

走进4.0



走进 4.0

徐靖
2024/10/19

一个凸优化问题

maximize GPA(college, major, learning method)

subject to $\|\text{time}\|_1 = 24 \times 365 \times 3$

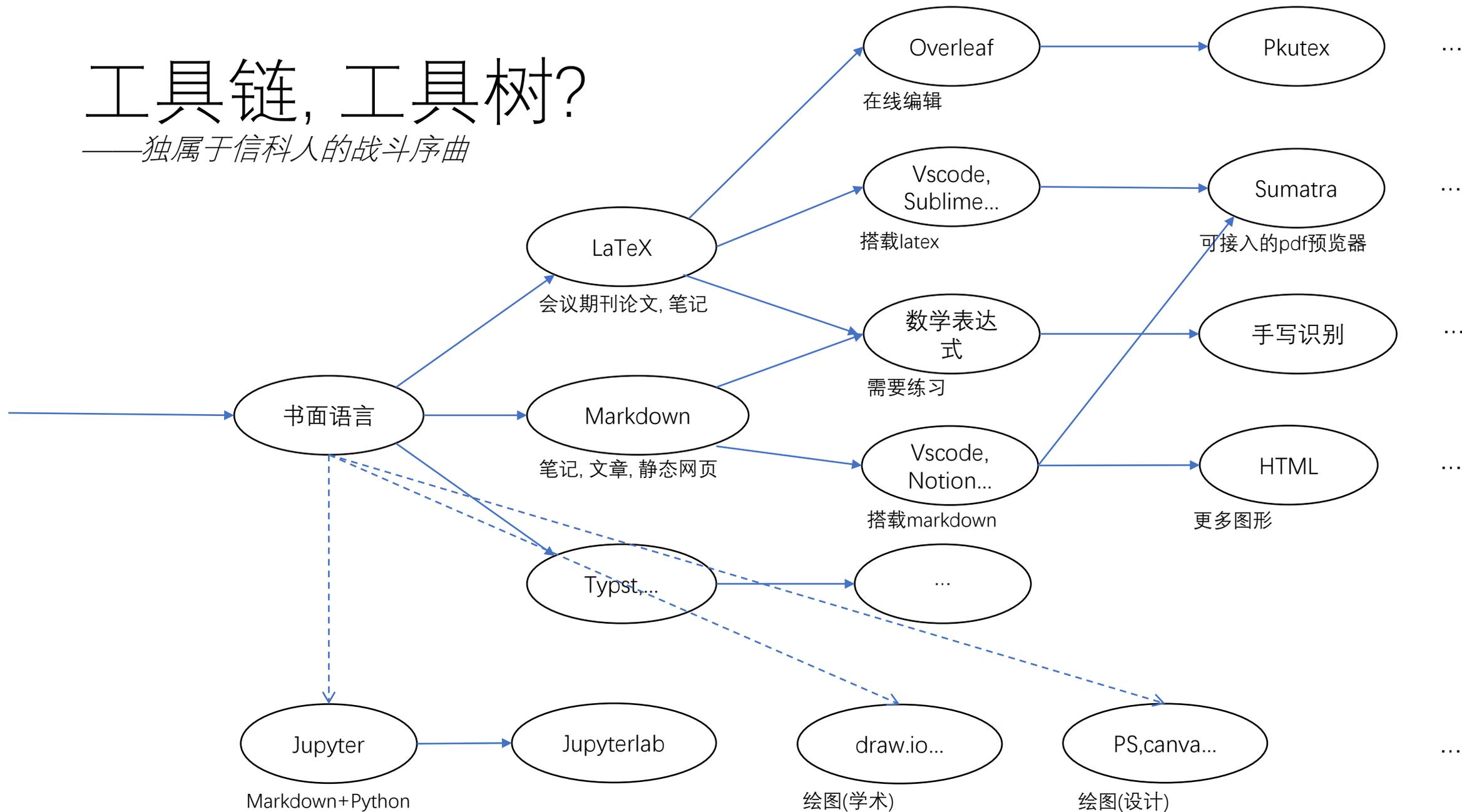
- $\frac{\partial \text{GPA}}{\partial \text{learning method}}$
- $\frac{\partial \text{GPA}}{\partial \text{college}} \Big|_{pku}$
- $\frac{\partial \text{GPA}}{\partial \text{major}} \Big|_{xk(cs)}$

