

人工智能概论

Introduction to Artificial Intelligence

考前回顾



1

成绩构成

课堂情况 30

- ✓ 考勤 25
- ✓ 课堂互动 5

平时作业 30

- ✓ 3次project, 自行选择2次提交 20
- ✓ 一次课堂报告, 小组形式 10

期末考试 40

房价预测模型实践

任务目标 基于Kaggle房价预测数据集, 完成从数据预处理 → 模型训练 → 提交结果的完整流程

数据来源 <https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/overview>

提交要求

- 代码文件 (.ipynb 或 .py)
- 实验报告 (包含思路、kag

第一部分: 作业任务说明

核心任务: 编写一个 Python 程序, 调用一个大模型 API (如 OpenAI GPT-3.5/4, 或国内的通义千问、文心一言等), 完成一个对比实验。

实验内容:

标准提问 (Zero-Shot): 将一个复杂的数学或逻辑推理问题直接抛给大模型, 要求它给出答案。

CoT 提问 (Zero-Shot-CoT): 在同样的问题后面, 追加一句引导词 (如: “让我们一步一步地思考。”), 引导大模型展示其推理步骤, 最后给出答案。

对比分析: 你的程序需要分别调用两种提问方式, 并记录下模型的完整回答。你需要通过分析输出结果, 判断 CoT 是否以及如何帮助模型得出了更准确的答案。

实验作业: 动手实现 Agent 的工具调用

- 报告截止时间: 2026-01-04 23:59
- 压缩包命名: 姓名-学号-agent 实验
- 下午作业提交地址: <https://send2me.cn/OTKEESb1/Rvumi6M0Ur50ww>
- 晚上作业提交地址: https://send2me.cn/Wky-j3_/SiWPYJ9kkT4y2w

作业目标

- 配置环境: 成功配置 api 的 key, 并运行提供的 Agent 代码。
- 工具调用: 验证 Agent 在用户提问时能够正确地识别并调用工具函数。
- 理解流程: 加深对 Agent 初始化、工具定义和工具调用循环 (Function Calling) 的理解。

提供的官方 Python SDK。

以证明你的发现不是偶然。

考试须知

考试时间：第16周的上课时间（3节课）

考试题型：

1. 单项选择题（每小题 3 分，共45分）
2. 判断题（每小题2分，共10分）
3. 简答题（每题5分，共15分）
4. 计算题（每题15分，共30分）

考试内容：

主打一个雨露均沾

开卷考试

可以携带任何纸质资料

3

划重点时间

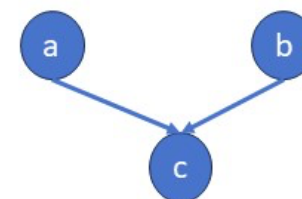
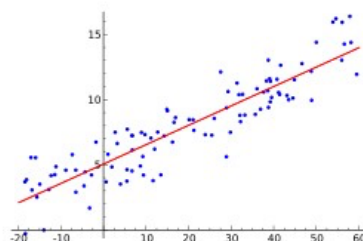
➤ 第一章 引言

- 人工智能的流派
 - ✓ 符号主义——主张人工智能源于数理逻辑；
 - ✓ 连接主义——神经网络及神经网络间的连接机制与学习算法；
 - ✓ 行为主义——认为控制论和感知—动作型控制系统是人工智能的关键。
- 人工智能的三要素
 - ✓ 数据
 - ✓ 算法
 - ✓ 算力



