

连享会 · Open 2025

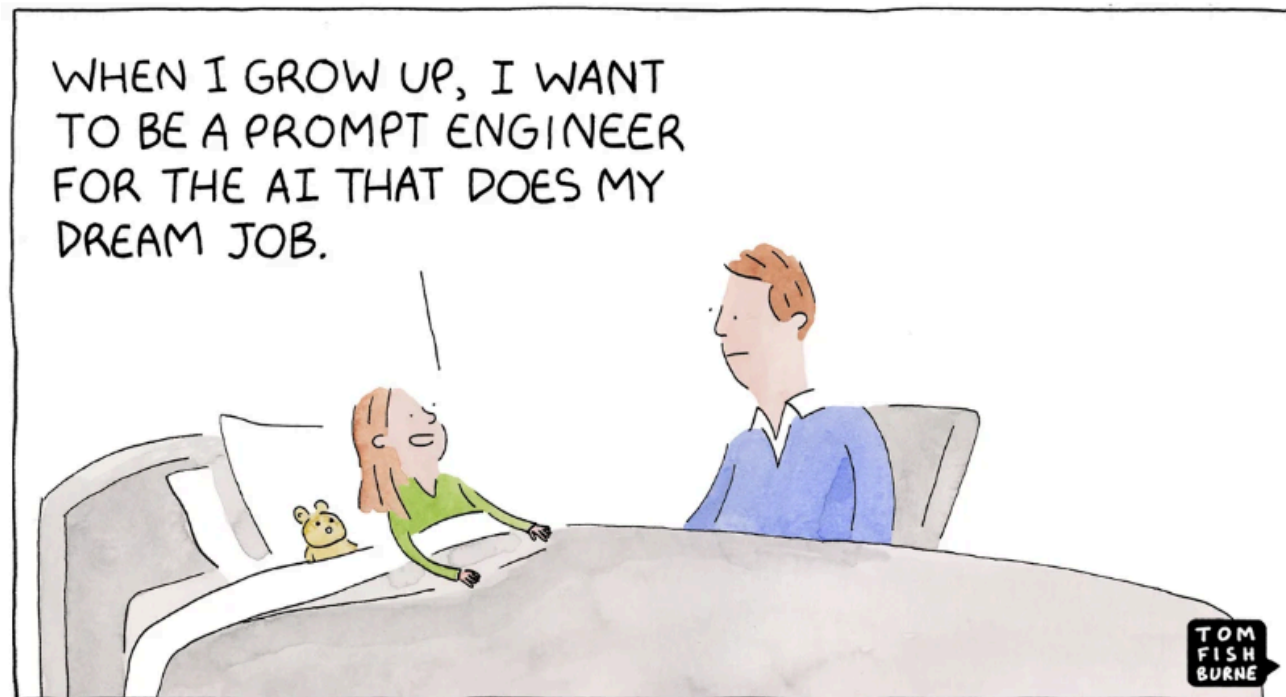
[https://lianxhcn.github.io/research\\_with\\_AI](https://lianxhcn.github.io/research_with_AI)

## Empirical Research with AI

连玉君 (中山大学)

[arlionn@163.com](mailto:arlionn@163.com)

🍏 课件: [www.lianxh.cn](http://www.lianxh.cn) → [公开课]



- AI 时代：实证研究模式如何变化？
- AI 工具如何提升实证研究的效率？
- AI 工具如何帮助我们更好地思考和解决问题？
- 实例：
  - 如何用 AI 工具发现好的 IV？
  - 如何用 AI 理解复杂的计量文献？
- 技术：
  - Stata, R 还是 Python？
  - Markdown, Github 和 Jupyter Notebook
  - AI 助力论文复现

# 1. AI 时代：实证研究模式如何变化？

## 引子：你是干什么的？

一位窗帘商人，占据了大部分市场，想再多赚点，便请来一位智者 .....

- 商人：“我该如何提升利润？”
- 智者：“你是做什么的？”
- 商人：“我做漂亮的布艺窗帘。”

智者笑而不语，抄着手离开了。过了一段时间，商人又请来智者，

- 商人：“我该如何提升利润？”
- 智者 沉思片刻，问道：“你是做什么的？”

又隔了一段时间，商人还没有找到答案，便再次请来智者，

智者拉动着漂亮的窗帘，屋内的光线忽明忽暗，商人有些茫然 .....

- 智者：“你是做什么的？”

商人恍然大悟！

 ChatGPT 对话过程

## 思考

- 我为什么要学 Stata / Python / R ?
- 我是做什么的?