$$\frac{\partial \text{GPA}}{\partial \text{major}} \Big|_{xk(\text{cs})}$$

技术改变(学习)生活

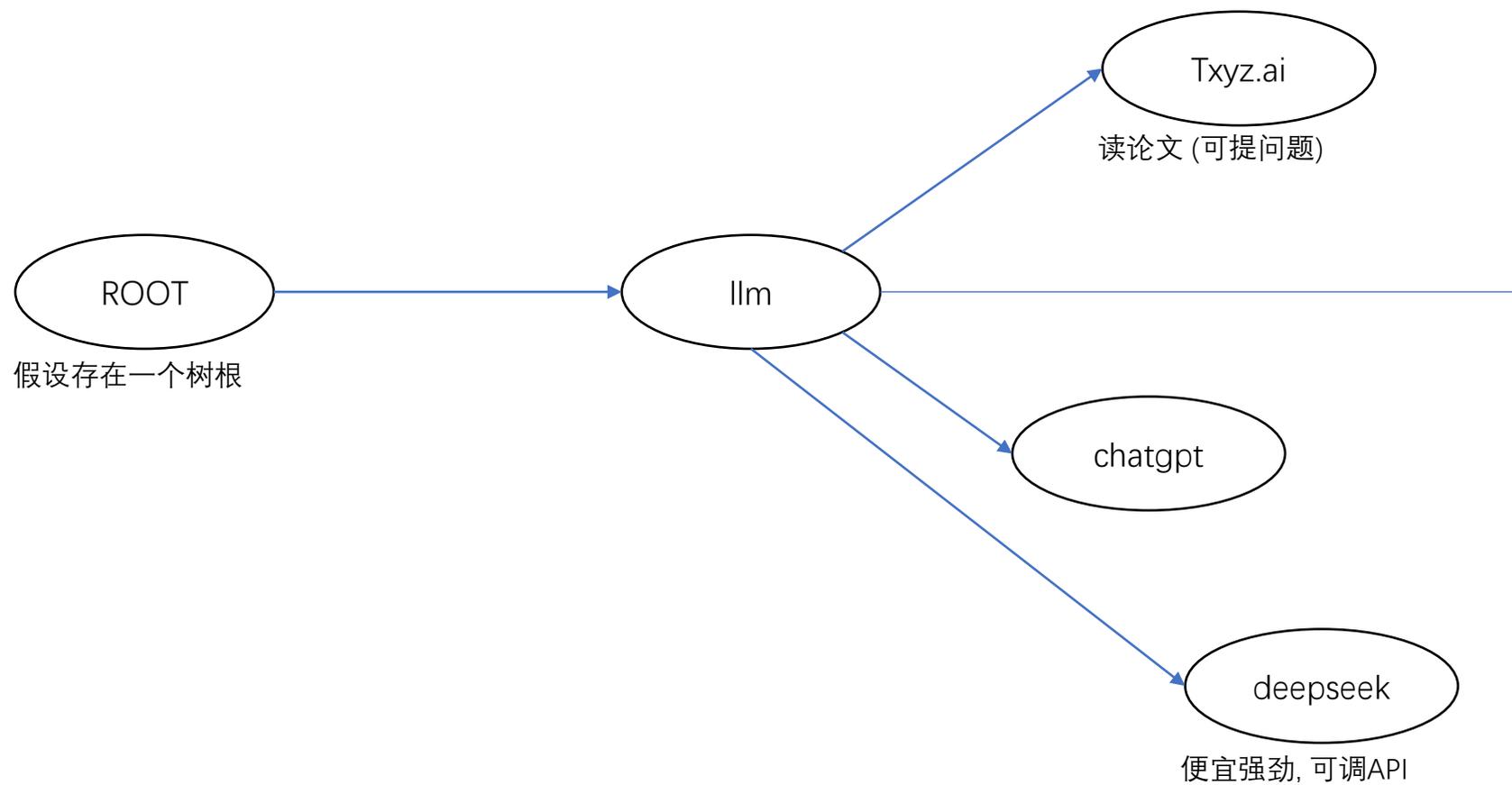
工具链, 工具树?