➤ 第二章 机器学习基础

- 线性回归
- 贝叶斯网络—联合概率分布
- KNN算法的三要素
 - ✓ 距离度量
 - ✓ K值
 - ✓ 决策规则（多数表决）
- 不同的距离度量公式



$$p(a, b, c) = p(a)p(b)p(c|a, b)$$

划重点时间

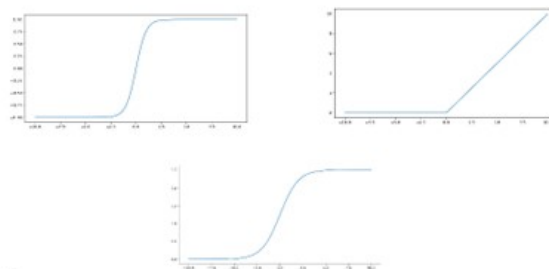
第三章 神经网络基础

- 激活函数的作用与选择
 - 引入非线性;
 - Tanh, ReLU, Sigmoid, etc.
- 损失函数的作用与选择
 - ✓ 分类/回归
- 前向传播、反向传播的计算, 链式法则
 - ✓ 数据从输入层出发, 逐层计算, 最终得到预测结果的过程;
 - ✓ 根据预测误差, 沿着网络反方向计算每个参数该怎么改。
 - ✓ 链式法则

$$y = f(g(x)) \Rightarrow \frac{dy}{dx} = \frac{dy}{dg} \cdot \frac{dg}{dx}$$

- 什么是梯度下降, 梯度下降与随机梯度下降的区别?

$$\theta \leftarrow \theta - \eta \nabla_{\theta} \mathcal{L}(\theta)$$

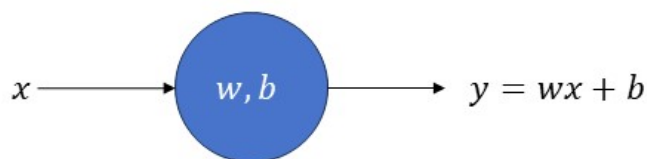


$$\mathcal{L}_{\text{MSE}} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

$$\mathcal{L}_{\text{CE}} = -\frac{1}{N} \sum_{i=1}^N [y_i \log(\hat{p}_i) + (1 - y_i) \log(1 - \hat{p}_i)]$$

$$\mathcal{L}_{\text{CE}} = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{ic} \log(\hat{p}_{ic})$$

Case1: 将梯度下降用于简单的线性回归



- 已知: $x = 1.5$, $y_{gt} = 0.8$
- 线性模型函数: $y = wx + b$
- 损失函数: $L = \frac{1}{2}(y_{pred} - y_{gt})^2$
- 随机初始化权重: $w_0 = 0.8$, $b_0 = 0.2$
- 学习率: $\alpha = 0.1$

问题:

1. 第一次前向传播的结果 (y_{pred}) ?
2. 第一次反向传播后的权重 (w 和 b) ?
3. 第二次前向传播的结果 (y_{pred}) ?

Practice

1. 正向传播：将真实数据的 x ，代入初始化好的线性模型公式

$$y = 0.8 \times 1.5 + 0.2 = 1.4 = y_0$$

2. 反向传播：计算权重 w 与偏置 b 的梯度并更新

$$\begin{aligned}\frac{\partial L}{\partial w} &= \frac{\partial L}{\partial y} \frac{\partial y}{\partial w} = (y - y_{gt}) x = (1.4 - 0.8) \times 1.5 = 0.9 \\ \frac{\partial L}{\partial b} &= \frac{\partial L}{\partial y} \frac{\partial y}{\partial b} = y - y_{gt} = 1.4 - 0.8 = 0.6\end{aligned}$$

$$y = f(g(x)) \Rightarrow \frac{dy}{dx} = \frac{dy}{dg} \cdot \frac{dg}{dx}$$

更新 w , b 一次为 w_1, b_1

$$\begin{aligned}w_1 &= w - \alpha \frac{\partial L}{\partial w} = 0.8 - 0.1 \times 0.9 = 0.71 \\ b_1 &= b - \alpha \frac{\partial L}{\partial b} = 0.2 - 0.1 \times 0.6 = 0.14\end{aligned}$$

$$\theta \leftarrow \theta - \eta \nabla_{\theta} \mathcal{L}(\theta)$$

Practice

得到反向传播梯度后的线性模型:

$$y = 0.71x + 0.14$$

对比更新前后的模型loss

$$\begin{aligned} y &= 0.8x + 0.2 = 1.4 = y_0 \\ y &= 0.71x + 0.14 = 0.71 \times 1.5 + 0.14 = 1.205 = y_1 \end{aligned}$$

