《多模态大模型原理与应用》

Lecture 6
思维链与价值对齐

刘阳
中山大学
人机物智能融合实验室（HCP Lab）
liuy856@mail.sysu.edu.cn

本节内容

CONTENTS

大模型的推理局限

LLMs在复杂推理任务上的挑战

⚠问题描述

现象

复杂推理任务表现差

算术、逻辑、常识推理

原因

缺乏中间推理步骤

直接生成答案，没有思考过程

解决思路

显式推理路径

让模型展示思考过程

示例：算术推理

问题

一家商店有15个苹果，早上卖出6个，下午又进货20个，现在有多少个?

直接回答（错误）

答案：41个 ✗

思维链（正确）

早上卖出后：15-6=9个
下午进货后：9+20=29个
答案：29个 ✓

什么是思维链?

Chain-of-Thought (CoT)：让模型写出思考过程

核心思想

传统方式

问题 → 答案

↓

CoT方式

问题 → 推理步骤 → 答案

★ 关键：中间步骤的外显化

经典提示词

英文

"Let's think step by step"

中文

"让我们一步步思考"

效果

在多个推理基准上显著提升性能，尤其是复杂推理任务

CoT 示例：算术推理

数学问题的思维链：分解复杂计算

示例问题

问题

一家商店有15个苹果，早上卖出6个，下午又进货20个，现在有多少个?

思维链推理

1. 早上卖出后剩余：15 - 6 = 9个
2. 下午进货后：9 + 20 = 29个
3. 答案：29个

更多示例

问题

一个长方形长8cm，宽5cm，周长是多少?

CoT

周长 = 2×(长+宽) = 2×(8+5) = 2×13 = 26cm

将复杂计算分解为简单步骤，每一步都容易验证

CoT 示例：常识推理

日常推理的思维链：结合常识知识

示例问题

问题

如果地上湿了，可能是什么原因?

思维链推理

1. 地面湿可能的原因：
   - 下雨
   - 洒水
   - 漏水
2. 需要结合更多上下文判断
3. 如果是室外且天气预报有雨 → 可能是下雨
4. 如果是室内 → 可能是洒水或漏水

答案

最终答案

可能的原因包括：下雨、洒水或漏水。具体原因需要结合场景（室内/室外）、天气、时间等信息来判断。

CoT展示了推理的完整过程，而不仅仅是给出答案

特点

常识推理需要结合背景知识，CoT帮助模型显式地组织和应用这些知识

CoT 的两种形式

Few-Shot CoT vs Zero-Shot CoT：示例vs提示词

Few-Shot CoT

方法
提供带推理步骤的示例

示例格式

```
Q: 问题1
A: 推理步骤... 答案
Q: 问题2
A: 推理步骤... 答案
Q: 测试问题
A:
```

+ 性能更好

− 需要设计示例

Zero-Shot CoT

方法
加入"让我们一步步思考"提示

提示格式

```
Q: 问题
A: 让我们一步步思考
```

+ 简单，无需示例

− 性能稍差

Few-Shot CoT 示例

带推理示例的提示：从示例学习推理模式

完整示例

示例 1

Q: 7 + 8 = ?

A: 让我们计算：7 + 8 = 15。答案是15。

示例 2

Q: 12 - 5 = ?

A: 让我们计算：12 - 5 = 7。答案是7。

测试

Q: 9 + 6 = ?

A: 让我们计算

模型从示例中学习推理模式，然后应用到新问题

Zero-Shot CoT 的魔法提示

Let's think step by step：简单提示词的强大效果

魔法提示词

"Let's think step by step"