## 变与不变

- 实证研究和学术研究的本质都没有变
  - 大胆假设，小心求证
  - 多数情况下，「问题」比「方法」更重要
  - 提出问题的过程，就是「思考」的过程，就是不断「假设-逻辑推理(测试)-证伪」的过程
- 研究假设的生成过程变化了
  - AI+：更快地生成假设：提供多种思考视角
  - AI+：更好地理解已有成果：哪里能去，哪里不能去
- 研究方法和工具的变化
  - 写代码 → 写提示词：数据清洗、研究设计、模型构建
  - 从「单点技能」转向「跨界整合」
  - **短板效应**：增强 v.s. 减弱？

## 利用 AI 精简研究：逐步指南



Made with Napkin

## 过去的学习中心是什么？

- **经典理论与范式：**微观、宏观、计量三大支柱；
- **数学与统计工具：**线性代数、概率论、数理统计、计量经济学；
- **主流软件技能：**Stata、R、Matlab、SAS；
- **文献阅读与论文写作能力：**跟踪一条 literature line，围绕它选题、建模、实证。

## AI 时代的变化在哪里？

- 从「单点技能」转向「跨界整合」
  - 整合数据、模型、算法、文献，并用 AI 协助迁移
- 理论与数据的互动方向反转
  - 从「理论先行 → 实证验证」向「数据先感知 → 模型解释」过渡
- 工具从模型核心转向问题核心
  - 研究工具 = 解决问题的桥梁，而非目的本身
  - 从「写代码」转向「写提示词」：重点不是命令语法，而是能否清晰表达问题、设计流程、定位输出
  - 工具：[ChatGPT](#)、[Causal.Claims](#)、[DeepSeek](#)、[PaperQA](#)



## AI 工具如何提升实证研究的效率？

- Korinek, A. (2023). Generative AI for Economic Research: Use Cases and Implications for Economists. Journal of Economic Literature, 61(4), 1281–1317.

[Link](#) (rep), [PDF](#), [-PDF2-](#)

[Appendix](#), [Google](#).

[作者主页-Tips-AI](#)

TABLE 4  
SUMMARY OF LLM CAPABILITIES AND RATING OF USEFULNESS

Category	Task	Usefulness
Ideation & feedback	Brainstorming	●
	Feedback	◐
	Providing counterarguments	◐
Writing	Synthesizing text	●
	Editing text	●
	Evaluating text	●
	Converting hand-written equations <sup>24/6</sup>	● <sup>+</sup>
	Generating titles & headlines	●
Background research	Summarization	●
	Condensing YouTube videos <sup>24/6</sup>	●
	Literature research	◐ <sup>*</sup>
	LLM-powered search <sup>24/6</sup>	◐
	Formatting references	●
	Translating text	●
	Explaining concepts	◐
Coding	Writing code	● <sup>+</sup>
	Explaining code	● <sup>+</sup>
	Translating code	●
	Debugging code	● <sup>+</sup>

TABLE 4  
SUMMARY OF LLM CAPABILITIES AND RATING OF USEFULNESS

cont.

Category	Task	Usefulness
Data analysis	Locating data sources <sup>24/6</sup>	●
	Creating figures	●
	Extracting data from text	●
	Reformatting data	●
	Classifying and scoring text	● <sup>+</sup>
	Extracting sentiment	● <sup>+</sup>
	Simulating human subjects	●
Math	Setting up models	●
	Deriving equations	● <sup>+</sup>
	Explaining models	●
Research promotion	Social media posts	●
	Presentation slides <sup>24/11</sup>	●
	Blog posts <sup>24/11</sup>	●
	Conducting interviews <sup>24/11</sup>	●
	Podcasts <sup>24/11</sup>	●

Notes: The third column reports my subjective rating of LLM capabilities as of November 2024:

○: experimental; results are inconsistent and require significant human oversight

●: useful; requires oversight but will likely save you time

●: highly useful; incorporating this into your workflow will save you time

Superscripts 24/6 or 24/11 in 2nd row represents year and month of new inclusions

Superscripts in last row denote upgraded ratings in 2024/06(\*) and 2024/11(+).

