

一、名词解释

玻尔兹曼机

玻尔兹曼机 (Boltzmann Machine) 则可以通过让每个神经元按照一定的概率分布发生状态变化，来避免陷入局部最优解。

目标检测

检测图像中物体的位置进而识别物体。

随机梯度下降

随机梯度下降算法每次从训练集中随机选择一个样本来进行学习

扩张（空洞）卷积

针对图像语义分割问题中下采样会降低图像分辨率、丢失信息而提出的一种卷积形式。

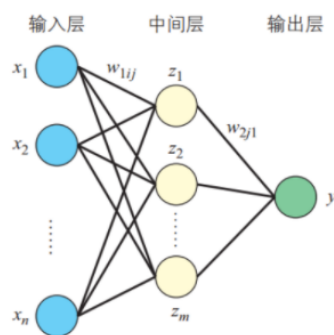
Triplet loss

输入是三元组，包括锚 (Anchor) 示例、正 (Positive) 示例、负 (Negative) 示例，通过优化锚示例与正示例的距离小于锚示例与负示例的距离，实现样本之间的相似性计算；

二. 简答题

1. 请简述反向传播算法的思想，并用图和公式进行说明。

BP 算法就是通过比较实际输出(y) 和期望输出(r) 得到误差信号，把误差信号从输出层逐层向前传播得到各层的误差信号，再通过调整各层的连接权重以减小误差。权重的调整主要使用梯度下降法。



- 首先调整中间层与输出层之间的连接权重，把误差函数 E 对连接权重 w_{2j1} 的求导展开成复合函数求导：

$$\begin{aligned}\frac{\partial E}{\partial w_{2j1}} &= \frac{\partial E}{\partial y} \frac{\partial y}{\partial u_{21}} \frac{\partial u_{21}}{\partial w_{2j1}} \\ &= -(r - y)y(1 - y)z_j\end{aligned}$$

这里， u_{21} 表示中间层到输出层的值， z_j 表示中间层的值。

83

Figure 1 来自老师 PPT

2. 请简述深度可分离卷积的主要思想，并举例说明。

Depthwise Separable Convolution，先做 Depthwise 卷积，再做 Pointwise 卷积，实现空间维（卷积核大小）和通道维（特征图）的分离。

Pointwise 采用的是 1×1 卷积的方式，此步涉及到的参数个数为： $1 \times 1 \times 3 \times 4 = 12$ 。深度可分离卷积总的参数个数： $27 + 12 = 39$ ，相比标准卷积，参数个数大幅降低。

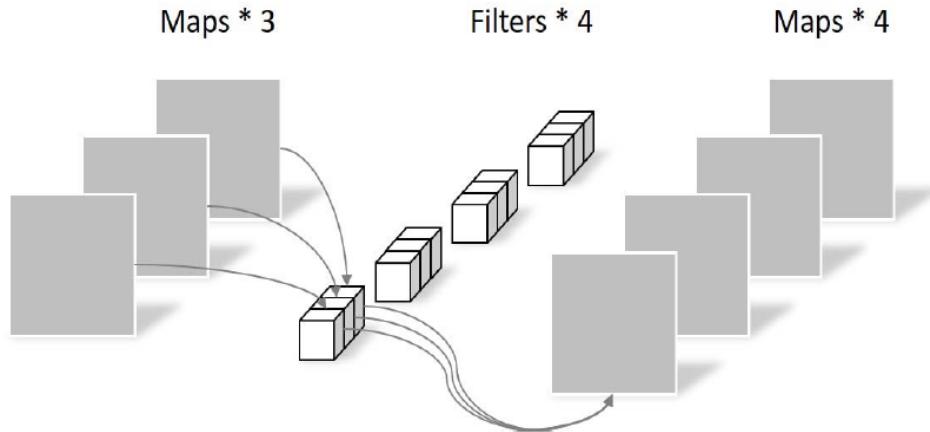


Figure 2 来自老师 PPT

3. 请简述 SSD 网络的主要思想和实现过程，并画图说明。

SSD 沿用了 YOLO 中基于卷积结果直接进行 Bbox 回归和分类的方法，同时又参考了 Faster R-CNN，大量使用 anchor 来提升识别准确度。在参考 Yolo 和 Faster R-CNN 的基础上，提出了多次 Bbox 回归和分类的方法。

