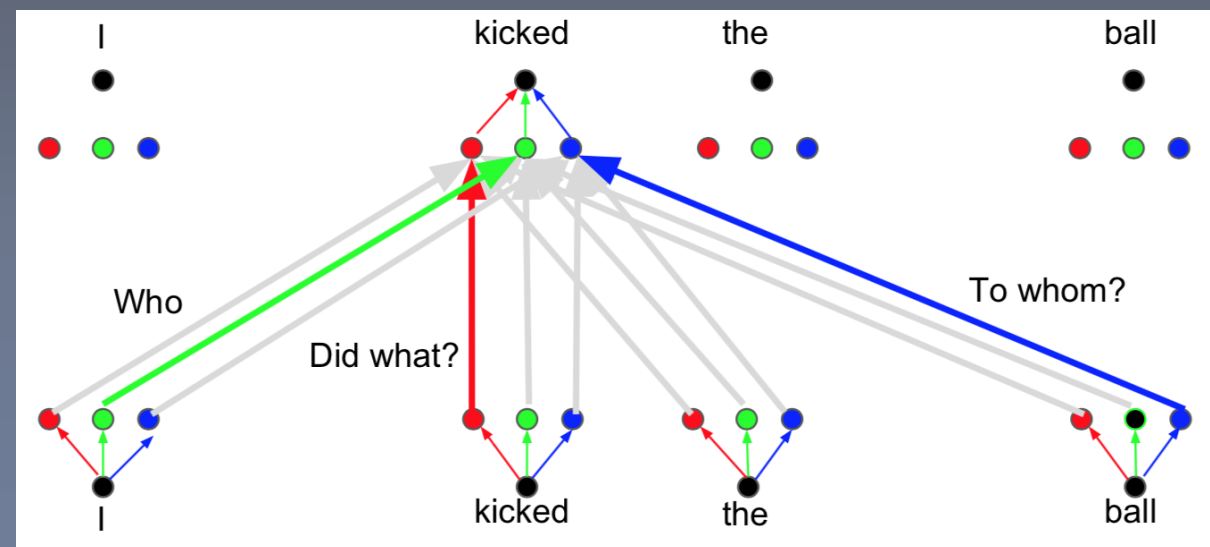
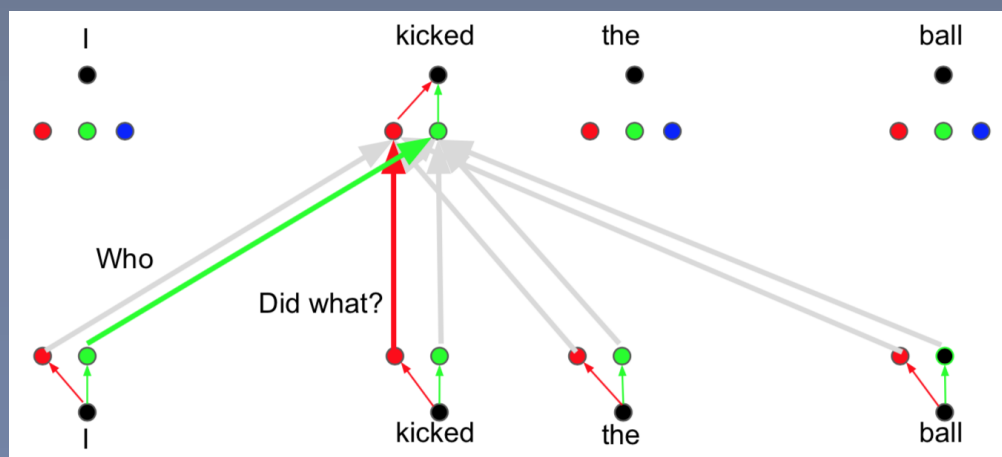
Question: How to convey multi-channel information?