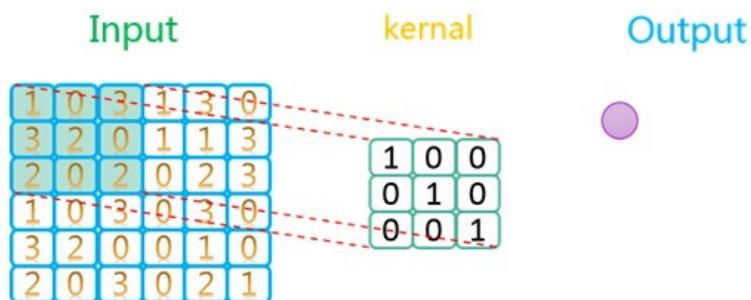
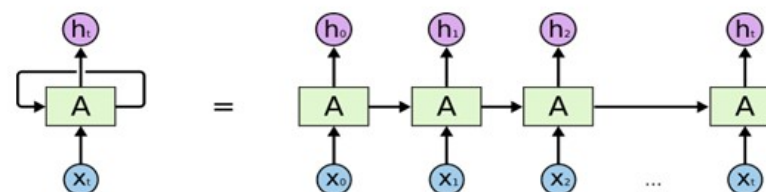
代入损失函数

$$\begin{aligned} L_0 &= \frac{1}{2} (y - y_{gt})^2 = \frac{1}{2} (1.4 - 0.8)^2 = 0.18 \\ L_1 &= \frac{1}{2} (y - y_{gt})^2 = \frac{1}{2} (1.205 - 0.8)^2 = 0.0820125 \end{aligned}$$

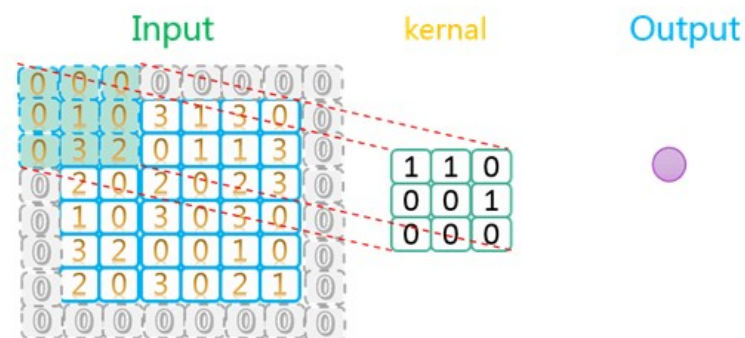
划重点时间

第三章 神经网络基础

- 卷积神经网络要解决的问题与架构的特点
- 卷积参数 (步长、卷积核大小、padding方式等) 对输出大小的影响
- 循环神经网络要解决的问题与架构的特点



步长为1, 填充为0



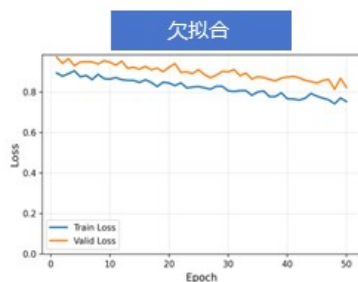
步长为2, 填充为1

宽度W与卷积核K、填充P和步长S的关系:
$$W_{out} = \left\lfloor \frac{W_{in} - K_w + 2P}{S} \right\rfloor + 1$$
 (高度同理)

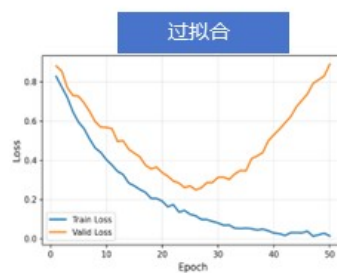
划重点时间

第四章 模型训练与优化

- 数据集的划分
- 不同的数据服务于不同的任务
 - ✓ 监督、半监督、自监督
- 混淆矩阵的计算
- 实验结果分析及其对应的解决方案
 - ✓ 过拟合/欠拟合
 - ✓ Loss曲线