——独属于信科人的战斗序曲



大语言模型

——最接近工具树根的工具



实战

——如果你遇到以下情形, 你的方案是?

2023秋, 你选修了<离散数学与结构(1)>, 你发现ltr老师十分变态, 这门课横跨数理逻辑, 抽象代数, 概率统计, 图论, 数论, 随机过程, 信息论, 傅里叶分析. 为了应对期末考试, 你需要准备一份4页纸的cheatpaper.(这在课程允许范围内)

- 方案1 找学长要一份

调查课程的往年信息, 你发现ltr老师是第一年教这门课, 课程难度也远高于往届, 没有先例供参考

- 方案2
 - 使用北大买过了的adobe pdf集成作业题, 一页可以放下 $2*3*3=18$ 面pdf
 - 使用word调页边距和字体, 反复实验找到了适配本人视力的模版, 一页可以装3w字
 - 由于你擅长公式编辑, 两天能码完笔记, 时间成本可以接受

个人主页

——信科人要学会包装自己

- 为什么需要个人主页?
 - 装b 展示代码和项目
 - 简历, 应对面试
 - 写文章
- 样例
 - [程序员、技术作者Paul Graham](#)
 - [Web设计师Jeffrey Zeldman](#)
 - [哈佛大学讲师David Malan](#)
 - [似乎很多同学在用的一个好看模版](#)
- 如何配置
 - <https://github.com/ICUlizhi/ICUlizhi.github.io>