让我们一步步思考

效果

- 无需示例
- 性能显著提升
- 在多个推理基准上有效

说明LLM已学会推理模式，只需激活

对比实验

无CoT提示

Q: 问题 A: 答案

性能：基准水平

Zero-Shot CoT

Q: 问题 A: 让我们一步步思考

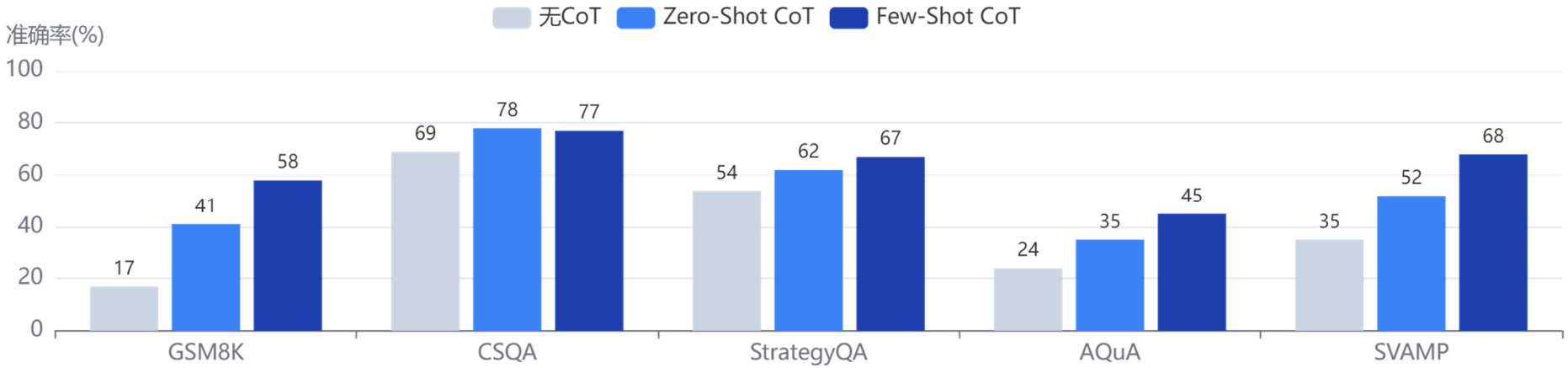
性能：显著提升

简单的提示词可以激活模型内部的推理能力，无需复杂的示例设计

CoT 的性能提升

在推理基准上的效果：显著提升推理能力

性能对比



GSM8K（数学）
17% → 58%
Few-Shot CoT

CSQA（常识）
69% → 78%
Zero-Shot CoT

StrategyQA
54% → 67%
综合提升

为什么 CoT 有效?

