中级AI算法工程师招聘测试题

适用岗位: 中级算法工程师
考试时间: 120分钟
总分: 100分
姓名:_____ **日期:**_____

一、算法与数据结构进阶(25分)

1. 算法设计与优化(15分)

1.1 (8分) **Top-K问题**

给定一个未排序的整数数组 `nums` 和整数 `k`，找出数组中第 k 大的元素。

要求:

- 描述至少两种不同的算法（如堆、快速选择）
- 分析时间复杂度（平均和最坏情况）
- 如果需要频繁查询不同的k值（例如先查第5大，再查第10大），你会如何优化?
- 如果数据是流式的，无法全部加载到内存，如何近似求解?

1.2 (7分) **最长递增子序列**

给定整数数组，找出最长严格递增子序列(LIS)的长度。

要求:

- 给出$O(n^2)$的动态规划解法（状态定义和转移方程）
- 解释如何优化到$O(n \log n)$（提示：使用二分查找维护一个辅助数组）
- 如果要求输出具体的子序列，算法需要如何修改?

---

2. 图算法应用(10分)

2.1 (10分) **并查集(Union-Find)**

并查集用于处理不相交集合的合并和查询问题。

问题:

(a) (4分) 实现并查集的核心操作（用伪代码或文字描述）:

- find(x): 查找x所属的集合
- union(x, y): 合并x和y所在的集合

(b) (3分) 解释两种优化策略：

- 路径压缩(Path Compression)
- 按秩合并(Union by Rank)
- 这两种优化如何降低时间复杂度?

(c) (3分) 应用场景：

- 给定n个节点，初始没有边。进行m次"连接边"或"查询连通性"操作
- 分析使用并查集的时间复杂度
- 如果用DFS/BFS每次查询连通性，复杂度是多少？为什么并查集更优?

二、深度学习算法理解(35分)

3. Transformer与Attention机制(15分)

3.1 (8分) Attention机制深入理解

Self-Attention的计算公式：Attention(Q,K,V) = softmax(QK^T/√d_k)V

问题:

- 为什么要除以√d_k？从梯度稳定性角度解释（不要求严格推导，说明直觉即可）
- 标准Self-Attention的时间复杂度和空间复杂度是多少？对于长序列（如4096 tokens）的瓶颈在哪里?
- Multi-Head Attention相比Single-Head有什么优势？不同的head会学到什么?
- 在Encoder-Decoder架构中，有哪几种不同的Attention？它们的Q、K、V分别来自哪里?

3.2 (7分) 优化长序列Attention

标准Attention对长序列的$O(n^2)$复杂度是瓶颈。请介绍至少两种优化方法：

选择以下任意两种详细解释：

- Sparse Attention（稀疏注意力）
- Sliding Window Attention（滑动窗口）
- Flash Attention（内存优化）

- Linear Attention（线性复杂度）

每种方法需说明：

- 核心思想是什么
- 如何降低复杂度
- 可能的性能损失

---

4. 反向传播与优化器(12分)

4.1 (6分) 反向传播理解

考虑一个包含Batch Normalization的网络层：

```
输入 x -> BN -> ReLU -> 输出 y
```

问题:

- 简述Batch Normalization的前向计算步骤（归一化 + 缩放平移）
- 反向传播时，BN层的梯度需要考虑哪些因素？为什么比普通线性层复杂?
- ReLU的反向传播如何计算？如果输入是负数会发生什么?
- 为什么"梯度消失"在深层网络中是个问题？BN如何缓解这个问题?

4.2 (6分) 优化器对比

问题:

- SGD、SGD with Momentum、Adam的主要区别是什么?
- Adam为什么通常比SGD收敛更快？它结合了哪两种方法的优点?
- 什么是learning rate warm-up？为什么训练大模型时需要?
- AdamW相比Adam改进了什么？为什么weight decay的实现方式很重要?
- 在什么情况下SGD可能比Adam效果更好？（提示：泛化性能）

---

5. 模型架构设计(8分)

5.1 (4分) 残差连接(Residual Connection)

ResNet引入的残差连接：$y = F(x) + x$

问题:

- 为什么残差连接能让网络训练得更深？从梯度流的角度解释
- 在Transformer中，残差连接放在哪些位置？
- Pre-LN (Add & Norm)和Post-LN (Norm & Add)有什么区别？哪个更常用？

