# 《深度学习平台与应用》作业三
# 20251130

一、选择题

1. 关于循环神经网络（RNN）及 LSTM（Long Short Term Memory），下列说法中不正确的是：(D)

   A. 在标准的 LSTM 架构中，遗忘门（Forget Gate）$f_t$ 决定了上一时刻的状态$C_{t-1}$ 有多少信息被保留到当前时刻。

   B. 相比于 Vanilla RNN，LSTM 通过引入门控机制和细胞状态（Cell State）$C_t$ 的加法更新，能够缓解梯度消失问题，从而更容易捕捉长距离依赖。

   C. 在 LSTM 的更新公式中，输出门（Output Gate）$o_t$控制了当前隐藏状态 $h_t$ 的输出，其计算公式通常涉及 Sigmoid 激活函数。

   D. LSTM 中计算当前时刻隐藏状态 $h_t$ 的最终公式为 $h_t = o_t \odot \sigma(C_t)$，其中 $\sigma$ 代表 Sigmoid 函数。

2. 关于 Transformer 架构及其核心组件“注意力机制”（Attention Mechanism），下列说法正确的是：(B)

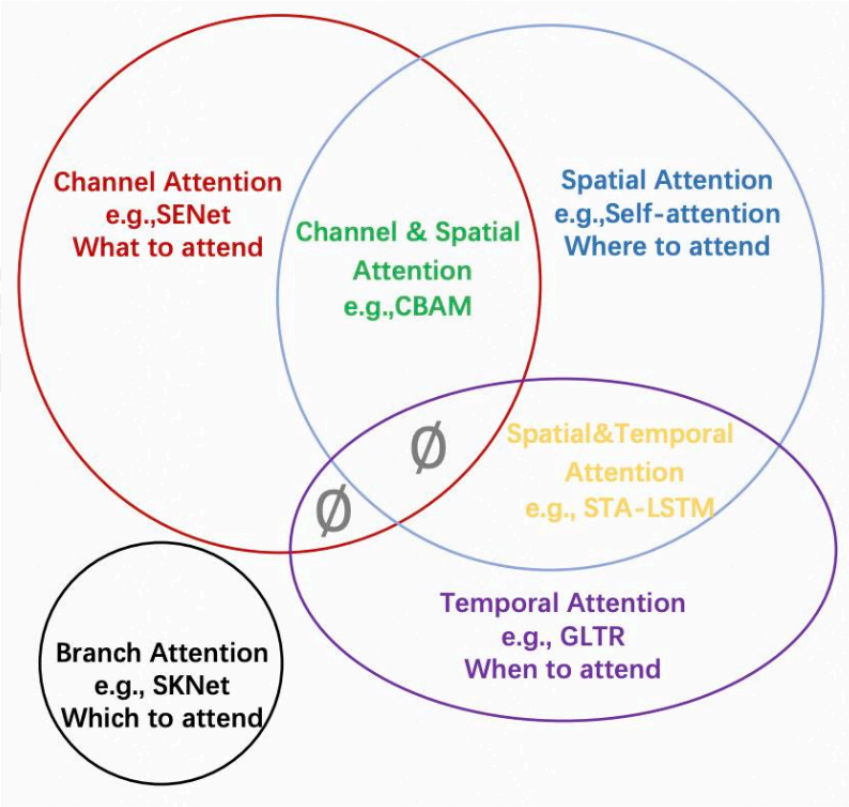
   A. 在自注意力（Self-Attention）计算中，除以缩放因子 $\sqrt{d_k}$ 的主要目的是为了放大点积结果，从而在反向传播时获得更大的梯度。

   B. Transformer 模型完全抛弃了递归结构（Recurrence），为了让模型利用序列的顺序信息，必须在输入嵌入（Input Embeddings）中加入位置编码（Positional Encodings）。

   C. 在 Transformer 的 Decoder 中，Masked Multi-Head Attention 的作用是让模型在训练时能够看到当前位置之后的所有单词，从而更好地进行上下文建模。

   D. 视觉 Transformer（ViT）将图像分割成多个 Patch，直接将这些 Patch 展平后输入到 Transformer Encoder 中，不需要任何额外的线性投影（Linear Projection）或位置编码。

3. 根据对注意力机制分类的总结，以下关于不同类型注意力机制的描述，正确的是：(C)




A. SENet (Squeeze-and-Excitation Networks) 是一种典型的空间注意力机制（Spatial Attention），主要关注“Where to attend”。

B. CBAM (Convolutional Block Attention Module) 仅包含通道注意力，不包含空间注意力。

C. Self-attention（自注意力机制）属于空间注意力（Spatial Attention）的一种，主要关注特征图的 H（高度）和 W（宽度）维度上的依赖关系。

D. 所有的注意力机制都必须同时包含通道（Channel）、空间（Spatial）和时间（Temporal）三个维度的建模。

4. 在 Transformer 架构中，Positional Encoding（位置编码）的主要作用和实现方式是： (C)