实现过程

编码过程，使用 `gt_boxbounding_box` 的归一化后的中心点表示的 offset 作为基准，与模型的输出做 loss。

解码过程：模型的输出 `bounding_bo` 与真实值的 offset。推理阶段，用 offset 与 `bounding_box`，即可得到最终目标框

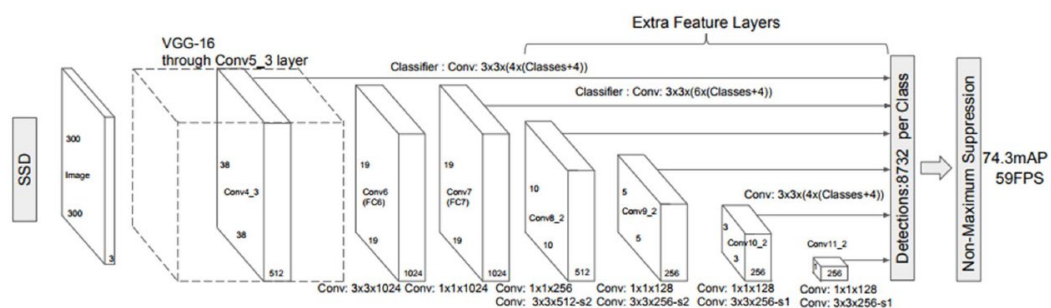


Figure 3 来自老师 PPT

4. 请简述 LSTM 网络的主要思想，并用图和公式进行说明。

长短期记忆（Long short-term memory, LSTM）是一种特殊的 RNN，主要是为了解决长序列训练过程中的梯度消失和梯度爆炸问题。简单来说，就是相比普通的 RNN，LSTM 能够在更长的序列中有更好的表现。

LSTM 依靠贯穿隐含层的细胞状态实现隐含单元之间的信息传递，其中只有少量的线性干预和改变。LSTM 引入了“门”机制对细胞状态信息进行添加或删除，由此实现长程记忆。

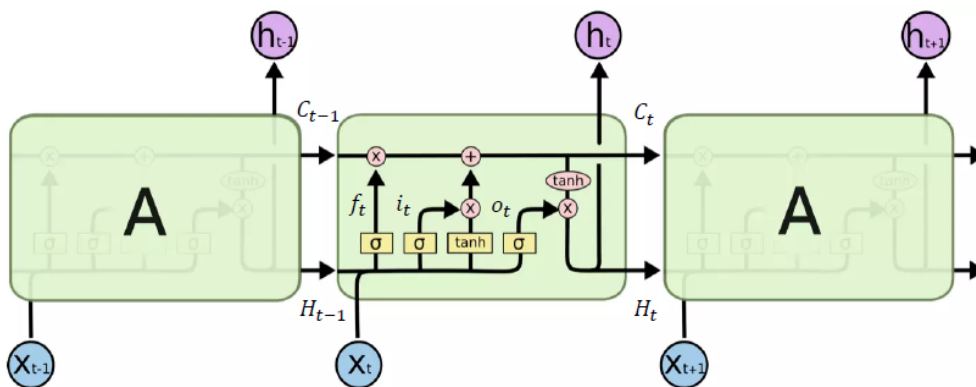


Figure 4 来自老师 PPT

遗忘门: LSTM 通过遗忘门 (forget gate) 实现对细胞状态信息遗忘程度的控制, 输出当前状态的遗忘权重, 取决于 H_{t-1} 和 X_t 。

$$f_t = \sigma(U_f x_t + W_f h_{t-1} + b_f)$$

输入门: LSTM 通过输入门 (input gate) 实现对细胞状态输入接收程度的控制, 输出当前输入信息的接受权重, 取决于 H_{t-1} 和 X_t 。

$$i_t = \sigma(U_i x_t + W_i h_{t-1} + b_i)$$

$$\tilde{C}_t = \tanh(U_C x_t + W_C h_{t-1} + b_C)$$

输出门: LSTM 通过输出门 (output gate) 实现对细胞状态输出认可程度的控制, 输出当前输出信息的认可权重, 取决于 H_{t-1} 和 X_t 。