图：当信科人学会包装自己

$$\frac{\partial \text{GPA}}{\partial \text{college}} \Big|_{pku}$$

信息与信息壁垒

打破信息壁垒

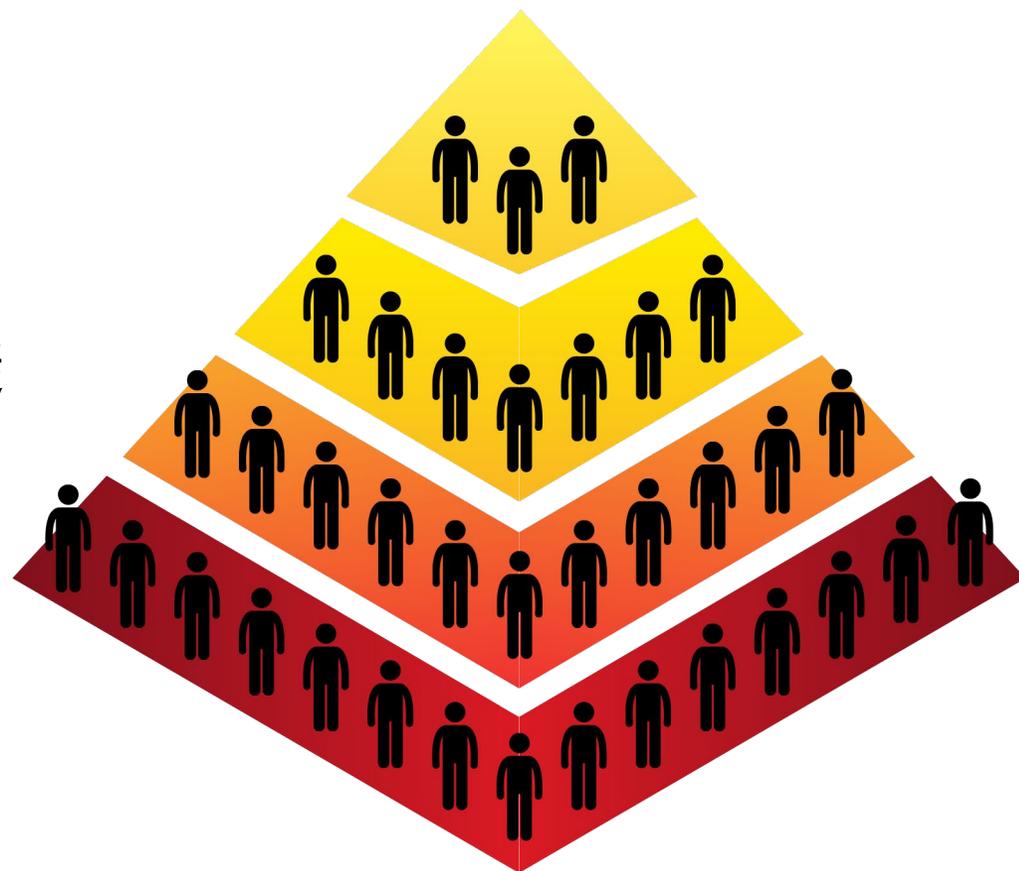
——信息论 \times

- 人无法度量未知
- 人无法避免被已知所影响
- 不受困扰的方法
 - 多和学长姐沟通
 - 对已知问题做全方位, 多角度的调研
 - 学会合作, 展现自我价值并交换信息
 - 成为信息的中转或创造者
- 为什么产生信息壁垒?
 - 假设有10w人都学高数, 没学定义为水平0, 1w个人学了定义为1, (1k, 2), (100, 3), (10, 4), (1, 5)
 - 最后一个大神开创了某种做北大高数题稳定拿满的绝世功法A, 花费时间为 t_1
 - 水平为2的某个人找到速解导数的方法B, 发到了b站上, 得到1w个人点赞, 学会B需要 t_2
 - B被公开, 大神持有A构成了对其他人的信息壁垒
 - 假设北大对这10w个人招生, 只招100个人, 于是出现了 $A_1, A_2, A_3, \dots, A_n$.

成为信息的创造者

——给xk以我, 而非给我以xk

- “学习”是一个高维主体, 非某种单一指标 (如数学成绩, 政治课成绩) 能衡量
 - 学习结果不再是类金字塔分布
- 在pku和eecs学习, 方法论的差异将不可避免地增大
- 信息壁垒的维护成本进一步降低