## 搭建 AI 辅助代码环境

- Anaconda + VScode 环境配置
  - 详情：DS with Python, [Chapter 6](#) | [Chapter 7](#)
- 基于提示词的编程和数据分析
- Stata+Python: Jupyter Notebook

## AI 工具

- [ChatGPT](#) - 由 OpenAI 开发的聊天机器人，基于 GPT-3.5 架构。可以用于编写代码、回答问题、生成文本等。
- [ChatGPT Plus](#) - 付费版本的 ChatGPT，提供更快的响应时间和更高的可用性。
- [DeepSeek](#) - 一款基于 AI 的搜索引擎，支持多种语言的搜索和翻译。
- [豆包](#) - 一款基于 AI 的社交网络应用，支持多种语言的交流和分享。
- [kimi](#) - 一款基于 AI 的智能助手，支持多种语言的语音识别和翻译。
- **其它/高级/多模态：** [杨阳 - 生成式人工智能专题](#)

# 理念

## 自然语言编程 vs. 传统编程

- 「自然语言编程」与 Python、C++ 等传统编程本质上都是向计算机发出指令，要求其执行特定操作。
- 区别在于：
  - 传统编程语言（如 Python、C++）有严格的语法和结构。
  - 自然语言编程则用人类语言（如中文、英文）描述操作。

## 思维方式与沟通能力

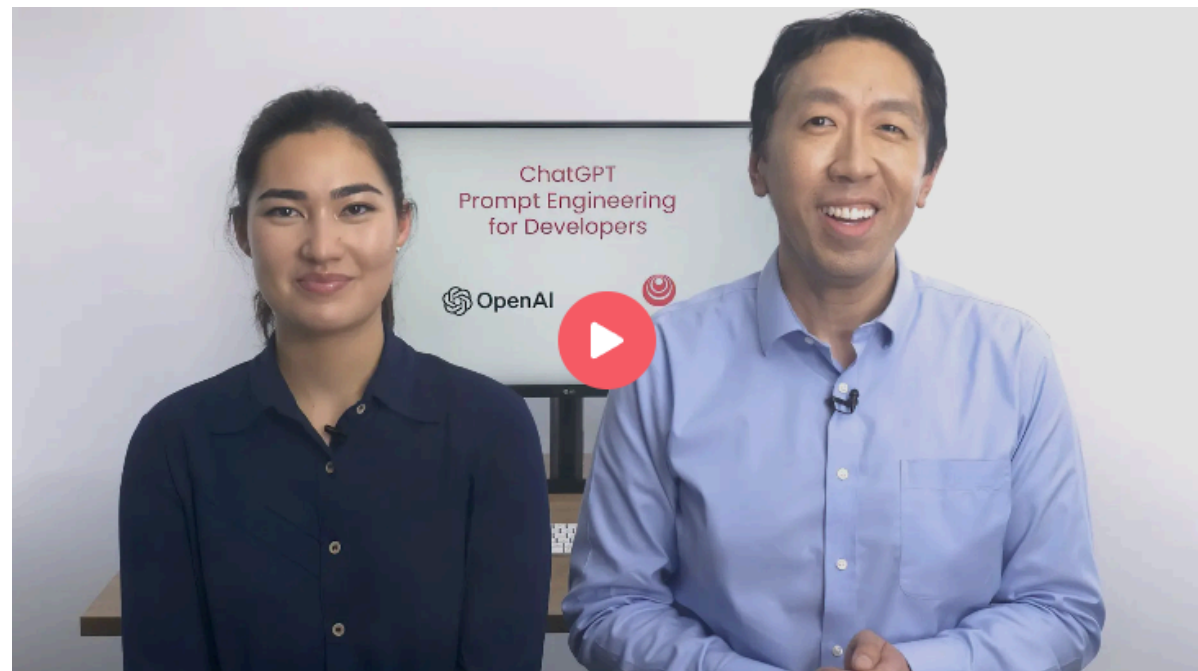
- 初学时，自然语言编程似乎更简单。
- 真正发挥其潜力，关键在于**思维方式**和**沟通方式**（如何提问）。
- 学习曲线很陡峭：
  - 知识广度：你要知道很多东西以及他们的关联，才能提出好的问题。
  - 知识深度：基本概念、核心理论、核心算法
  - 逻辑思维：界定问题、拆解问题、追问 (横向 v.s. 纵向)
  - 语言表达：简洁、准确、清晰

## 最核心的理念转变

- 提示词 = 自然语言的“代码”
- 写好提示词，就像写好 Python/C++ 代码一样重要。
- 许多高校已开设「提示词工程」课程，「Prompt 工程师」将成为热门职业。

## 推荐学习资料

- [Prompt Engineering Guide](#)
- 吴恩达老师的 [ChatGPT Prompt Engineering for Developers](#)



# 提示词

## Tips

- 先粗后细 e.g. 生成讲义 v.s 先细后粗
- 顺藤摸瓜-迁移 e.g. 各种抽样方法
- 虚构角色 e.g. 你是一个资深的英文经济学期刊的编辑 → 推文

## 收集整理自己的提示词

- 推文 · 提示词
- 连玉君的提示词: <https://github.com/arlionn/UseChatGPT>
  - 你可以 Fork 这个仓库
- 应用实例: [连玉君 - Empirical Research with AI](#)



# Empirical Research with AI