并行的注意力头

parallel attention heads



a



3、预训练模型

Pre-Training



预训练模型

种类

预训练模型:

BERT、ALBERT、XLNET、BERT-WWM、Roberta。

都是基于 transformer 结构的预训练语言模型，包括了 Bert 及其后继者 Bert-WWM、Roberta、XLNet、Albert 等，统称为 BERT 家族。它们不仅在结构上很相似，而且在使用方法上更是高度一致。

预训练模型

简介

Bert

Bert 是一种基于 Transformer Encoder来构建的预训练语言模型，它是通过 Masked Language Model(MLM) 和 Next Sentence Prediction(NSP) 两个任务在大规模语料上训练得到的。

开源的 Bert 模型分为 base 和 large，它们的差异在模型大小上。大模型有更大的参数量，性能也有会几个百分点的提升，当然需要消耗更多的算力，BERT 家族其他模型也类似。

bert

预训练任务



深度之眼
deepshare.net



重点

- MLM: masked language model

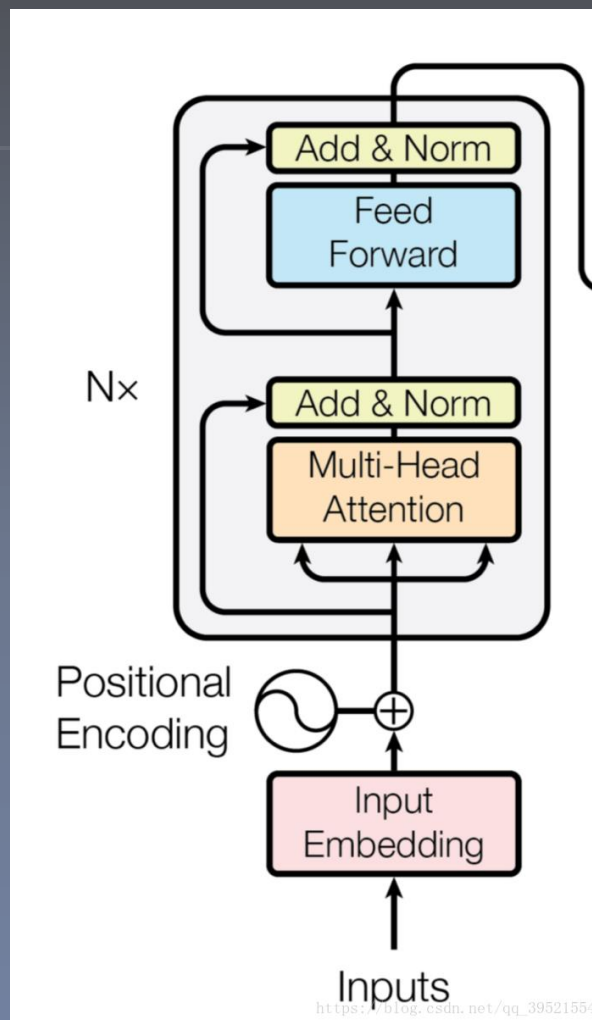
在每一轮迭代中随机选择15%的词隐藏，目标是通过它们的上下文来预测这些单词。操作是取这些词对应的最后一个隐单元向量后接一个softmax来预测这个词。

- Next Sentence Prediction

预测第二个句子是否可以自然的接在第一个句子后面，是个二分类问题，用于理解句子间的关系。

bert

结构



$$O = (W^v * I) * softmax((W^k * I)^T * (W^q * I))$$



预训练模型

输入截断方式

常用的截断的策略有三种：

- pre-truncate
- post-truncate
- middle-truncate (head + tail)

预训练模型

简介

Bert-WWM

模型结构与 Bert 完全一样，只是在 MLM 训练任务上做了一个小的改进。Bert 在做 MLM 采用的是 token 级别的 mask，而 Bert-WWM 则采用了词级别的mask，更加合理一些。

预训练模型

简介

Roberta

Bert 的优化版，模型结构与 Bert 完全一样，只是在数据量和训练方法上做了改进。简单说就是更大的数据量，更好的训练方式，训练得更久一些。

- 相比原生 Bert 的16G训练数据，RoBerta 训练数据量达到了161G；
- 去除了 NSP 任务，研究表明 NSP 任务太过简单，不仅不能提升反倒有损模型性能；
- MLM 换成 Dynamic Masking LM；
- 更大的 batch size 以及其他超参数的调优。