思维链的理论解释：分解复杂度、激活知识

分解复杂任务

将复杂问题分解为简单子任务

每个子任务更容易解决

✓ **降低认知负担**

缓冲中间结果

中间结果存储在文本中

减少工作记忆负担

✓ **避免信息丢失**

知识激活

推理步骤触发相关知识

激活预训练中的推理模式

✓ **利用预训练知识**

可解释性

人类可审核推理过程

发现错误、调试模型

✓ **提高可信度**

CoT 的局限性

思维链的不足：线性路径的限制

单一路径

无法探索多种可能

只能沿一条路径推理

错误传播

前面错了后面全错

没有纠错机制

缺乏回溯

不能修正错误

一旦走错无法回头

缺乏规划

没有全局策略

逐步推进，缺乏前瞻

需要更强大的推理方法

CoT适合简单推理，复杂问题需要探索多种可能性的方法

自洽性思维链（Self-Consistency）

多路径投票机制：采样多条推理路径

核心思想

多路径采样

生成N条不同的CoT推理（如N=40）

答案聚合

每条推理得到一个答案

多数投票

选择出现最频繁的答案

提升鲁棒性，减少单次错误

工作流程

1. 输入问题
2. 生成40条CoT推理
3. 收集40个答案
4. 多数投票 → 最终答案

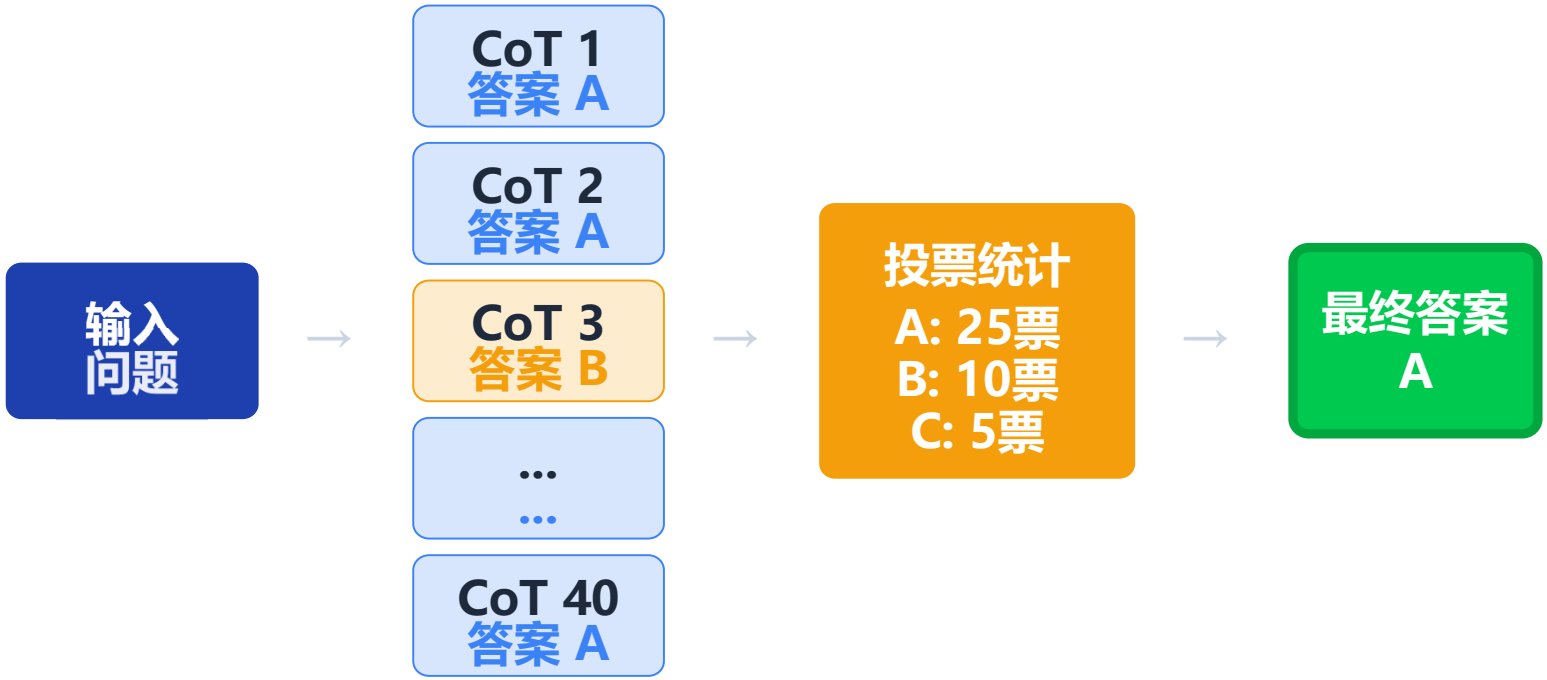
效果

GSM8K: 58% → 74%，显著提升推理准确性

Self-Consistency 示意图

多路径推理的可视化：集成思想

多路径推理过程

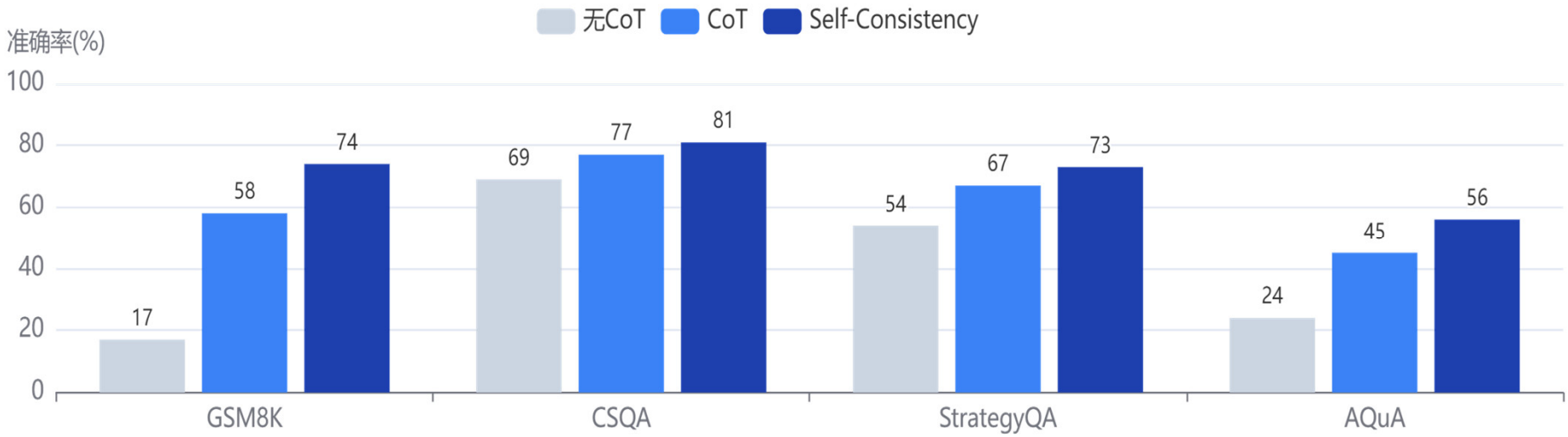


通过多条推理路径的投票，减少单次推理的错误，提高答案的可靠性

Self-Consistency 的效果

性能进一步提升：在CoT基础上再提升

性能对比



GSM8K
58% → 74%
CoT → SC

代价
N倍
推理成本增加

