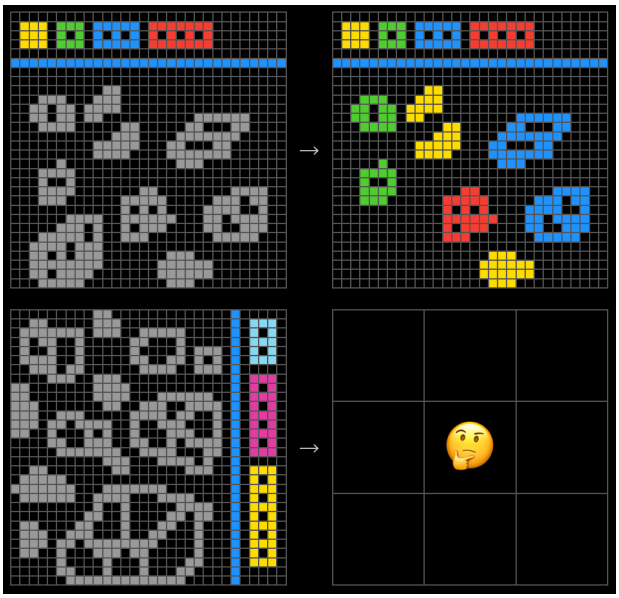
提示词工程考试说明（2025年秋）

背景

在人工智能快速发展的浪潮中，大语言模型（LLM）已成为推动技术变革的核心力量。LLM不仅在自然语言处理任务上取得了SOTA表现，其强大的上下文学习（In-context Learning）和零/少样本（Zero/Few-shot）学习能力也为解决复杂的逻辑推理问题开辟了新的途径。提示词工程（Prompt Engineering）作为驾驭LLM能力的关键技术，已从简单的问答交互演变为一门涉及指令设计、示例组织、思维链（Chain-of-Thought）引导和任务分解的专业学科。通过精心设计的提示词，我们能引导LLM挖掘数据背后的深层逻辑，完成传统编程范式难以解决的抽象任务。



一个ARC-AGI评测基准的示例

抽象与推理语料库（Abstraction and Reasoning Corpus, ARC） 是一个广受关注的AGI评测基准。它旨在衡量智能系统（包括人类和AI）的“流体智力”，即在面对全新问题时，仅通过少量示例（Few-shot Examples）快速学习、抽象出背后规则并解决问题的能力。ARC任务通常表现为一系列输入-输出网格（grid）变换，要求解题者找出隐藏的变换逻辑。由于其高度的抽象性和对泛化能力的极致要求，ARC至今仍是AI领域的一大挑战。

考试内容

本次考核聚焦于利用提示词工程（Prompt Engineering）使大语言模型解决ARC逻辑推理任务。考生将获得一个验证集文件（附件：`val.jsonl`），该文件中每行代表一个独立的ARC任务。每个任务包含：

1. **训练样本（Train Examples）**：几组（通常为2-3组）“输入网格（input）”到“输出网格（output）”的变换示例，用于展示该任务隐藏的抽象规则。

2. **测试样本（Test Example）**：一个“输入网格（input）”，考生需要引导LLM基于从训练样本中学到的规则，推理出这个测试输入对应的“输出网格（output）”。

`val.jsonl` 中，每一条数据的字段说明：

代码块

```
{
  "train": [ // 训练样本列表，用于展示变换规则
    {
      "input": [ // 训练样本1的输入网格（二维数组）
        [0, 0, 0, 0],
        [0, 5, 6, 0],
        [0, 8, 3, 0],
        [0, 0, 0, 0]
      ],
      "output": [ // 训练样本1的输出网格（二维数组）
        [5, 0, 0, 6],
        [0, 0, 0, 0],
        [0, 0, 0, 0],
        [8, 0, 0, 3]
      ]
    },
    {
      "input": [ // 训练样本2的输入网格
        [0, 0, 0, 0],
        [0, 3, 4, 0],
        [0, 7, 6, 0],
        [0, 0, 0, 0]
      ],
      "output": [ // 训练样本2的输出网格
        [3, 0, 0, 4],
        [0, 0, 0, 0],
        [0, 0, 0, 0],
        [7, 0, 0, 6]
      ]
    }
  ],
  "test": [ // 测试样本列表（本次任务中固定只有一个）
    {
      "input": [ // 需要预测的测试输入网格
        [0, 0, 0, 0],
        [0, 2, 3, 0],
        [0, 4, 9, 0],
        [0, 0, 0, 0]
      ],
      "output": [ // 对应的正确答案（Groud Truth），考生在构造prompt时不可见
        [2, 0, 0, 3],
```

```
          [0, 0, 0, 0],
          [0, 0, 0, 0],
          [4, 0, 0, 9]
        ]
      }
    ]
  }
```