预训练模型

简介

XLNet

XLNet 对 Bert 做了较大的改动，二者在模型结构和训练方式上都有不小的差异。

- Bert 的 MLM 在预训练时有 MASK 标签，但在使用时却没有，导致训练和使用时出现不一致；并且 MLM 不属于 Autoregressive LM，不能做生成类任务。XLNet 采用 PML(Permutation Language Model) 避免了 MASK 标签的使用，且属于 Autoregressive LM，可以做生成任务。
- Bert 使用的 Transformer 结构对文本的长度有限制，为更好地处理长文本，XLNet 采用升级版的 Transformer-XL。



预训练模型

简介

Albert

Albert (Bert 瘦身版本)，希望用更简单的模型，更少数据，得到更好的结果。它主要从以下两个方面减少模型的参数量：

- 对 Vocabulary Embedding 进行矩阵分解，将原来的矩阵 $V \times E$ 分解成两个矩阵 $V \times H$ 和 $H \times E$ ($H \ll E$)。
- 跨层参数共享，可以避免参数量随着网络深度的增加而增加。



预训练模型

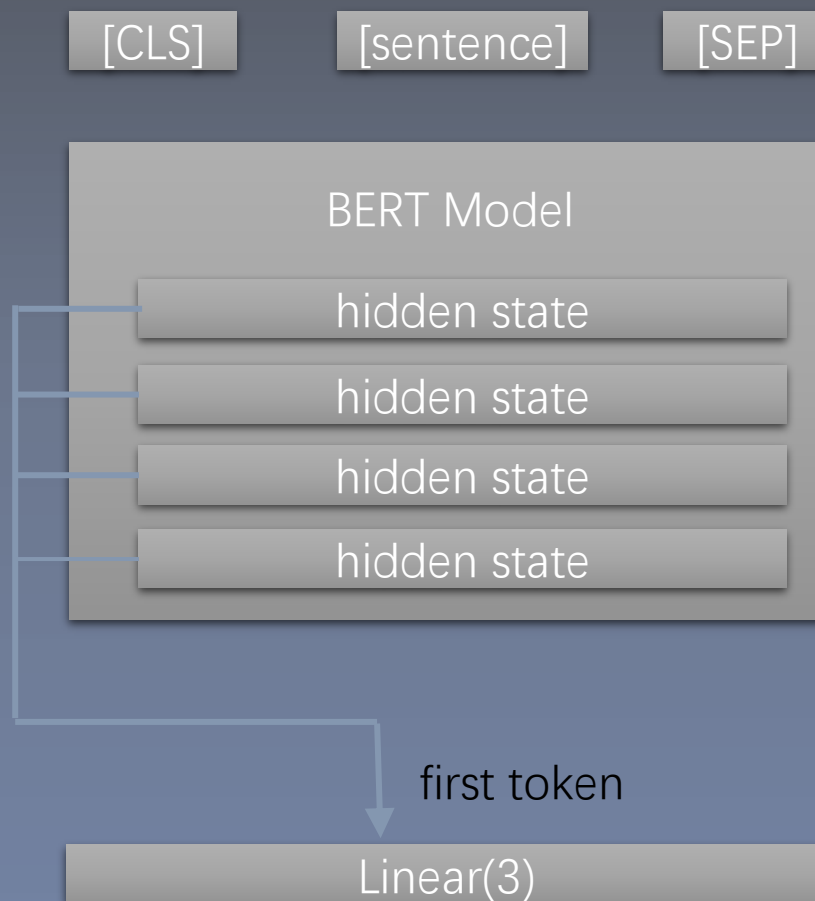
简介

这些模型的性能在不同的数据集上有差异，需要试了才知道哪个表现更好，但总体而言 XLNet、Roberta、Bert-WWM 会比 Bert 效果略好，large 会比 base 略好。ALbert也有多个版本，large版本训练时间其实也没有降低，tiny版本会好很多。