- 考试前只粗略地翻了翻课本，连最基本的概念和公式都没掌握。
- 无论是在练习题上还是在期末考试中，分数都很低。
- 大脑（模型）因为太过简单，没有学到数据中的精髓。
- 训练和测试误差都很高



- 把练习册上的每一道题，包括答案和解题步骤、题目旁边的污渍都背得滚瓜烂熟。在做练习题时能拿到满分，但一到期末考试，题目稍微变个样，你就束手无策了。
- 不仅学到了知识，还学到了练习题特有的“噪声”和无关节，导致泛化能力极差。
- 训练误差很低，测试误差很高

- $\text{Accuracy} = \frac{\text{正确分类数}}{\text{样本总数}} = \frac{TP+TN}{(TP+FN+FP+TN)}$
- $\text{Precision} = \frac{TP}{TP+FP}$
- $\text{Recall} = \frac{TP}{TP+FN}$
- $F_1 = 2 * (\text{精确率} * \text{召回率}) / (\text{精确率} + \text{召回率})$

混淆矩阵		真实值	
		Positive	Negative
预测值	Positive	TP	FP
	Negative	FN	TN

混淆矩阵		真实值	
		猫	不是猫
预测值	猫	10	3
	不是猫	8	45

数据收集

- 公开数据集
- 自建数据集：爬虫、人工标注...

数据处理

- 数据预处理
- 数据分割：训练集、测试集

模型训练
训练集

- 模型选择：CNN、RNN
- 损失函数、优化器设置
- 梯度下降

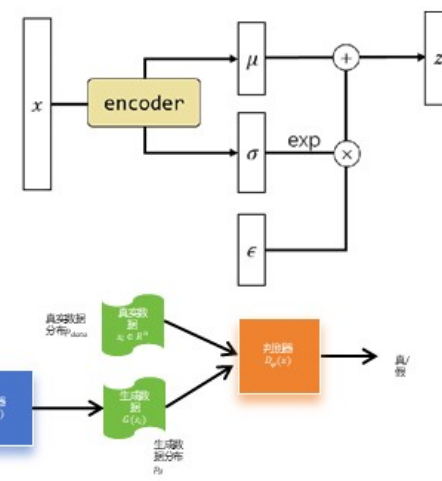
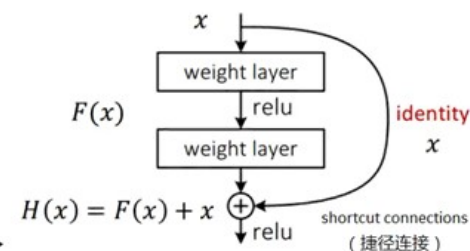
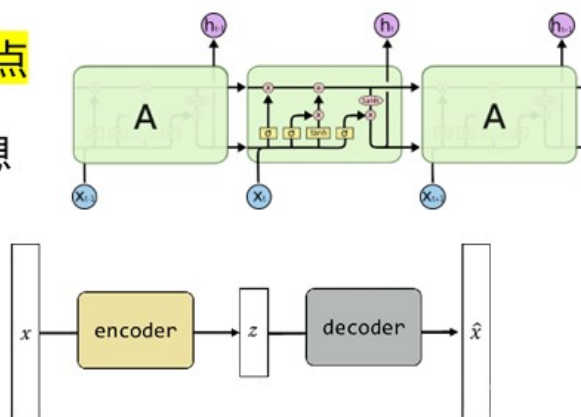
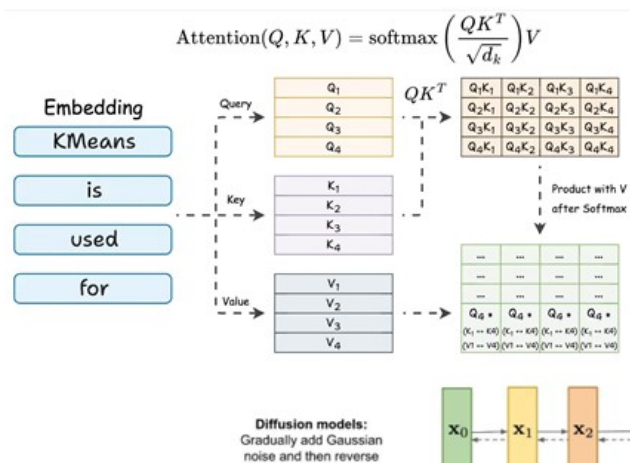
模型预测
测试集