Prompt水平哪家强

——用自然语言处理处理自然语言课程

- B级暑校英语课,老师极度反对用ai学习
- 任务量极大,每天读4-6个小时名著
- 本项目最终帮助同学们在日均时间成本1-2h内取得90+的总评

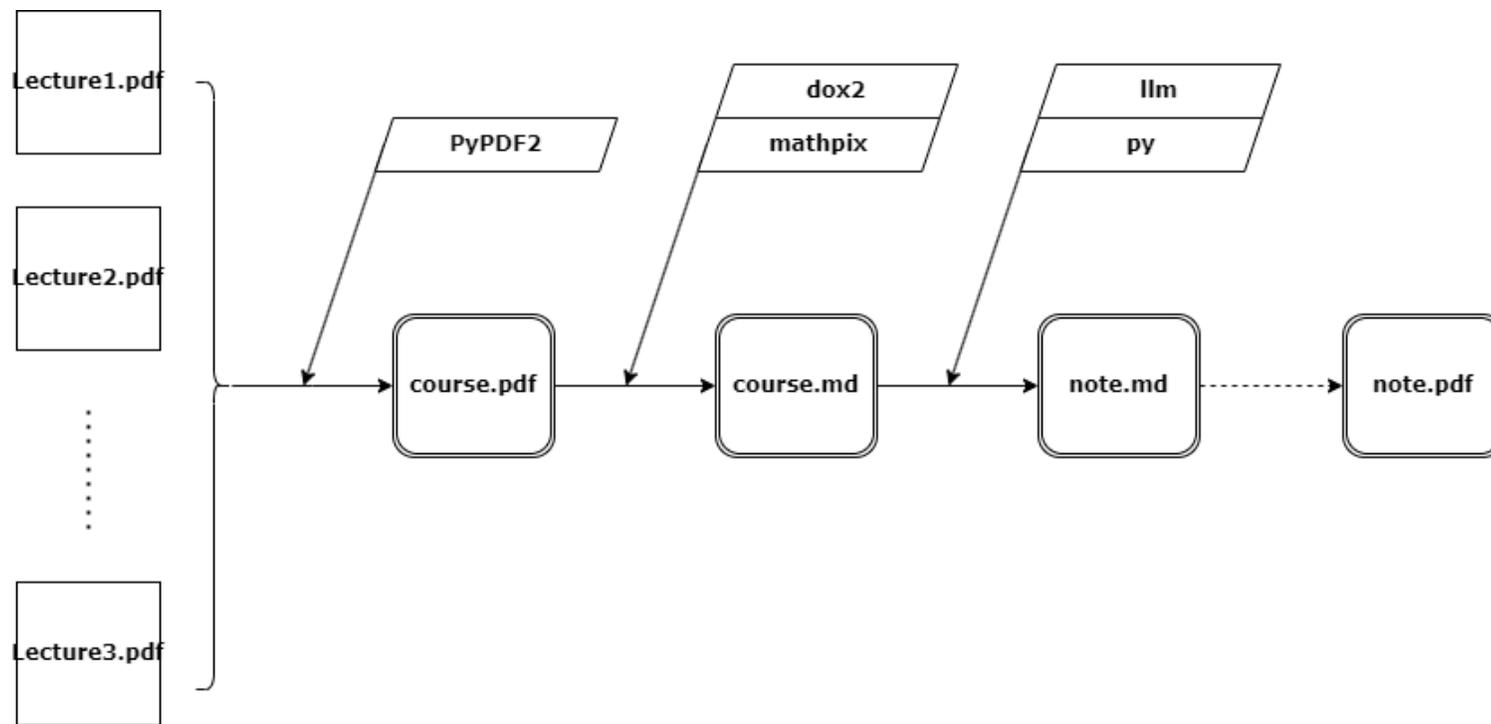
Task	My work
词汇总结	llm从文本中提取
词义猜测	ai生成
想法与问题	这个还是得自己写
quiz准备	ai生成和组织复习资料

```
def get_gpt_responses(chunks, sample, prompt0):
    responses = []
    i = 0
    for chunk in chunks:
        prompt = chunk + prompt0
        response = client.chat.completions.create(
            model="deepseek-chat",
            messages=[
                {"role": "system", "content": sample["system"]},
                {"role": "user", "content": sample["user"]},
                {"role": "assistant", "content": sample["assistant"]},
                {"role": "user", "content": prompt}
            ],
            stream=False,
            temperature=1.0
        )
        responses.append(response.choices[0].message.content.strip())
        i += 1
        print(f"Chunk {i} of {len(chunks)} processed.")
    return responses
```

- <https://github.com/ICUlizhi/PKU-2024-IBAL-cracked-version>

懒人必备：一键生成笔记

——pdf2note(注: 没做完)



- <https://github.com/ICUlizhi/pdf2note>

谢谢!

主页 <https://iculizhi.github.io/>

微信



徐靖

中国大陆 北京



扫一扫上面的二维码图案，加我为朋友。

徐靖
2024/10/19