**一、填空题（共 10 分，每空 1 分）**

1. 自然语言具有 \_\_歧义性\_\_\_\_\_\_\_（常见现象）。
2. 信息抽取任务通常可分解为 实体关系抽取 和 命名实体识别\_\_.
3. 在知识图谱中，节点表示 \_\_实体\_\_\_\_\_\_\_，有向边表示 \_实体间的关系
4. 对于“北京师范大学”，其 trigrams（3-gram）依次是  
   ***北京师、京师范***、师范大***、范大学***.  
   N-gram 模型所作的核心假设称为 \_马尔科夫链\_\_\_\_\_ 假设。  
   假设当前状态**只与前一状态有关**，而**与更早的状态无关**
5. LSTM 主要解决了传统 RNN 中的 \_\_\_梯度消失\_\_\_\_ 问题。  
   注意：仅是梯度消失，而非梯度爆炸

对于梯度爆炸，通常的做法是：梯度裁剪（Gradient Clipping）：在反向传播时，将梯度的范数（或每个维度）裁剪到一个阈值以内，防止梯度过大导致参数更新发散。其他常用手段还有：良好地初始化参数、使用权重正则化或使用更稳定的优化器（如 Adam、RMSProp）等。

1. 对预训练语言模型进行下游任务微调时，常用的参数更新方法是 \_反向传播梯度\_\_\_\_\_. 微调？？？？
2. 为了解决 OOV（out-of-vocabulary）问题，解码时可采用 \_\_子词分割\_\_\_\_\_\_\_ 方法。
3. Transformer 为了让模型在不破坏**自注意力**并行的前提下仍**保留单词顺序信息**，采用了 \_\_位置编码\_\_\_\_\_\_\_.
4. \_\_\_Encoder/编码器\_\_\_\_ 预训练范式可以在训练时同时利用左右两个方向的上下文信息，而不做 next-token 预测。

**二、判断题（共 10 分，每题 2 分）**

请在下列说法后面打“√”（正确）或“×”（错误）：

1. LSTM 是 RNN 的一种改进。                   （√）
2. TextRank 是一种生成式信息抽取算法。           （×）
3. BART 架构只包含 decoder 部分。            （×）
4. 自回归（next-token）语言模型只能利用单向上下文。     （√）
5. 依存句法分析的结果可以表示为一棵有向无环树。     （√）

**三、简答题（共 40 分，每题 10 分）**

**1. 简述自然语言处理（NLP）研究的主要内容。**

**答案要点**

* **分析理解**：分词/标注、句法分析、语义表示、指代消解、情感分析等；
* **信息抽取**：实体识别、关系抽取、事件抽取；
* **语言生成**：机器翻译、文本摘要、对话系统、代码生成等；
* **检索与对话**：问答系统、信息检索、对话管理；
* **预训练与迁移**：大规模语言模型预训练及下游微调。

**2. 简述 self-attention（自注意力）的原理与作用。**

**答案要点**

* **原理**：将输入序列每个位置映射为 Query、Key、Value，计算所有位置之间的注意力权重：  
  。
* **作用**：
  1. 允许序列中每个位置直接“看到”全局上下文；
  2. 捕获长距离依赖，无需逐步传递；
  3. 并行计算，提高效率；
  4. **可扩展到多头机制，学习不同子空间的依赖模式**。

**3. 简述预训练语言模型的三种主要范式。**

**答案要点**

1. **自回归语言模型（Autoregressive LM）**：如 GPT 系列，按序列左→右（或右→左）预测下一个词；最大化序列似然。

**特点：生成能力强，适合对话，代码等生成任务。**

1. **掩蔽语言模型（Masked LM）**：如 BERT，随机遮蔽若干词，利用双向上下文预测它们；

**强于句子级别/片段级别的表示学习，适合下游的分类，序列标注，句子匹配等理解任务。**

1. **序列到序列／Encoder-Decoder LM**：如 BART、T5，将原始序列映射到另一序列，可做复合 Mask／删除／重排等任务。

既有编码器提取上下文表示，又有解码器生成输出，适合翻译，摘要，结构化转换等需要“理解-生成一体化”的任务。

**4. 解释 zero-shot、one-shot、few-shot 学习，并举例说明。**

**答案要点**

* **Zero-shot**：无示例，直接给自然语言提示（prompt），让模型完成新任务。
  + 例：“Translate to French: Hello.”
* **One-shot**：提示中含有一条示例；
  + 例：“English→French: Hello→Bonjour. Goodbye→?”
* **Few-shot**：提示中含多条示例（多于 1 条，少于几十条）；
  + 例：“Cat→猫；Dog→狗；Bird→?”