A. 通过卷积层提取位置特征，替换原有的 Input Embedding。

B. 将位置信息与 Input Embedding 进行拼接（Concatenate），以增加特征维度。

C. 因为 Self-Attention 机制本身无法捕捉序列的顺序信息，所以需要将位置编码与 Input Embedding 相加（Add），且通常使用正弦

和余弦函数生成。

D. 位置编码是一个可学习的权重矩阵，必须通过反向传播从头训练，不能使用固定的数学公式。

5. 关于 Vision Transformer (ViT) 如何将二维图像处理为 Transformer 能够接受的输入格式，下列描述正确的是：(C)

   A. 使用预训练的 CNN（如 ResNet）提取特征图，直接作为 Transformer 的输入。

   B. 将整张图片拉伸为一个长向量，直接进行线性投影。

   C. 将图像分割成固定大小的 Patches（例如 $16 \times 16$），将每个 Patch 展平后进行线性投影（Linear Projection），并加上位置编码。

   D. ViT 不需要位置编码，因为图像块本身具有空间结构。

6. 多头注意力（Multi-Head Attention）相比单头注意力的核心优势是什么？(B)

   A. 减少参数量，提升训练速度

   B. 允许模型并行关注不同子空间中的关系

   C. 避免使用缩放点积注意力

   D. 使注意力矩阵稀疏化，从而减少计算量

7. 在目标检测与分割模型的发展历程中（R-CNN 系列），针对特征对齐和计算效率的改进是关键。以下关于这些模型的描述，错误的是：(D)

   A. Fast R-CNN 引入了 RoI Pooling 层，使得我们不再需要对每个 Region Proposal 分别进行卷积计算，而是共享整张图的特征图，大大提高了训练和推理速度。

   B. Faster R-CNN 的核心创新在于提出了区域生成网络（RPN），它是一个全卷积网络，可以与检测网络共享卷积特征，实现了端到端的训练。

   C. Mask R-CNN 在 Faster R-CNN 的基础上增加了一个预测分割掩码（Mask）的分支，用于实例分割任务

D. 为了解决特征图与原始图像区域对齐的问题，Mask R-CNN 沿用了 Fast R-CNN 中的 RoI Pooling 操作，以保证特征提取的精确性。

8. R-CNN 被认为“非常慢”的主要原因是什么？(C)

   A. CNN 只能处理灰度图

   B. SVM 分类器训练困难

   C. 对每个 proposal 都独立运行 卷积网络处理，重复大量计算

   D. 特征图分辨率太低，导致需要重复补偿计算

## 二、问答题

1. 关于循环神经网络：

   (a): 请描述循环神经网络（RNN）的主要结构和流程。

   RNN 包含输入层、隐藏层和输出层。其核心在于隐藏状态（Hidden State）的循环连接。在每一时刻 $t$，隐藏层不仅接收当前时刻的输入 $x_t$，还接收上一时刻的隐藏状态 $h_{t-1}$。计算公式通常为 $h_t = \sigma(W_{xh}x_t + W_{hh}h_{t-1} + b)$。

   (b): RNN 展开之后的计算图长什么样？为什么说“各时间步共享参数”？

   **计算图**：展开后看作多个结构相同的神经网络单元按时间顺序串联，每个时刻 $t$ 输出 $h_t$ 并传给下一时刻。

   **参数共享**：指在所有时间步中，使用的权重矩阵（$W_{xh}, W_{hh}, W_{hy}$）是完全相同的。这意味着模型用同一套规则处理序列中的每一个元素。

   (c): 请分别举例说明 RNN 可以处理的几种输入/输出结构，以及对应的常见任务。

   **One-to-One**：固定输入到固定输出（如图像分类，非典型

RNN）。

**One-to-Many**：序列输出（如看图说话/Image Captioning）。

**Many-to-One**：序列输入，单个输出（如文本情感分析）。

**Many-to-Many**：序列到序列（如机器翻译、视频帧分类）。

(d): 为什么RNN速度慢。

RNN 的计算依赖于上一时刻的输出，导致无法像 CNN 或 Transformer 那样对整个序列进行并行计算，只能串行处理。

(e): 为什么RNN有梯度问题。

在反向传播时，梯度需要通过时间步连乘。如果权重矩阵的特征值小于 1，连乘会导致**梯度消失**（无法学习长距离依赖）；如果大于 1，会导致**梯度爆炸**。

(f): LSTM 相比普通 RNN，在结构上增加了哪些“门”？它们各自控制什么？

**遗忘门 (Forget Gate)**：控制上一时刻的细胞状态 $C_{t-1}$ 有多少信息需要保留。

**输入门 (Input Gate)**：控制当前时刻的新信息 $\tilde{C}_t$ 有多少需要更新到细胞状态中。

**输出门 (Output Gate)**：控制当前的细胞状态 $C_t$ 有多少需要输出为隐藏状态 $h_t$。

(g): 为什么说 LSTM 更有利于缓解RNN 的梯度问题?