权衡
准确性 vs 成本
适用高价值场景

思维树（Tree of Thought）

从链到树的扩展：探索多种可能性

核心思想

CoT：线性链

1 → 2 → 3

↓

ToT：树形结构

★ 节点：中间思考状态，分支：不同推理方向

ToT的优势

- **探索多种可能**
  不再局限于单一路径
- **评估与选择**
  评估每个候选的质量
- **搜索最优路径**
  找到最佳推理路径

适用场景
复杂规划、创意写作、数学证明等需要探索多种可能性的任务

ToT 的核心组件

Tree of Thought的四个要素：分解、生成、评估、搜索

1 思维分解

将问题分解为中间步骤

每个步骤是一个思维节点

2 思维生成

每步生成多个候选思维

采样或提议生成

3 状态评估

评估每个候选的质量

用LM打分或投票

4 搜索策略

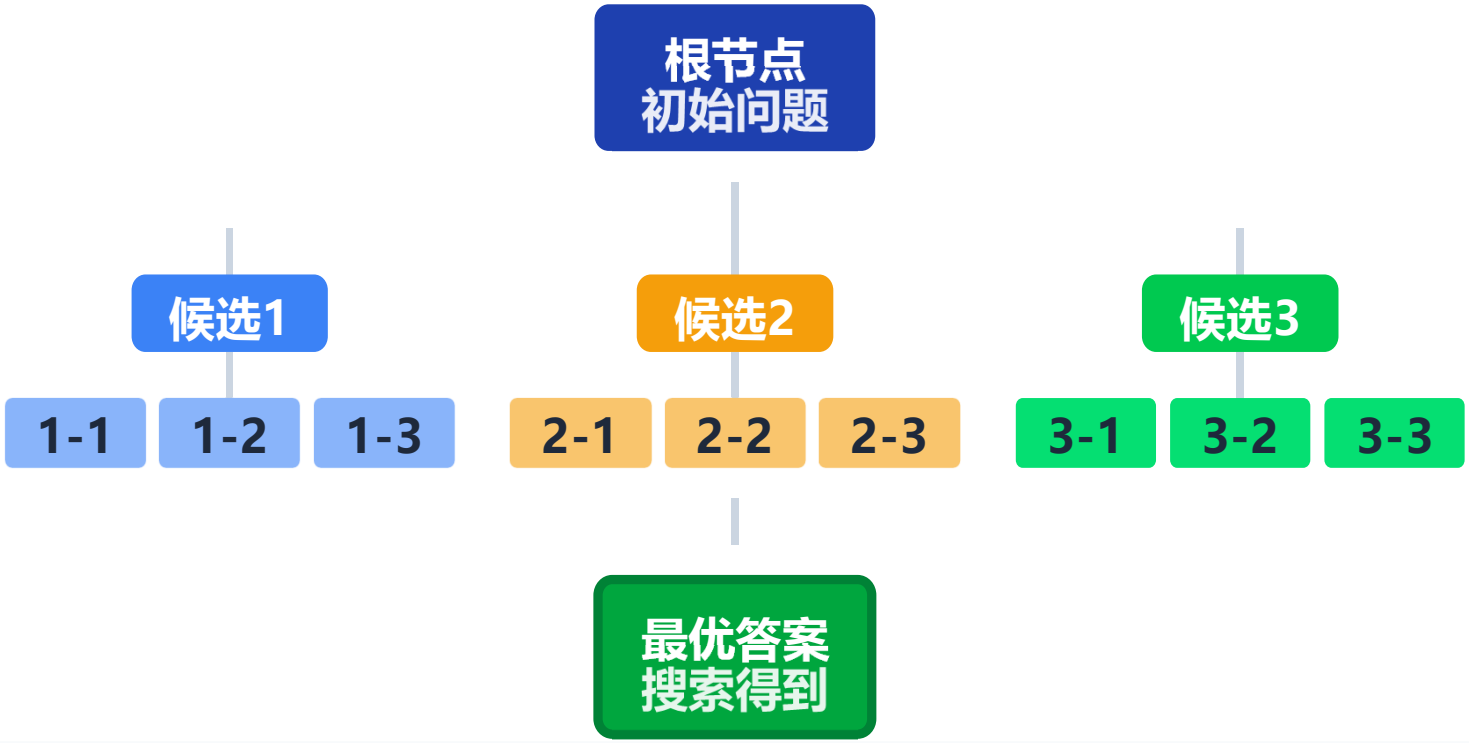
BFS、DFS或Beam Search

根据评估选择路径

ToT 示意图

树形搜索的可视化：多分支探索

树形推理结构



通过评估和搜索，在树中找到最优的推理路径

ToT 的搜索策略

如何在树上搜索：BFS vs DFS

BFS（广度优先）

策略
逐层展开，先探索所有候选

优点
不会错过好的候选

缺点
内存消耗大

适合：深度浅的问题

DFS（深度优先）

策略
深入探索，一条路走到底

优点
内存消耗小

缺点
可能陷入局部最优

适合：深度大的问题

Beam Search

策略
保留Top-K路径

优点
平衡广度和深度

最常用，效果好

启发式评估：用LM评估节点价值

ToT 的评估函数

如何评估思维状态：用LM打分

1 LM打分

让LM打分（1-10分）

"这个思路有多好？"

简单直接

2 LM比较

让LM比较两个候选

"A vs B哪个更好？"