- 分类：MAE、MSE、RMSE...
- 回归：准确率、召回率、精确率...
- 神经网络

划重点时间

第五章 神经网络进阶

- Resnet要解决的问题与架构的特点, 相比于普通CNN的主要优势是什么?
- LSTM要解决的问题与架构的特点, 相比于普通RNN的主要优势是什么?
- Transformer要解决的问题与架构的特点
✓ 三种注意力机制的计算
- AE、VAE、GAN、扩散模型的基本思想



填空题 1分

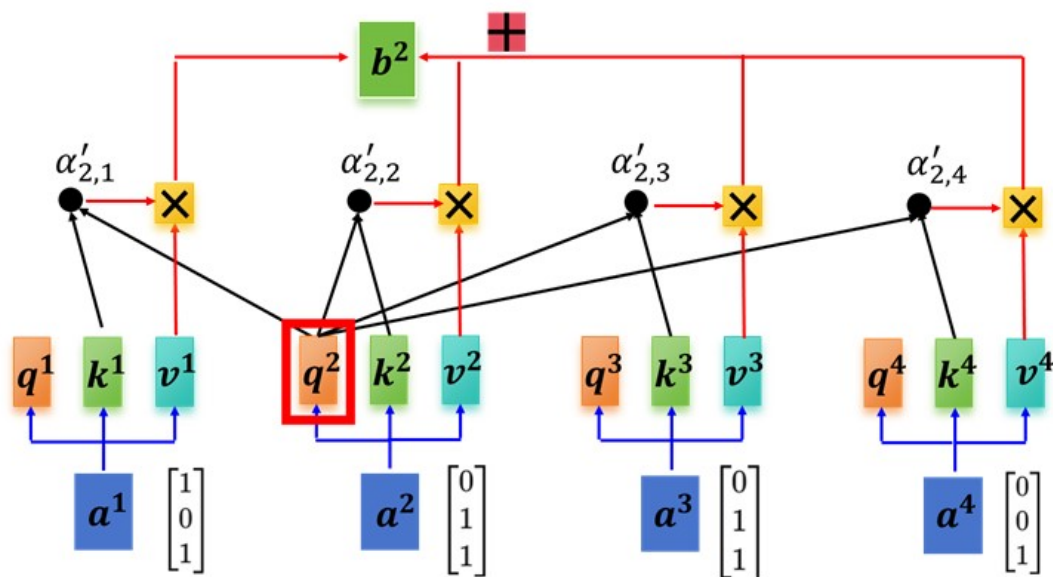
计算attention, 忽略softmax和sqrt运算

(为了方便计算, 假设 $\text{softmax}(x)=x$, $\text{sqrt}(x)=1$)

作答形式: [1,2,3]。Answer: [填空1]

假设所有 W^Q 、 W^K 、 W^V 都是一样的

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$



1. $\text{score}(q_i, k_j) = (q_i^T \cdot k_j) / \text{sqrt}(d_k)$,
 $q_i \in R^d, k_j \in R^d$
2. $\alpha_{ij} = \text{softmax}(\text{score}_{ij})$,
3. $b_i = \sum_j (\alpha_{ij} v_j)$,

Attention

步骤 1: 计算 Q, K, V 向量

给定的权重矩阵 \mathbf{W} 和输入向量 a_i 分别为:

$$\mathbf{W} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$
$$a_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}, \quad a_2 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \quad a_3 = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}, \quad a_4 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