5.2 (4分) **位置编码(Positional Encoding)**

Transformer需要位置编码因为Self-Attention对位置不敏感。

问题:

- 原始Transformer使用的正弦位置编码公式是什么样的？
- 为什么用正弦函数而不是简单的整数索引？
- 可学习的位置编码和固定的位置编码各有什么优缺点？
- 相对位置编码(Relative Positional Encoding)相比绝对位置有什么优势？

三、机器学习理论(25分)

6. 概率与生成模型(10分)

6.1 (6分) **变分自编码器(VAE)**

VAE通过最大化ELBO训练：

```
ELBO = E[log p(x|z)] - KL(q(z|x) || p(z))
```

问题:

- VAE的编码器和解码器分别输出什么？
- 为什么需要KL散度这一项？它起什么作用？
- 什么是reparameterization trick？为什么不能直接从q(z|x)采样？
- VAE生成的图像为什么通常比较模糊？

6.2 (4分) **对比：VAE vs GAN**

- VAE和GAN的训练目标有什么本质区别？
- 各自的优缺点是什么？
- 在什么场景下会选择VAE而不是GAN？

---

7. 损失函数与正则化(8分)

7.1 (4分) **损失函数设计**

问题:

- 分类任务为什么用交叉熵而不是MSE?
- 在类别不平衡的情况下，如何调整损失函数？（Focal Loss, Class Weights等）
- 对比学习(Contrastive Learning)中的InfoNCE损失的核心思想是什么?
- 什么时候需要使用Huber Loss而不是MSE或MAE?

7.2 (4分) **正则化技术**

问题:

- L1和L2正则化的区别是什么？为什么L1会产生稀疏解?
- Dropout在训练和推理时的行为有何不同？为什么?
- Label Smoothing如何防止过拟合？原理是什么?
- Data Augmentation可以看作一种正则化吗？为什么?

---

8. 评估与调试(7分)

8.1 (4分) **模型评估指标**

对于二分类问题：

- 准确率、精确率、召回率、F1分数的定义
- 在什么情况下准确率会误导？举例说明
- ROC曲线和PR曲线有什么区别？什么时候用PR曲线更合适?
- 如何评估一个多分类模型？（macro/micro/weighted averaging）

8.2 (3分) **训练问题诊断**

如果训练中遇到以下情况，可能的原因和解决方法是什么?

- Loss变成NaN
- Loss不下降或震荡
- 训练集loss下降但验证集loss上升

四、大模型训练基础(15分)

9. 分布式训练策略(10分)

9.1 (6分) **并行策略理解**

假设训练一个7B参数的大模型，每张GPU有40GB显存。

问题:

- 训练时需要存储哪些内容？（权重、梯度、优化器状态、激活值）
- 估算一个batch需要多少显存（FP16权重 + FP32优化器状态）
- 数据并行(Data Parallel)是如何工作的？通信的瓶颈在哪里？
- 模型并行(Tensor Parallel)在Transformer中如何切分？举例说明
- 什么是梯度累积(Gradient Accumulation)？什么时候需要用？

9.2 (4分) **训练效率优化**

问题:

- 混合精度训练(Mixed Precision)的原理是什么？为什么能加速？
- FP16训练可能遇到什么问题？Loss Scaling如何解决？
- Gradient Checkpointing如何权衡显存和计算时间？
- 如何监控训练效率？哪些指标是重要的？

---

10. 代码实现与工程(5分)

10.1 (5分) **训练代码常见问题**

以下是一个简化的训练循环：

```python
for epoch in range(num_epochs):
    for batch in dataloader:
        outputs = model(batch['input'])
        loss = criterion(outputs, batch['labels'])
        loss.backward()
        optimizer.step()
        optimizer.zero_grad()
```

问题:

- 这段代码有什么潜在问题？（提示：梯度清零的位置、梯度裁剪、设备管理等）
- 如何添加梯度裁剪？为什么需要？
- 如果使用混合精度训练，需要添加什么？
- 如何保存checkpoint？应该保存哪些内容？
- 训练突然loss变NaN，如何debug？列出可能的原因和排查步骤

说明: 本测试适用于中级算法岗位，建议达到70分以上。重点考察深度学习算法理解、实践经验和问题解决能力。