相对判断更可靠

3 少数投票

多次采样的一致性

采样多次，看一致性

鲁棒性更好

选择建议

简单任务	复杂任务	高价值任务
LM打分	LM比较	少数投票

根据任务特点选择合适的评估方法

ToT 的优点

ToT的四大优势：通用、模块化、适应性、方便

通用性

CoT、IO、Self-Consistency都是ToT的特例

统一框架，涵盖多种方法

模块化

LM、分解、生成、评估、搜索独立可变

灵活组合，易于改进

适应性

适配不同问题、模型能力、资源限制

可调节搜索深度、分支数

方便

无需额外训练，只需预训练LM

即插即用，快速部署

ToT 的应用案例

ToT在复杂任务上的成功：解决CoT难以处理的问题

24 Game of 24

用4个数凑出24

CoT: 4% → ToT: 74%

创意写作

连贯性、创新性

生成更连贯的故事

填字游戏

Crossword Puzzles

解决复杂约束

提升幅度

Game of 24	创意写作	填字游戏	平均提升
+70%	+25%	+30%	20-30%

ToT在需要探索多种可能性的复杂任务上表现优异

思维图（Graph of Thought）

从树到图的泛化：共享子推理

核心思想

ToT：树形结构

每个节点一个父节点

A → B

↓

GoT：图结构

节点可有多个父节点

A → C ← B

C被A和B共享

GoT的优势

- **子推理可复用**
  避免重复计算
- **更灵活的结构**
  支持更复杂的依赖

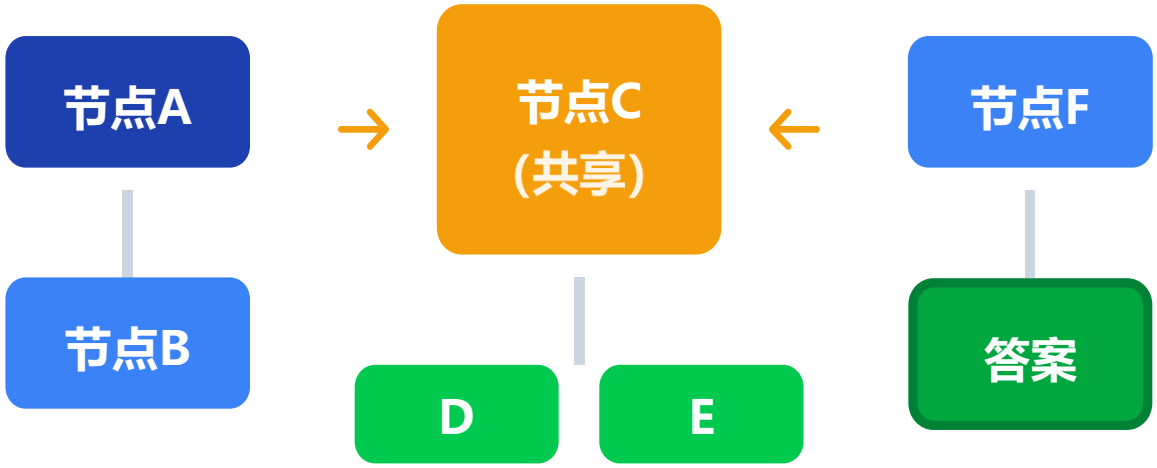
适用场景

复杂规划、知识图谱推理、多步骤任务

GoT 示意图

图形推理的可视化：节点共享

图结构推理



节点
中间推理状态

边
推理依赖关系

共享
多条路径复用

推理增强的最新进展

2024-2026的前沿方法：更强、更快、更可靠

Process Reward Models

每步打分，而不仅是最终答案

更细粒度的反馈

Tree-of-Thought Prompting

简化ToT，无需复杂实现

更易部署

Least-to-Most Prompting

从简单子问题开始，逐步增加难度

渐进式推理

Recursive Prompting

递归分解，子问题再分解

处理极复杂问题

推理增强在多模态中的应用

Visual CoT、Video CoT：图像、视频的逐步推理

Visual CoT

方法

先描述图像，再推理

示例

1. 图像中有3个苹果和2个橘子
2. 苹果比橘子多
3. 多3-2=1个

应用：视觉问答、图像理解

Video CoT

方法

逐帧理解，时序推理

示例

1. 第1帧：人在门口
2. 第5帧：人走到桌前
3. 第10帧：人拿起杯子

应用：视频理解、动作识别

挑战

计算成本

多模态数据量大

对齐质量

视觉和文本对齐困难

需要更多研究

本节内容

CONTENTS

一、思维链与推理增强

二、RLHF 与人类偏好对齐

三、RLAIF 与 AI 反馈对齐

为什么需要对齐?

从能力到安全的转变：能力≠对齐

⚠核心问题

预训练阶段

学习语言模式，不知道什么应该/不应该说

微调阶段

学习任务能力，但仍可能产生有害输出

风险

- 有害内容
- 偏见歧视
- 不诚实回答

对齐的目标

学习人类价值观

什么是对/错、好/坏

遵循人类意图

理解并执行用户指令

安全可控

拒绝有害请求

关键洞察

能力强≠安全，对齐是让AI系统可信赖的关键步骤

对齐的三大目标

Helpful、Harmless、Honest：3H原则

Helpful

有用

遵循用户指令

完成用户请求的任务

提供有价值的帮助

Harmless

无害

拒绝有害请求

不产生有害内容

保护用户安全

Honest

诚实

承认不确定性

不编造虚假信息

保持真实可信

对齐的挑战

为什么对齐困难：目标复杂、数据昂贵

价值观多样

不同文化、群体的偏好不同

难以定义统一的对齐标准

边界模糊

有害vs无害的灰色地带

难以明确界定

数据稀缺

高质量人类反馈昂贵

需要专家标注

奖励欺骗

模型可能过度优化奖励模型

找到奖励模型的漏洞

强化学习基础回顾

RL的基本概念：Agent、环境、奖励

RL组件

Agent（智能体）

策略模型（LLM）

决定采取什么行动

环境

输入提示（上下文）

模型交互的外部世界

奖励

人类偏好或奖励模型

评估行动的好坏

RL流程

1. Agent观察状态
2. Agent采取行动
3. 环境返回奖励
4. Agent学习最大化奖励

目标

最大化累积奖励，学习最优策略

RL 的数学表达

强化学习公式：最大化期望回报

核心公式

状态 s_t

当前对话状态（上下文）

动作 a_t

生成的Token

策略 π_θ(a_t|s_t)