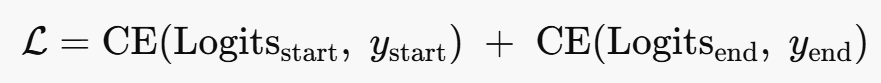
**四、综合题（共 20 分）**

**题目**：请分别设计一个基于 BERT 模型和一个基于 GPT 模型的阅读理解（Machine Reading Comprehension）微调方案，内容至少包括：

**参考答案要点**

**A. BERT-Based 阅读理解微调**

1. **数据准备**
   * 格式：常用 SQuAD 格式，输入为 (question, context)，标签为 answer span 的起止下标 (start, end)。
   * 将 context 和 question 拼接：[CLS] question [SEP] context [SEP]。
2. **微调任务与损失**
   * 增加两个全连接输出头，分别预测 start 和 end 索引。
   * 损失：start、end 两个 softmax-交叉熵之和。



1. **超参数与训练策略**
   * Learning rate：～
   * Batch size：16 或 32
   * Epochs：2–4
   * 梯度截断、权重衰减、Warm-up 和线性 Decay 学习率。
2. **评估指标**
   * Exact Match (EM)
   * F1 Score

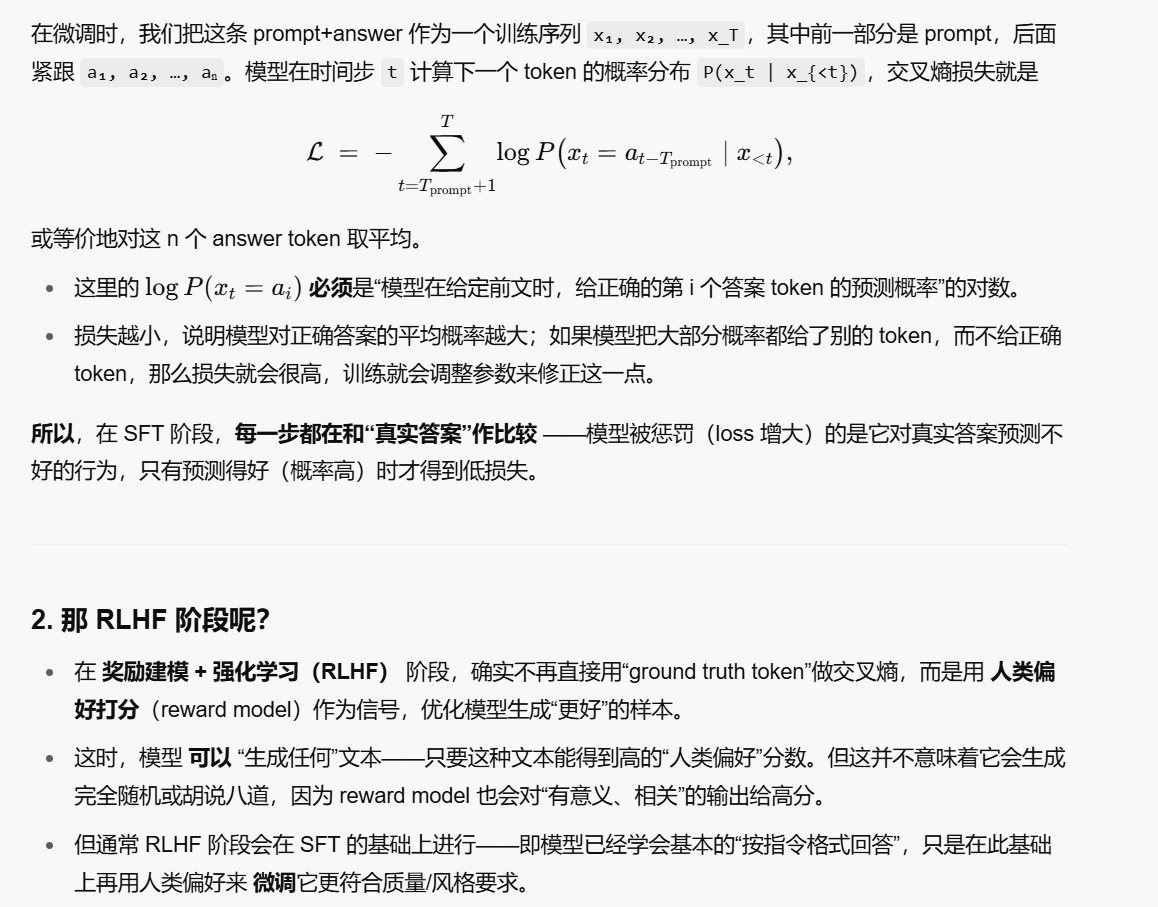
**B. GPT-Based 阅读理解微调**

1. **数据准备**
   * 将 (question, context, answer) 组织为串联式输入：

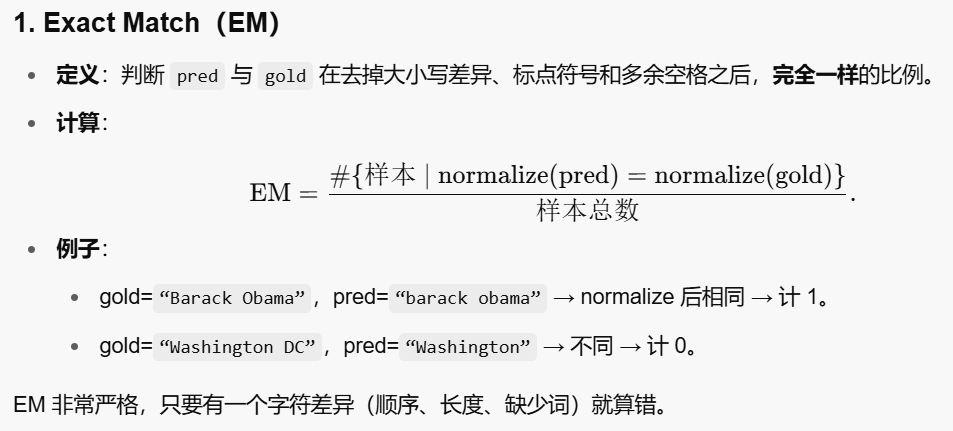
“Context: <context> Question: <question> Answer:”

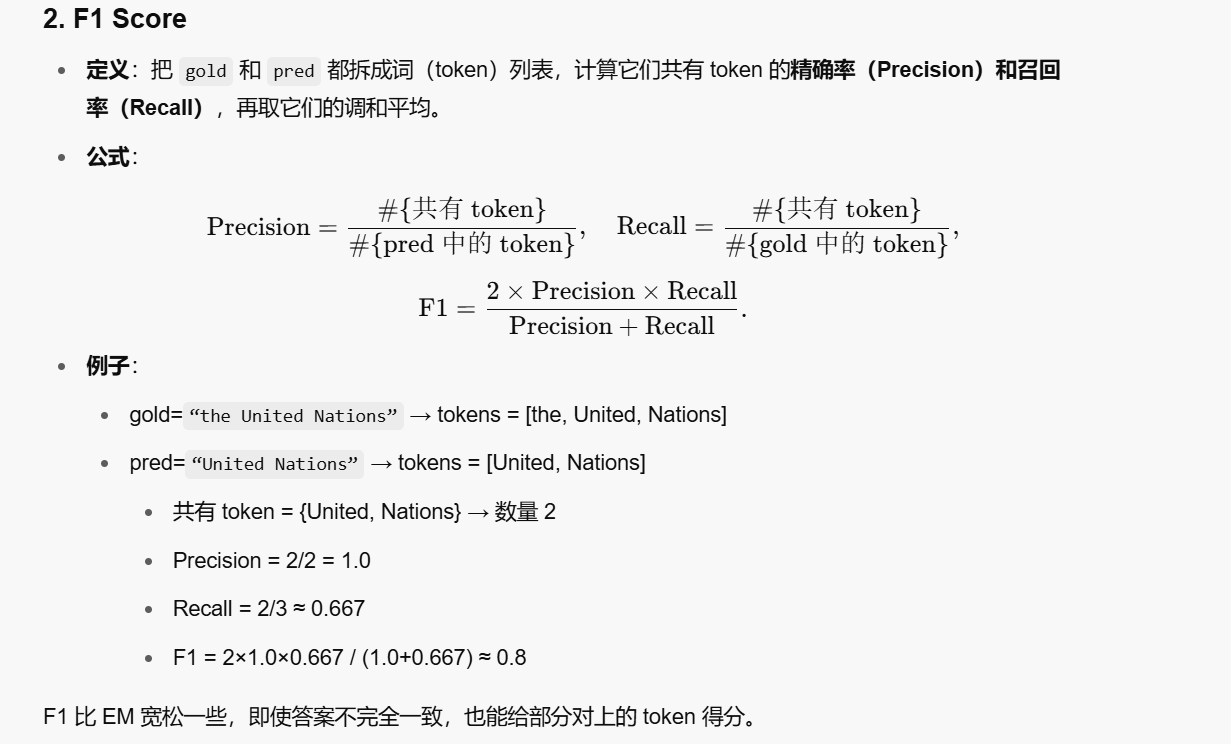
* + 目标：让模型在 Answer: 后继续生成正确答案词串。

1. **微调任务与损失**
   * 自回归 Language Model 任务：对上述 prompt+答案整体做 next-token 预测。
   * 损失：交叉熵损失，计算模型生成答案部分的所有 token 的概率。



1. **超参数与训练策略**
   * Learning rate：～
   * Batch size：8–16
   * 梯度累积、多卡并行
   * 可能使用 Label Smoothing 或对答案长度加权
2. **评估指标**
   * EM / F1（按生成答案与参考答案做字符串或 token-level 对齐比较）
   * 生成质量：BLEU、ROUGE（可选）





困惑度：