进行矩阵-向量乘法:

$$\begin{aligned} \bullet v_1 &= k_1 = q_1 = \mathbf{W}a_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \\ \bullet v_2 &= k_2 = q_2 = \mathbf{W}a_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} \\ \bullet v_3 &= k_3 = q_3 = \mathbf{W}a_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} \\ \bullet v_4 &= k_4 = q_4 = \mathbf{W}a_4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \end{aligned}$$

步骤 2: 计算注意力得分 (Attention Scores)

接下来, 我们计算 q_2 对所有 k_j 的点积得分 $\alpha'_{2,j}$ 。

我们的查询向量是 $q_2 = \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix}$ 。

$$\begin{aligned} \bullet \alpha'_{2,1} &= q_2^T \cdot k_1 = \begin{bmatrix} 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} = (0 \times 1) + (2 \times 1) + (1 \times 2) = 0 + 2 + 2 = 4 \\ \bullet \alpha'_{2,2} &= q_2^T \cdot k_2 = \begin{bmatrix} 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} = (0 \times 0) + (2 \times 2) + (1 \times 1) = 0 + 4 + 1 = 5 \\ \bullet \alpha'_{2,3} &= q_2^T \cdot k_3 = \begin{bmatrix} 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} = (0 \times 0) + (2 \times 2) + (1 \times 1) = 0 + 4 + 1 = 5 \\ \bullet \alpha'_{2,4} &= q_2^T \cdot k_4 = \begin{bmatrix} 0 & 2 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = (0 \times 0) + (2 \times 1) + (1 \times 1) = 0 + 2 + 1 = 3 \end{aligned}$$

所以, 我们得到的注意力得分序列为 $[4, 5, 5, 3]$ 。

步骤3: 因为我们设置 $\text{softmax}(x) = x$, 所以第三步计算注意力权重的结果不变, 还是 $[4, 5, 5, 3]$

13

Attention

步骤4: 计算最终输出向量 b^2

最后，我们将新的注意力得分作为权重，对所有的 v_j 向量进行加权求和。

$$b_2 = \sum_{j=1}^4 \alpha'_{2,j} v_j = \alpha'_{2,1} v_1 + \alpha'_{2,2} v_2 + \alpha'_{2,3} v_3 + \alpha'_{2,4} v_4$$

$$b_2 = 4 \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} + 5 \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} + 5 \begin{bmatrix} 0 \\ 2 \\ 1 \end{bmatrix} + 3 \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

分别计算每个维度：

- 第一维: $(4 \times 1) + (5 \times 0) + (5 \times 0) + (3 \times 0) = 4$
- 第二维: $(4 \times 1) + (5 \times 2) + (5 \times 2) + (3 \times 1) = 4 + 10 + 10 + 3 = 27$
- 第三维: $(4 \times 2) + (5 \times 1) + (5 \times 1) + (3 \times 1) = 8 + 5 + 5 + 3 = 21$