模型生成Token的概率分布

回报与目标

奖励 r_t

r(s_t, a_t)：当前动作的即时奖励

回报 R_t

$R_t = \sum \gamma^k r_{t+k}$

未来累积奖励（折扣）

目标

$\max_\theta \mathbb{E}[R_t]$

最大化期望回报

关键

通过优化策略参数θ，使模型生成高奖励的回复

RLHF 的三个阶段

Reinforcement Learning from Human Feedback：SFT→RM→PPO

阶段1目标

让模型学会遵循指令，生成高质量回复

阶段2目标

学习人类偏好，预测人类喜欢什么

阶段3目标

用RM引导，优化模型生成策略

三阶段循序渐进，从基础能力到人类对齐

阶段 1：监督微调（SFT）

Supervised Fine-Tuning：教模型基本遵循指令

训练过程

数据

人工标注的（指令，回复）对

高质量、多样化

训练方式

标准监督学习

最大化回复的似然

目标

模型学会生成高质量回复

数据示例

指令

"解释什么是机器学习"

回复

机器学习是人工智能的一个分支，通过算法让计算机从数据中学习规律...

规模

InstructGPT：约13K标注数据

SFT 示例

指令微调的数据：高质量人工回复

SFT数据示例

指令（Prompt）

"写一首关于春天的诗"

回复（Response）

春风拂面，万物复苏，
桃花盛开，燕子归来。
大地换上新装，
生机盎然，希望满怀。

数据要求

- 高质量
- 多样性
- 安全性

标注者要求

- 专业培训
- 遵循指南
- 质量审核

阶段 2：奖励模型训练

Reward Model (RM)：学习预测人类偏好

★训练过程

数据

收集对多个回复的排序

人类标注哪个更好

模型

LM输出标量奖励

预测人类偏好分数

目标

RM(回复好) > RM(回复差)

损失函数

排序损失

$L = -\log(\sigma(r_{好} - r_{差}))$

σ：sigmoid函数

r_好：好回复的奖励

r_差：差回复的奖励

核心思想

拉大"好"与"差"回复之间的奖励差距

人类偏好数据收集

如何收集偏好数据：比较式标注

⚖比较式标注

流程

1. 给定提示
2. 生成2个回复（A、B）
3. 人类标注：A更好/B更好/相似

优点

比绝对打分更可靠

人类更擅长做相对比较

标注界面

提示

"解释量子计算"

回复A

简短回答

回复B

详细解释

标注：B更好（更详细）

规模

InstructGPT：约33K比较数据

偏好数据示例

比较式标注案例：人类判断更好的回复

偏好标注示例

提示

"写一首关于春天的诗"