更多情况下，它们会被一起使用，最后做 ensemble。

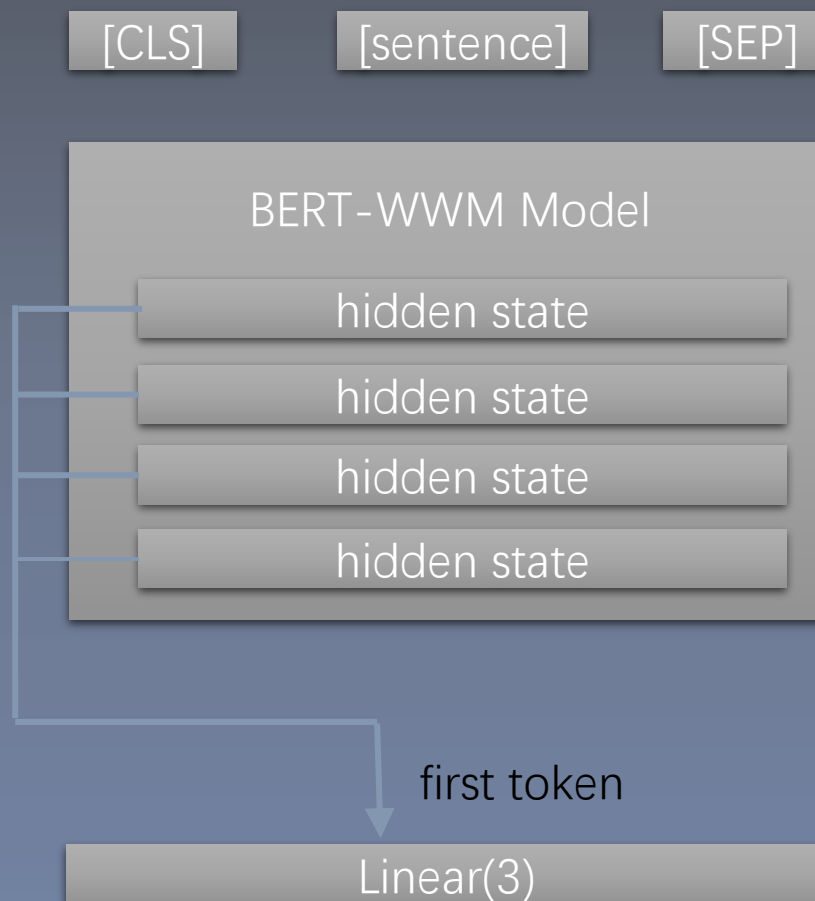
预训练模型

fine-tune



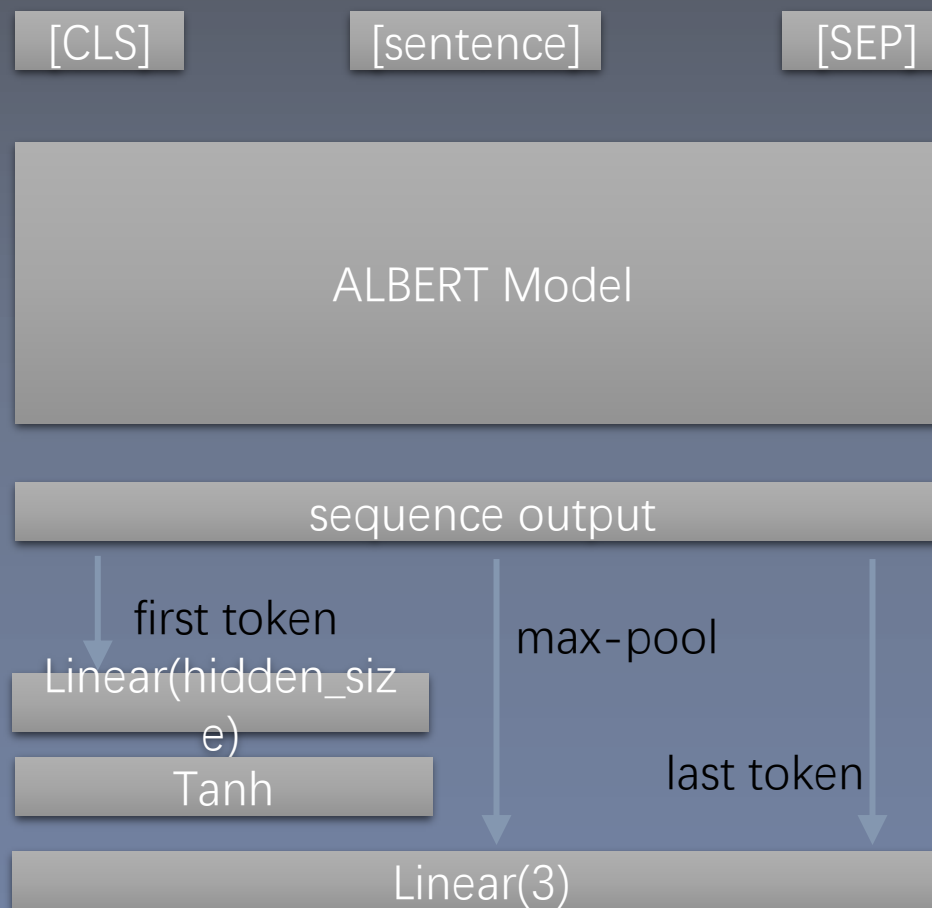
预训练模型

fine-tune



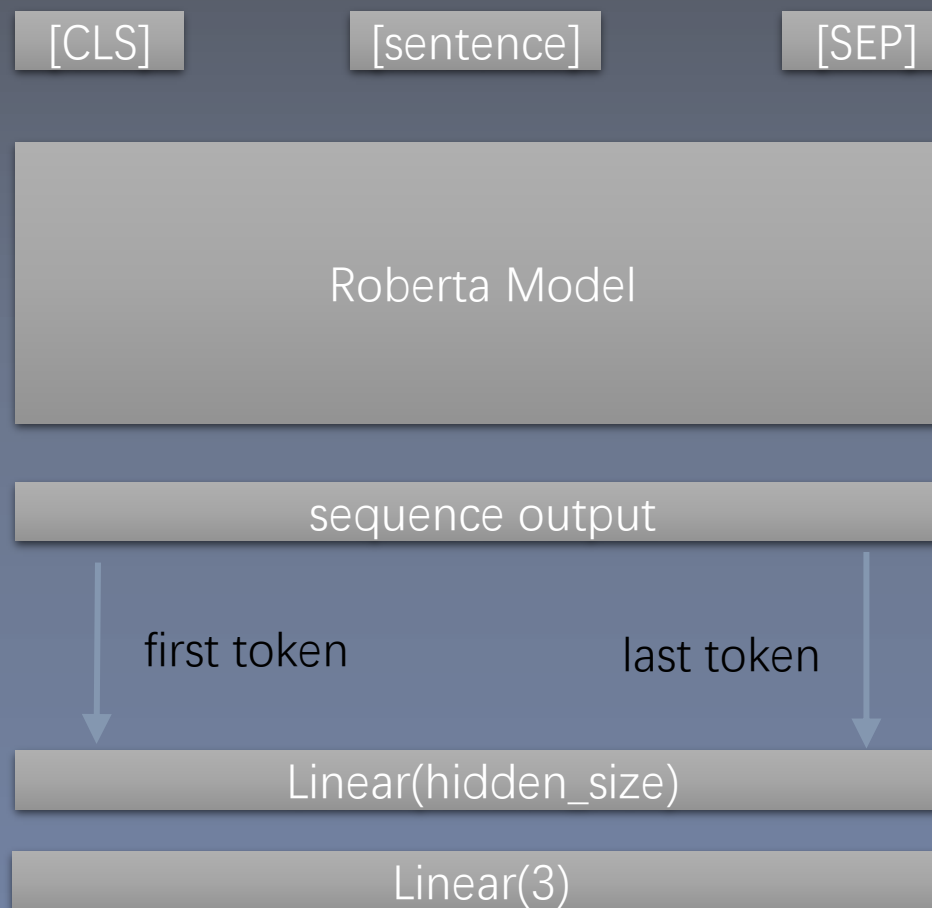
预训练模型

fine-tune



预训练模型

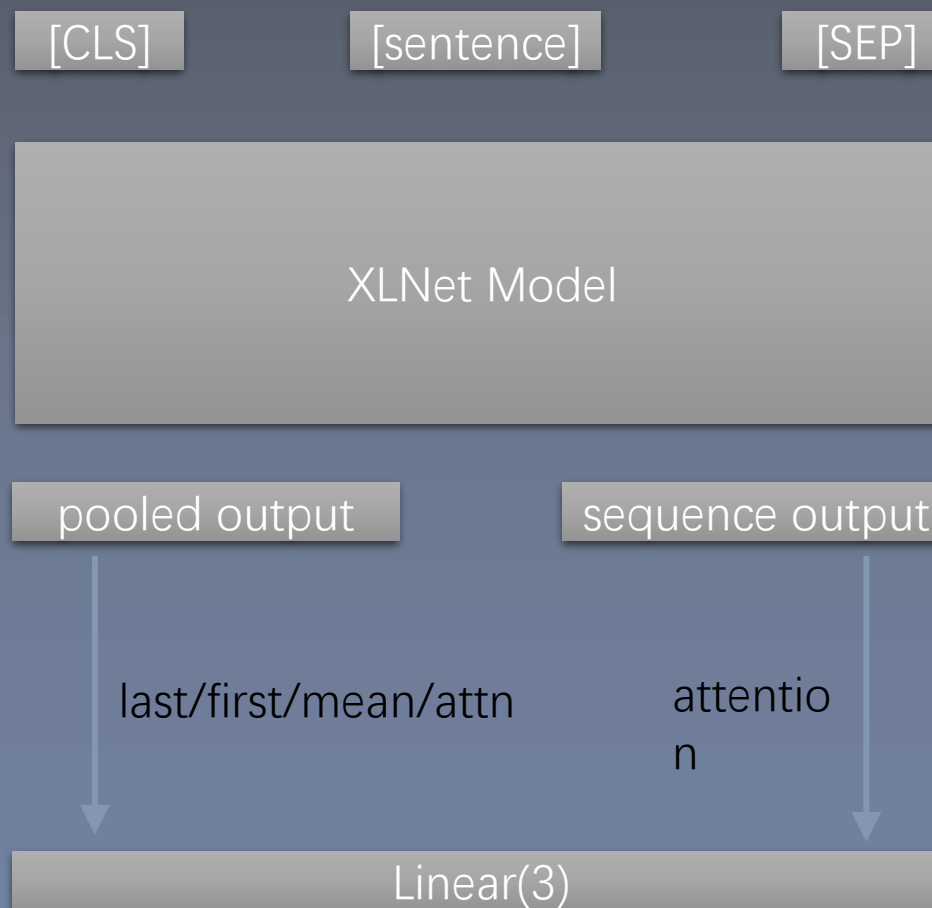
fine-tune





预训练模型

fine-tune



4、作业

Introduction of Data

作业



1. position encoding, 为何需要用, 如何设计?
2. Batch Normalization与Layer Normalization区别? bert中为什么用后者?
3. 熟悉GELU激活函数, 与RELU差异。
4. 实际操作, Semi-Supervised DA方法;
5. 对比实施模型融合的相关方法。

深度之眼
deepshare.net

疫情期间网民
情绪识别AI大赛
教练指导带打

立即扫码即可报名

N T
R V
E W



—— 结 语 ——

感谢大家参加本次的比赛直播

课下请一定要

亲自动手操作一次！



5、互动时间



deepshare.net

深度之眼

联系我们：

电话：18001992849

邮箱：service@deepshare.net

QQ：2677693114



公众号



客服微信