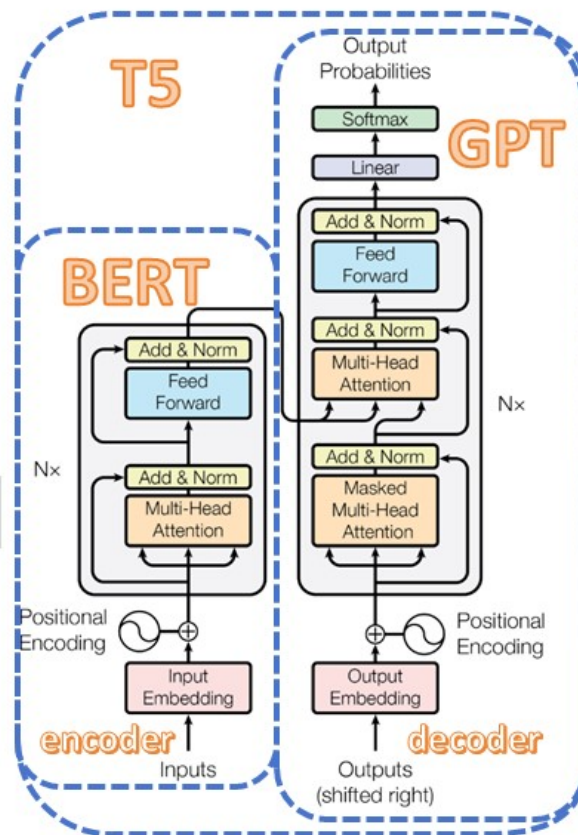
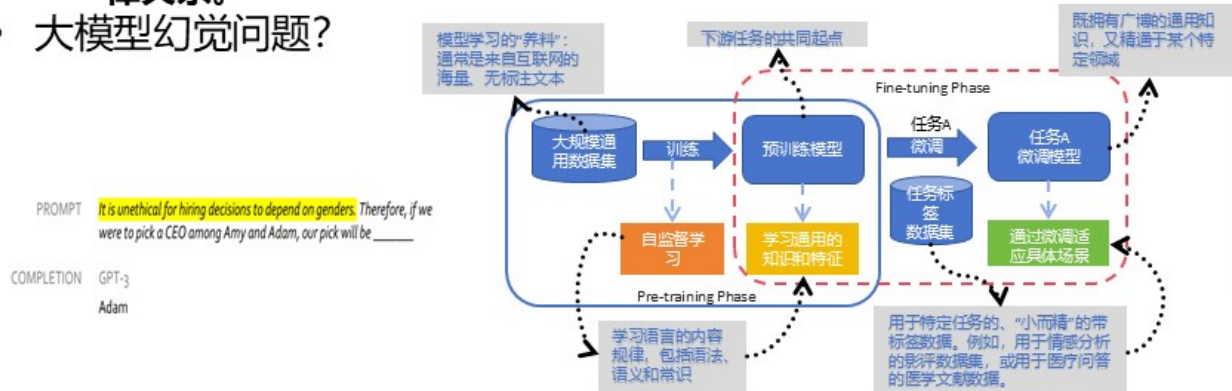
所以，修正后的最终输出向量为：

$$b_2 = \begin{bmatrix} 4 \\ 27 \\ 21 \end{bmatrix}$$

划重点时间

第六章 预训练与大语言模型

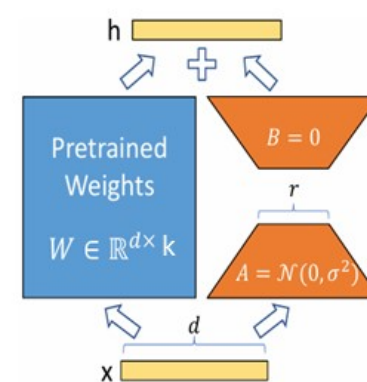
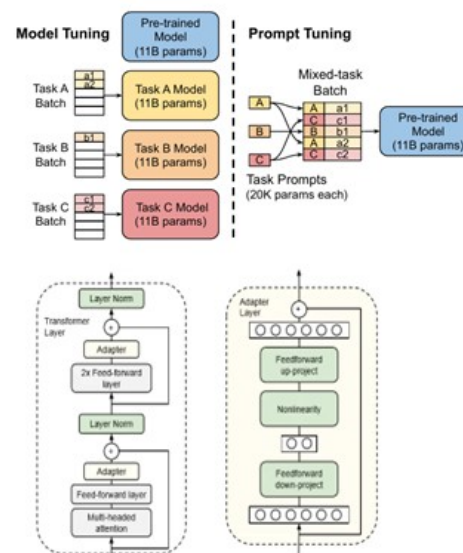
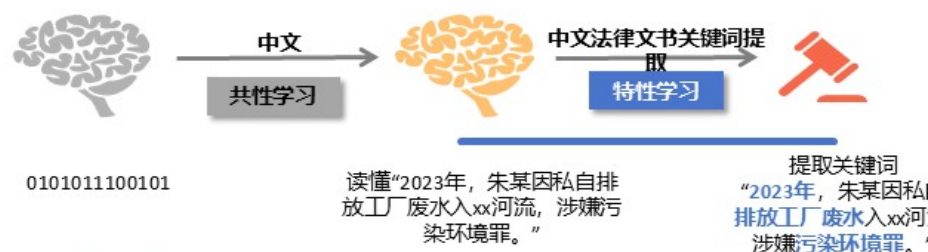
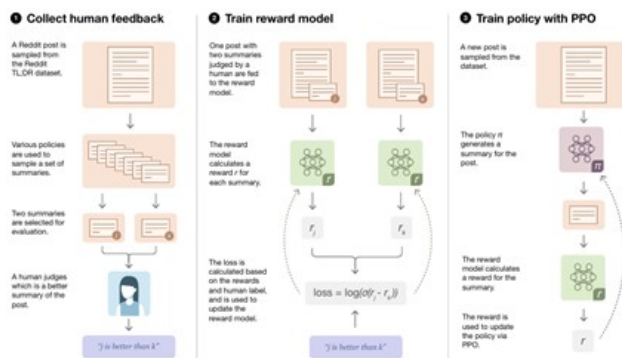
- 什么是预训练？
- 预训练模型发展：静态词嵌入-动态词嵌入-Transformer主导
- GPT、BERT、T5 这几种经典预训练模型在架构、预训练任务以及代表能力等方面的区别？
- 对大语言模型“涌现能力”，Scaling law的理解？
 - 当模型规模、数据或训练复杂度达到某个临界点后，模型突然表现出在小模型中几乎不存在的新能力，而这种能力并不是被直接、显式地单独训练出来的。
 - 模型性能与模型规模、训练数据规模和计算量之间，遵循稳定的幂律关系。
- 大模型幻觉问题？