回复A

春天来了，花儿开了。

简短，缺乏细节

回复B ✓

春风拂面，万物复苏，
桃花盛开，燕子归来。
大地换上新装。

详细，更有诗意

标注结果

人类标注：B更好（更详细、更有诗意）

奖励模型架构

RM的模型设计：在LM基础上加奖励头

架构设计

基座
SFT后的模型
继承语言能力

↓

修改
移除语言建模头
不再预测下一个Token

↓

新增
加入奖励头（标量输出）
输出人类偏好分数

示意图

输入（Prompt + Response）

↓

Transformer编码器

↓

奖励头 → 标量分数

训练目标
最大化偏好数据的似然，学习预测人类偏好

奖励模型的损失函数

Bradley-Terry模型：排序损失

损失函数

$$L = -\log(\sigma(r_{好} - r_{差}))$$

符号说明

- σ：sigmoid函数
- r_好：好回复的奖励
- r_差：差回复的奖励

目标

拉大"好"与"差"回复的奖励差距

直观理解

当 r_好 > r_差

σ(r_好 - r_差) → 1

损失小 ✓

当 r_好 < r_差

σ(r_好 - r_差) → 0

损失大 ✗

核心思想

鼓励模型给好回复打高分，给差回复打低分

阶段 3：PPO 强化学习

Proximal Policy Optimization：用RM引导策略优化

PPO组件

策略模型

从SFT模型初始化

需要优化的模型

奖励

RM打分

指导模型优化方向

约束

KL散度惩罚

不偏离SFT模型太远

PPO特点

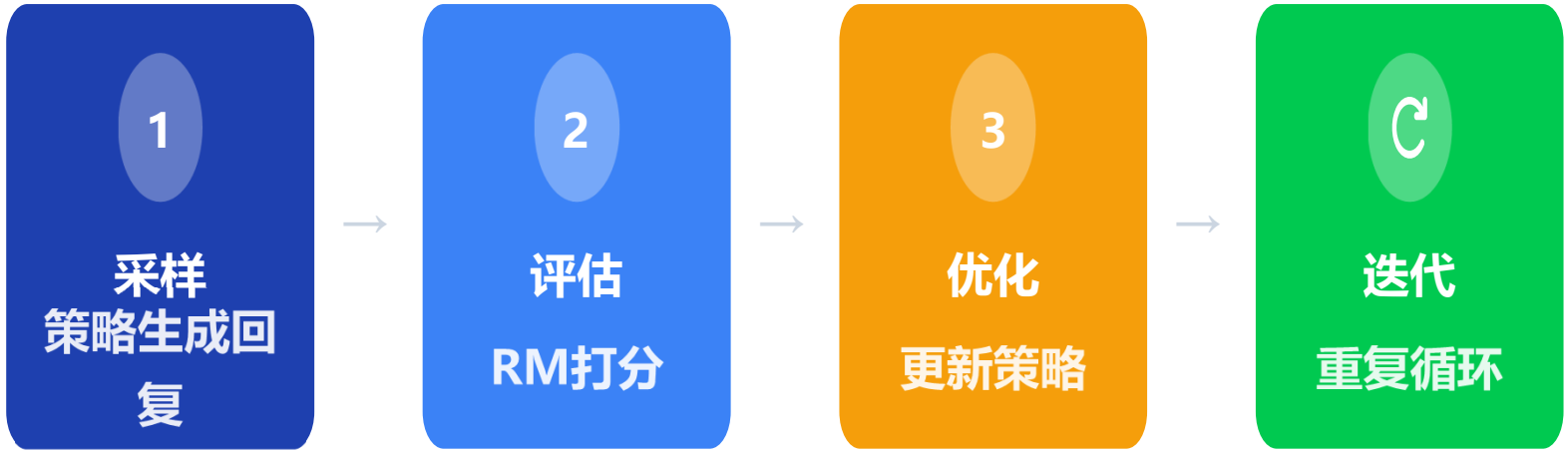
- **稳定**
  避免策略剧烈变化
- **高效**
  样本利用率高
- **常用**
  RLHF的标准算法