$$o_t = \sigma(U_o x_t + W_o h_{t-1} + b_o)$$

$$h_t = o_t \cdot \tanh(C_t)$$

5. 请简述注意力机制的主要原理, 并基于 Encoder/Decoder 架构用图和公式进行说明。

神经网络模型注意力机制: 注意力机制的本质是**权重**。

在这个新结构中, 定义每个输出的条件概率为:

$$p(y_i | y_1, \dots, y_{i-1}, \mathbf{x}) = g(y_{i-1}, s_i, c_i)$$

其中 s_i 为解码器 RNN 中的隐层状态:

$$s_i = f(s_{i-1}, y_{i-1}, c_i)$$

这里的上下文向量 c_i 取决于编码器状态序列, 通过使用注意力系数 α_{ij} 对 h_j 加权求得:

$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j.$$

注意力机制的常用公式类型

$$s = f(Q, K) = \begin{cases} Q^T K_i & \text{点积模型} \\ \frac{Q^T K_i}{\sqrt{d}} & \text{缩放点积模型} \\ W_a[Q, K_i] & \text{连接模型} \\ Q^T W_a K_i & \text{双线性模型} \\ V^T_a \tanh(W_a Q + U_a K_i) & \text{加性模型} \end{cases}$$

Figure 5 来自老师 PPT

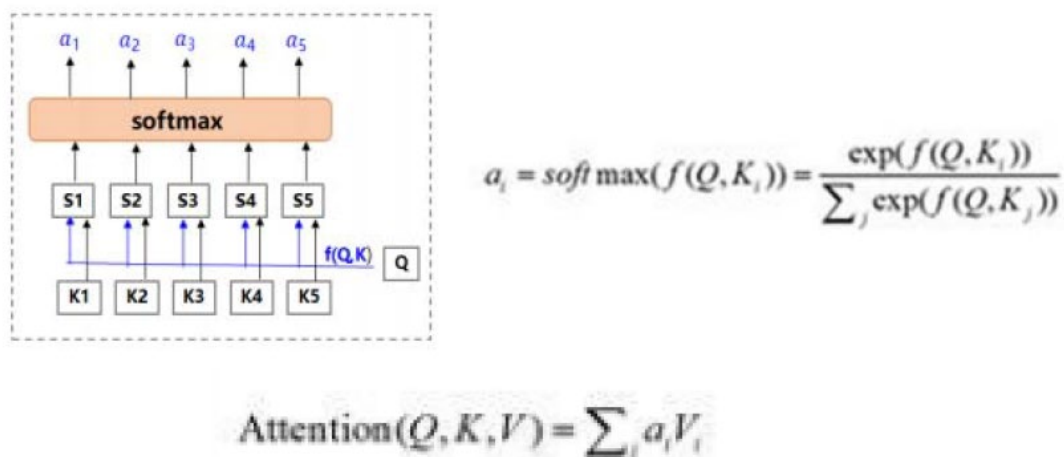


Figure 6 来自老师 PPT

Transformer 的 Encoder 和 Decoder

• Transformer

Encoder

- 输入 $x \in \mathbb{R}^{n \times d_{model}}$
- 输出大小不变
- Positional Encoding
- 6个Block
 - Multi-Head Self-Attention
 - Position-wise Feed Forward
 - Residual connection
 - LayerNorm(x + Sublayer(x))
 - 引入了残差, 尽可能保留原始输入x的信息
- $d_{model}=512$

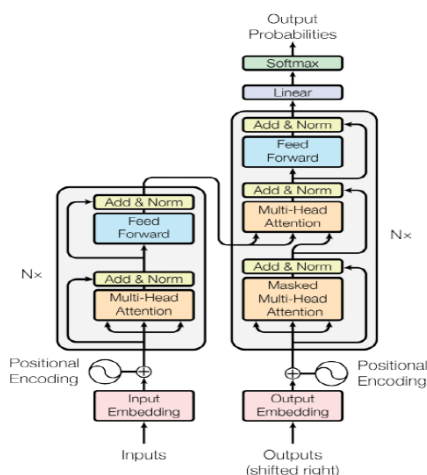


Figure 1: The Transformer - model architecture.