考生需要通过提示词工程，设计合适的提示词（prompt），使大语言模型能够基于 `train` 中的示例，正确预测 `test[0]["input"]` 对应的 `test[0]["output"]`。

评价模型推理性能的指标为**准确率（Accuracy）**，即**完全匹配（Exact Match）**的比例。

具体来说，给定一个模型预测的输出网格 P (一个二维列表) 和真实的答案网格 G (一个二维列表)，我们按如下方式计算“完全匹配”：

- 对于所有 i, j， P 和 G 中的每一个对应元素必须完全相等（`P[i][j] == G[i][j]`）。

只有满足上述条件时，才记为一次“完全匹配”，得分为1；否则得分为0。

考核方式

考生需要使用OpenAI API接口形式（https://platform.openai.com/docs/guides/text?api-mode=chat）构造prompt，并完成以下两个函数：

函数一：构造提示词

代码块

```python
def construct_prompt(d):
    """
    构造用于大语言模型的提示词

    参数:
    d (dict): jsonl数据文件的一行，解析成字典后的变量。
              注意：传入的 'd' 已经过处理，其 'test' 字段列表
              只包含 'input'，不包含 'output' 答案。

    返回:
    list: OpenAI API的message格式列表，允许设计多轮对话式的prompt
    示例: [{"role": "system", "content": "系统提示内容"},
           {"role": "user", "content": "用户提示内容"}]
    """

    # 实现提示词构造逻辑
    return []
```

函数二：解析输出

```
代码块
1  def parse_output(text):
2      """
3      解析大语言模型的输出文本，提取预测的网格
4
5      参数:
6      text (str): 大语言模型在设计prompt下的输出文本
7
8      返回:
9      list: 从输出文本解析出的数组 (Python列表，元素为整数)
10     示例: [[0, 1, 2], [3, 4, 5], [6, 7, 8]]
11
12     """
13     # 实现输出解析逻辑
14     return []
```

友情提示：

- 一些平台如DeepSeek，火山引擎，硅基流动等注册会有部分免费的api额度，且此考核项目已保证在迭代开发过程中总体对api消耗金额需求较少
- 不限制使用任何外部的AI工具辅助该考试的完成；同时欢迎尝试使用创智学院自研的：
  - AI辅助Research平台（包含普通问答和深度研究功能）：https://www.opensii.ai/

提交要求

考生需要提交：

1. **Python文件**：（模版文件：template.py）

- 文件中有且仅能包含上述两个函数
- 不允许import第三方库，仅可import Python标准库（如random, re, json等）
- 禁止出现直接编码公开的相应测试集，并采用穷举法搜索官方正确答案（测试集也不会采用公开benchmark中的现成样例）

2. **探索报告**：pdf文件，简易记录探索过程，包含不同提示词策略的尝试、效果和分析，作为主观分数的参考之一

提交方式：（截止时间 北京时间 10月23号 23:59 ，期间可以多次提交，会自动覆盖先前的提交文件）

1. 进入链接：https://send2me.cn/OE3s6acL/QVubIC9-c_LT2A

2. 精确填写个人信息，包括报名号、姓名（由填写错误导致的得分缺失，后果自负！）

评分标准

本项考核的分数由两部分组成：

1. **推荐性能客观得分**（占总分80%）
   - 所有考生在私有测试集（test.jsonl，格式与val.jsonl完全相同，但是考生不可见，**且无法直接读取其中的答案部分字段**，且私有测试集中包含难度更大的问题，且不存在于公开的benchmark测试集中）上的平均准确率性能进行排名并赋分
   - 测试统一使用DeepSeek-V3.2**非思考模式**模型（使用官方api测试，即https://api-docs.deepseek.com/zh-cn/中model="deepseek-chat"），temperature=1.0, max_tokens=8k, 其他参数均按照官方文档的默认值。考生需要自己编写调用大模型、评估分数的主运行函数进行开发，实际后台跑分时的运行逻辑为：

     代码块

     ```
     # 后台评测如何引入考生提交的函数

     # 1）考生提交
     your_submission/
     ├─ template.py   # 你实现的两个函数：construct_prompt、parse_output（仅此两个函数）
     └─ report.pdf    # 探索报告（作为主观评分依据之一，不参与调用）

     # 2）评测后台
     eval_runner/
     ├─ run.py        # 评测主程序（固定），会import考生实现的两个函数
     ├─ template.py   # 考生实现的两个函数
     └─ data/
        ├─ val.jsonl    # 公开验证集（仅用于你本地自测，不计分）
        └─ test.jsonl   # 私有测试集（评分用，答案不对外）
     ```

   - 每个样本会进行多次采样取平均指标以保证结果稳定性
2. **提示词主观评价得分**（占总分20%）
   - 由专家老师基于指定评价准则进行评分
   - 评价内容包括提示词的创新性、合理性、可解释性等