划重点时间

第六章 预训练与大语言模型

- 微调有什么方法、上下文学习是什么？
- 高效微调的主要方法
 - ✓ 提示微调
 - ✓ LoRA、Adaptive LoRA、QLoRA
 - ✓ Adapter
- RLHF
 - ✓ 监督微调、训练奖励模型、强化学习
 - ✓ 近端策略优化(PPO)
 - ✓ 直接偏好优化(DPO)



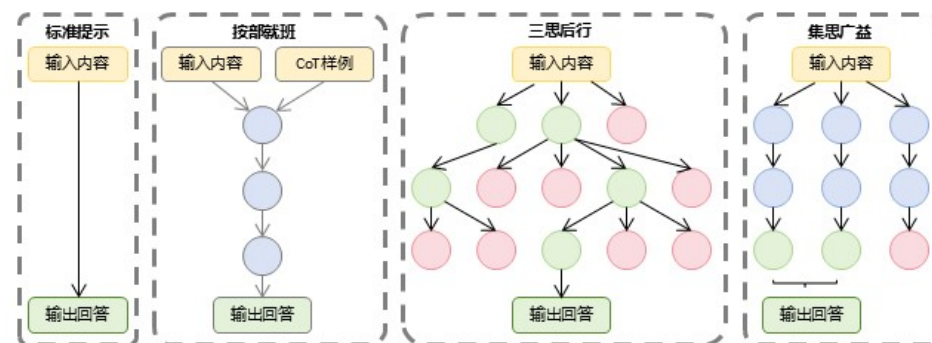
划重点时间

➤ 第七章 大模型推理与规划

- 提示工程的出现的原因及其定义
- CoT针对的是什么问题，其主要内容是什么？System1和2的区别是什么
- LLM agent的定义，及其常见的各种模块



Agent = 大模型(LLM) + 记忆 + 工具 + 规划



SYSTEM 1
Intuition & instinct

95 %
基础知识,
常识

SYSTEM 2
Rational thinking

5%
逻辑推理,
思考



划重点时间

➤ 第八章 信息检索

- 信息检索定义与分类
 - 稀疏 (布尔、TF-IDF、BM25)
 - 稠密 (ANN[基于树, 哈希, 量化, 图])
- RAG的各个部件的功能
- 向量数据库与知识图谱



➤ 第九章 人工智能应用与前沿方向

- 自然语言处理的应用有哪些?
 - NLU
 - NLG
- 计算机视觉的应用有哪些?
- 什么是多模态? 与单模态的区别是? 多模态任务有哪些?