Decoder

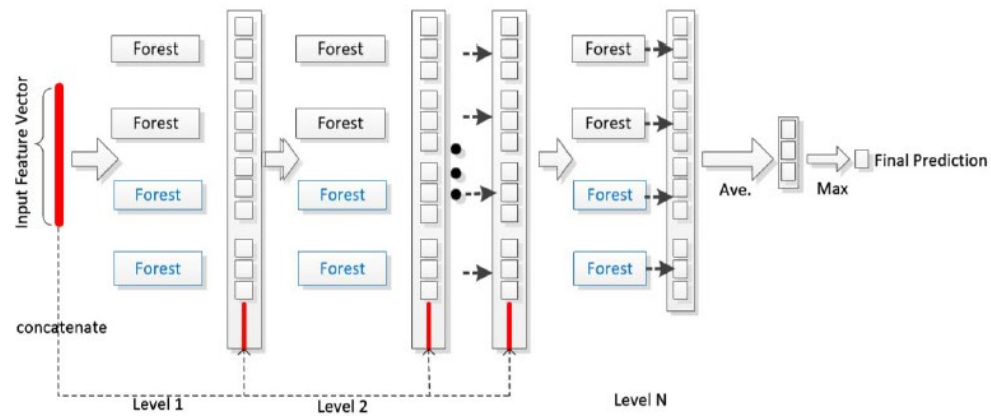
- Positional Encoding
- 6个Block
 - Multi-Head Self Attention (with mask)
 - 采用 0-1mask 消除右侧 单词对当前单词 attention 的影响
 - Multi-Head Self Attention (with encoder)
 - 使用Encoder的输出作为一部分输入
 - Position-wise Feed Forward
 - Residual connection

Figure 7 来自老师 ppt

6. 请简述深度森林的主要原理, 并画图说明。

神经网络可以堆叠为深度神经网络, 并且取得了显著的效果。那我们可以考虑, 是不是可以将其他的学习模型堆叠起来, 以获取更好的表示性能, 深度森林就是基于这种想法提出来的一种深度结构。

2017 年，深度森林由周志华老师提出，基于树模型的方法，主要使用集成学习思想方法的深度学习框架，也可作为一种在某些任务下替代深度神经网络的方法。



147

Figure 8 来自老师 PPT

三、计算题

1 计算卷积后的特征图

输入矩阵为，卷积核分别为下图所示，使用 Same 卷积，卷积步长为 1，激活函数为 ReLU。

5	6	0	1	8	2
2	5	7	2	3	7
0	7	2	4	5	6
5	3	6	9	3	1
6	5	3	1	4	6
5	2	4	0	8	7

1	-1	0
-1	1	-1
0	-1	1

-1	2	-1
1	5	1
-1	0	-1

卷积核

对于第一个卷积核大小为 3*3，same 卷积 padding 为 1，经过 relu

2	3	0	0	9	0
0	0	8	0	0	4
0	5	0	0	0	0
1	0	0	1	0	0
0	0	0	0	2	0
0	0	4	0	0	0

对于第二个卷积核大小为 3*3，same 卷积 padding 为 1，经过 relu

36	40	4	9	48	15
26	43	32	17	39	29
9	39	34	18	24	43
26	35	31	56	30	11
44	28	19	25	30	25
34	20	22	7	48	51

2、计算交叉熵

交叉熵损失函数计算公式为

$$L = -y \log y' - (1 - y) \log (1 - y') = \begin{cases} -\log y', & y = 1 \\ -\log (1 - y'), & y = 0 \end{cases}$$

Softmax 过程的公式为

$$S_i = \frac{e^i}{\sum_j e^i}$$

得到 softmax 值为 0.1990, 0.1950, 0.1874, 0.2010, 0.2177,

再进一步计算交叉熵损失，即 $-1 * \log(0.2177) = 1.5246$

3 平均绝对误差和均方误差分别为

输入 X_i

[0.42,0.46,0.53,0.58,0.7,0.88]

Label Y_i

[0.3,0.4,0.5,0.6,0.7,0.8]

平均绝对误差是 L1 损失
公式为

$$L_1 = \frac{1}{N} \sum_{i=0} |X_i - Y_i|$$

带入公式算得 0.0517

均方损失也是 MSE

$$L_2 = \frac{1}{N} \sum_{i=0} |X_i - Y_i|^2$$

0.0043

四、模型设计题

1. 请给出动作识别模型的设计方案，要求有自己的新思路和新观点。

2. 请给出机器翻译模型的设计方案，要求有自己的新思路和新观点。