- LSTM 引入了细胞状态 $C_t$，其更新主要是加法运算（$C_t = f_t C_{t-1} + i_t \tilde{C}_t$）。在反向传播时，梯度可以通过加法路径直接传递，避免了连乘带来的梯度快速衰减，从而保留长距离信息。

2. 关于Transformer模型:

(a): Transformer中有哪些关键组成。

Encoder（编码器）、Decoder（解码器）、Multi-Head Self-

Attention（多头自注意力）、Position-wise Feed-Forward Networks（前馈网络）、Positional Encoding（位置编码）、Residual Connection（残差连接）和 Layer Normalization（层归一化）。

(b): 请解释为什么Transformer中需要位置编码，位置编码是怎么添加的。

**原因**：Transformer 内部没有循环或卷积结构，无法识别输入序列的顺序（即打乱词序输入，Attention 输出结果不变）。

生成与 Input 维度相同的向量（通常使用正弦/余弦函数），直接与Input相加。

(c): 请解释self-attention机制的流程。

将输入向量分别乘以三个可学习矩阵 $W^Q, W^K, W^V$，得到 Query, Key, Value 向量。

计算 $Q$ 和 $K$ 的点积，衡量相关性。

除以 $\sqrt{d_k}$ 进行缩放，再经过 Softmax 得到注意力权重。

用权重对 $V$ 进行加权求和，得到最终输出。

(d): self-attention 和 multi-head attention 有哪些区别，multi-head attention 有什么好处。

**区别**：Self-attention 只有一组 $Q, K, V$；Multi-head 将 $Q, K, V$ 拆分成 $h$ 组，分别计算注意力后拼接。

**好处**：允许模型同时关注不同位置、不同表示子空间的信息，增强表达能力。

(e): Feed forward 层在Transformer中的作用是什么。

在注意力层提取全局特征后，通过两个全连接层（中间含 ReLU）对每个位置的特征进行非线性变换和特征整合，增强模型的拟合能力。

(f): Add & Norm 在Transformer中的作用是什么。

**Add (残差连接)**：解决深层网络退化问题，利于梯度传播。

**Norm**：稳定各层输入的分布，加速训练收敛。

(g): 比较卷积网络，双向循环网络，全连接网络和注意力机制的异同。

**CNN**：局部连接，权重共享，擅长提取局部特征，并行性高。

**RNN**：时序依赖，擅长处理序列，无法并行，长距离依赖弱。

**FC**：全局连接，参数量大，无空间/时序结构归纳偏置。

**Attention**：全局连接（捕捉长距离依赖），数据驱动的动态权重，计算复杂度与序列长度平方成正比。

(h): 什么是mask attention以及mask attention在Transformer中是如何应用的。

在计算 Attention Score 时，将某些位置的权重设为负无穷（Softmax 后为 0）

应用：在 Transformer 的 Decoder 中，用于防止模型在预测当前词时看到后面的词。

3. 关于目标检测和图像分割：

(a): 语义分割与目标检测的主要区别是什么。

**目标检测**：输出物体的边界框（Bounding Box）和类别。

**语义分割**：对图像中每一个像素进行分类，不区分同类别的不同个体。

(b): 什么是全卷积网络？它如何解决语义分割中的高分辨率问题?

将 CNN 最后的全连接层替换为卷积层，使网络可以接受任意尺寸输入并输出特征图。

**解决高分辨率**：使用转置卷积或上采样操作，将缩小的特征图恢复到原图尺寸，实现像素级预测。

(c): 目标检测中如何解决“多个物体”的问题?

传统方法使用滑动窗口。

现代方法使用候选区域 **(Region Proposals)**（如 R-CNN）或预定义的锚框 **(Anchors)**（如 YOLO, Faster R-CNN），网络回归每个框的

坐标偏移和类别。

(d): 什么是R-CNN？它如何提升目标检测的效率？

R-CNN：Region-based CNN。先使用选择性搜索提取候选框，然后缩放每个框并输入 CNN 提取特征，最后用 SVM 分类。选择性搜索相较于基于滑动窗口的暴力搜索算法，生成更可能包含物体的候选框，减少了需要处理的区域数量。

(e): Fast R-CNN 与 R-CNN 的区别是什么？为什么Fast R-CNN更高效？

R-CNN 对每个框独立执行一次CNN；Fast R-CNN 只对整图执行一次 CNN，然后在特征图上通过 **RoI Pooling** 提取各个框的特征。

**高效原因**：共享了卷积计算，避免了数千次重复的特征提取过程。

(f): 实例分割和语义语义分割有什么区别。

**语义分割**：只区分类别。

**实例分割**：区分类别还要区分同类的不同个体。

(g): Mask R-CNN 在目标检测的基础上，如何扩展为实例分割任务？

在 Faster R-CNN 的基础上，并行增加了一个 Mask 分支，用于为每个 RoI 输出一个二值掩码。同时将 RoI Pooling 改进为 RoI Align 以实现像素级对齐。