核心思想

用RM作为奖励信号，引导策略模型生成人类更喜欢的回复

PPO 的工作流程

RLHF的训练循环：采样、评估、优化



步骤1：采样
策略模型根据提示生成多个候选回复

步骤2：评估
奖励模型给每个回复打分

步骤3：优化
计算PPO目标，更新策略参数

步骤4：迭代
重复多轮直到收敛

通过不断迭代，策略模型逐渐学会生成高奖励（人类偏好）的回复

PPO 目标函数

带约束的策略优化：平衡奖励与稳定性

目标函数

$$\max_\theta (\text{奖励} - \text{KL惩罚})$$

奖励项

$\mathbb{E}[r(y)]$

最大化RM奖励

KL惩罚项

$-\beta \cdot KL(\pi_\theta || \pi_{ref})$

不偏离参考策略

组件说明

π_θ
当前策略（待优化）

π_{ref}
参考策略（SFT模型）

β
KL惩罚系数（超参数）

平衡

奖励提升 vs 稳定性，避免模型过度优化

KL 散度约束的作用

为什么需要KL约束：防止模式崩溃

✗ 无约束的问题

过度优化

模型可能找到RM的漏洞

结果

- 生成异常文本
- 重复无意义内容
- 失去语言能力

模式崩溃：为获得高奖励而牺牲质量

✓ 有约束的好处

保持相似性

与SFT模型保持相似

平衡

- 奖励提升
- 语言质量保持

可控优化：渐进式改进

RLHF 示意图

三阶段的完整流程：从SFT到对齐模型

RLHF完整流程

阶段1
学习基本能力

阶段2
学习人类偏好

阶段3
优化对齐

RLHF 的效果

InstructGPT的性能提升：偏好对齐显著改善

有用性

85%

人类偏好InstructGPT

相比GPT-3显著提升

真实性

-20%

幻觉减少

更诚实的回答

有害性

-50%

有害输出减少

更安全的回复

规模效应

1.3B > 175B

对齐模型优于无对齐大模型

对齐比规模更重要

RLHF显著提升模型的有用性、真实性和安全性，对齐比单纯扩大规模更重要

RLHF 的挑战

当前存在的问题：数据、奖励、多样性

数据昂贵

人类标注成本高

需要专业标注者

奖励欺骗

模型找到RM漏洞

过度优化奖励

多样性损失

过度对齐导致保守

输出多样性下降

分布外泛化

RM在新分布上失效

泛化能力有限

奖励欺骗（Reward Hacking）

模型欺骗奖励模型：过度优化的副作用

⚠现象

模型找到RM的偏好模式

通过分析RM的打分规律，生成能获得高奖励但实际质量低的回复

典型表现

- 过度使用客套话
- 冗长啰嗦
- 重复安全短语

原因

RM不完美，存在偏见和漏洞

缓解方法

- **多样性惩罚**
  鼓励多样化输出
- **定期更新RM**
  修复RM漏洞
- **KL约束**
  限制偏离程度

关键

RM只是人类偏好的近似，需要谨慎使用

RLHF 的最新改进

2024-2026的优化方向：更高效、更稳定

DPO

Direct Preference Optimization

无需RM，直接用偏好数据优化

PPO改进

多阶段训练、自适应KL

更稳定的训练过程

Constitutional AI

自我批评与改进

减少人工标注

分布式RLHF

大规模并行训练

加速训练过程

多模态 RLHF

视觉-语言模型的对齐：跨模态偏好学习

多模态对齐

偏好维度

- 相关性：图文是否相关
- 准确性：描述是否准确
- 安全性：是否包含有害内容

数据收集

图文对 + 人类偏好排序

奖励模型

跨模态RM，处理图文输入

挑战

图像有害内容检测
需要专业审核

跨模态对齐
图文语义对齐困难

数据稀缺
多模态偏好数据少

应用

DALL-E、Midjourney等图像生成模型的对齐

具身智能中的 RLHF

机器人对齐：动作策略的人类偏好

应用场景

场景

- 机器人抓取
- 导航
- 操作

偏好维度

- 安全：不伤害人类
- 高效：快速完成任务
- 自然：动作流畅

数据

人类演示 + 偏好比较

案例：RT-H

Robotics Transformer with Human Feedback

用RLHF优化机器人动作策略

人类偏好引导机器人学习更自然、安全的动作

挑战

实时反馈、安全性要求、数据收集困难

本节内容

CONTENTS

一、思维链与推理增强

二、RLHF 与人类偏好对齐

三、RLAIF 与 AI 反馈对齐

RLHF 的瓶颈：人类标注

Human Feedback的局限：规模化困难

成本高

专家标注昂贵

需要专业培训

速度慢

标注周期长

难以快速迭代

不一致

标注者间分歧

主观性强

覆盖有限

难以覆盖所有场景

长尾问题难处理

RLAIF 的提出

Reinforcement Learning from AI Feedback：用AI替代人类标注

核心思想

用AI提供反馈

用强大的AI（如GPT-4）替代人类标注

优势

- 规模化：可大规模生成
- 快速：无需人工等待
- 一致性：AI判断更一致

挑战

- 质量依赖教师模型
- 可能传播AI偏见

适用场景

低风险场景
一般性任务对齐

初步对齐
快速迭代原型

资源受限
无法承担人工标注

关键

AI反馈不是人类反馈的完全替代，而是补充

RLAIF 的工作流程

AI Feedback的生成：用LM作为裁判

步骤1
策略模型生成多个候选回复

步骤2
用强LM（如GPT-4）评估和排序

步骤3
构造AI偏好数据集

步骤4
训练RM，PPO优化策略

RLAIF与RLHF流程相同，只是反馈来源从人类变为AI

Constitutional AI (CAI)

Anthropic的对齐方法：用宪法原则自我改进

⚖核心思想

宪法原则

定义一套人类价值观原则
明确的行为准则

自我改进过程

1. 模型生成回复
2. 自我批评（找问题）
3. 自我修正（改进）

反馈来源

AI根据宪法原则评估

优势

- **减少人工标注**
  AI自我评估
- **可扩展**
  大规模自我改进
- **价值观明确**
  宪法原则清晰

代表模型
Claude系列（Anthropic）

Constitutional AI 示意图

CAI的完整流程：批评-修正循环

CAI流程

多轮批评-修正循环，直到回复符合宪法原则

宪法原则示例

CAI的价值观定义：明确的行为准则

示例原则

原则1
避免帮助非法活动

原则2
避免歧视性内容

原则3
尊重隐私

原则4
承认不确定性

原则5
拒绝有害请求时给出理由

RLAIF vs RLHF

两种方法的对比：互补而非替代

RLHF

反馈来源
人类

优点
质量高（专家）

缺点
成本高、规模有限

适用：高风险任务

RLAIF

反馈来源
AI模型

优点
成本低、大规模

缺点
质量依赖教师模型

适用：低风险、初步对齐

互补关系
RLHF和RLAIF不是替代关系，而是互补。实际应用中常采用混合方式

RLAIF 的最新进展

2024-2026的研究：混合反馈成为趋势

混合方法

人类+AI反馈结合

取长补短

主动学习

AI标注，人类审核不确定样本

提高效率

自我对齐

模型自我评估和改进

减少外部依赖

多模型一致性

多个AI投票

提高可靠性

RLAIF 的应用场景

哪里可以用RLAIF：规模优先的场景

开源模型对齐

LLaMA、Mistral等

低成本快速对齐

领域适配

法律、医疗等专业领域

专业反馈难获取

多语言对齐

资源稀缺语言

人工标注难获取

快速迭代

产品原型阶段

快速验证想法

RLAIF适合对成本敏感、需要快速迭代的场景，但高风险场景仍需要人类反馈

问题和讨论