

下次不必 从头讲

Obsidian 是把笔记保存在本地的工具。
这场分享从整理资料出发，
谈到如何备课、写稿，
以及让多个 AI 助手协作。

JKL（主持人、教师）在开场提出：
我们能不能从自己寻找笔记，
走向人和 AI 共同使用知识？

阅读地图 / 三位讲者的实践

这一场分享， 从哪些问题谈起

当天共四位分享，本笔记收录其中三位。树语介绍个人知识管理；麻辣酱讲备课和写稿；JKL 讨论多个 Agent 怎样合作。Agent（智能体）可以理解为执行任务的 AI 助手。

01	第二大脑怎样帮你做事	03
02	维护资料要算哪些成本	04
03	从收集到使用，怎样安排	05-06
04	让 AI 了解你，又不塞太多	07
05	怎样选择和试用插件	08
06	备课为什么慢，交付怎样查	09-10
07	写稿的检查可以怎样做	11
08	多个助手怎样共享项目资料	12-14
09	现场提问，以及回去后怎么开始	15-16

●

树语

文科背景的分享者

●

麻辣酱

竞赛数学教师

●

JKL

主持人、教师

你可以先读最接近自己工作的问题。各章会说明观点来自谁，再用讲者的经历解释具体做法。

能帮你处理问题，才值得一用

树语把第二大脑理解为做事时反复借助的一组外部安排。它可以是笔记库，也可以是一张纸、一个白板或一份清单。

「它能让过去的经验被调用，让当前的问题被解决，让未来的行动更加可靠。」

树语 · V2 04:44

看什么	只看存放	也看使用
用什么	你只关注选哪个软件。	树语先看任务需要什么，再选择纸、白板或软件。
留什么	你存下读过的材料，就觉得整理好了。	你留下做事还会用到的信息，让过去的准备继续发挥作用。
怎么判断	库里资料越来越多，但日常工作仍要重新查找。	眼前遇到问题时，你能借助已有资料往下做。

树语喜欢用白板，把脑子里理不清的内容摊在面前。这样的安排能减轻记忆负担，也能让思考过程变得看得见。

怎样理解 你不必先搭一个庞大的资料库。先观察自己在哪件事上容易忘、容易重复准备，再寻找能帮上忙的安排。

整理是否值得， 看四种成本

树语提醒，知识管理本身也花时间。保存资料、重新组织、查找和使用，这四个环节都要算进去。

日常工作更省力

你能方便地记下资料，所以愿意持续记录。

你能重新组织已有内容，不必每次再读懂一遍。

你能稳定找到所需信息，不必在相似结果里反复挑。

资料能帮你完成讲稿、课程或其他产出。

维护反而更费事

记录手续太多，你常常来不及存。

材料虽然留着，每次重排却都很吃力。

你收藏了很多内容，需要时仍然找不到。

你花时间整理，却很少把材料用到工作中。

这里的「重构」，就是把旧材料重新组织成现在需要的样子。树语特别重视最后的生产环节：前面几步都做了，资料仍可能只是一堆不再打开的数据。

可以用下一次任务来检查：以前做过的准备，究竟替你省下了哪一步？

先把想法留下， 再整理成能用的材料

树语会先用语音留下想法，再阅读整理稿、调整结构。录音帮树语保存现场内容，语音输入则方便随时表达。

01

先做判断

你先挑出值得保留的信息。



02

及时记录

你用录音或语音输入把内容留下。



03

按规则存放

你把材料放入分类或收集箱。

04

需要时查找

你在具体工作中找回相关内容。



05

形成新理解

你复盘材料，补充自己的认识。



06

用于任务

你用材料完成讲稿或课程。

这次分享的准备过程

树语说，当天大约中午前后才开始做 PPT：先对着笔记库输入主题和想法，让 AI 整理，再继续微调。

树语参加活动后会复盘录音，看看自己实际说了什么；遇到难懂的概念，也会用划词解释帮助理解。

录得下来只是第一步。你还需要安排一次回看，让记录进入后续工作。

资料之外， 还要安排下一步

树语认为，一套完整的第二大脑还包括行动管理、自动化，以及恢复与同步。因为资料留住之后，仍要有人或工具把事情往前推进。

1 先让信息有地方放

树语用 Obsidian 和幕布保存内容，用白板展开调研。

2 再把事情变成行动

树语用看板和下一步清单，安排接下来做什么。

3 把重复动作交给工具

你可以把重复步骤交给程序或定时任务执行。

4 给恢复和同步留办法

版本记录和同步盘帮助树语恢复、同步资料。

收集箱和项目分开放

树语演示的空库里，收集箱区分未处理与已处理。

项目分为候选、进行中、已完成三类。

不适合放进知识分类的内容，可以先随项目保存。

材料处理过后，仍不知道归哪一类的，树语会先放进已处理的归档区。

树语也承认，自己会忘记整理收集箱。所以有了目录还不够，仍要定期处理积下来的材料。

第二大脑不必包办一切。你可以分别选择适合记录、行动和共同编辑的工具。

让 AI 了解你， 也要分清资料层次

树语观察到，很多人每次与 AI 交流，都要重新说明自己是谁、项目怎么做、有什么要求。把这些长期有效的信息留下来，可以减少重复交代。

「每一次和 AI 去交流，
……都需要去重新交代
我这个项目
应该怎么做。」

树语 · V2 34:45

一份长文全讲完

你把所有项目和偏好一次交给 AI。

资料太多，当前真正需要的要求反而容易被忽略。

先概览，再按需细读

树语建议主文档保持简短，需要哪一部分时，再读取相应的详细文档。

这叫「渐进式披露」，意思是先提供必要的概览，再逐步补充细节。项目资料、行为偏好和个人看法，可以分开整理。

树语用提示词（给 AI 的指令）梳理自我认识，先了解现状，再思考希望怎样改变。

先说清想改善什么，再去找插件

插件是给软件增加功能的扩展工具。树语早期一次装过很多，后来连哪个改了笔记库都说不清。因此，树语建议按实际需求逐个尝试。

容易这样做 你先装上评价好的插件。	→	树语的建议 你先说明想改善搜索、记录、白板还是外观，再问 AI 有没有对应工具。
容易这样做 你直接在日常使用的主库里试新功能。	→	树语的建议 你另建实验库，每次加一个插件，观察它带来的变化。
容易这样做 你只看功能多不多，觉得装上总会用到。	→	树语的建议 你同时比较能力提升和使用复杂度，先排除成本高、帮助小的选择。

有些插件停用后，也未必能完全恢复之前的状态。实验库的作用，是让你敢于尝试，又能知道每次改动来自哪里。

试用前先想

这个功能有没有解决你的具体问题？如果它改动了资料，你能否接受这样的变化？

AI 有了资料， 也未必理解任务

麻辣酱服务不同类型的学生，备课和资料管理都很费时。用上 AI 后，仍遇到资料乱放、结果忽好忽差的问题。

先让 AI 直接从资料里抽题

麻辣酱把 PDF 交给模型，让模型按学生情况匹配题目。答案看着合理，原资料中却没有那道题。

再发现资料和背景都不稳定

材料散在不同存储位置。即使设置了工作区，AI 也可能把文件放到别处。

于是开始统一管理资料

麻辣酱大约从五月起用 Obsidian 管理资料，把学生档案和做事步骤一并保存。

「上下文」是什么

上下文是 AI 做任务时读到的背景、要求和相关材料。

背景不稳定，同一个任务的结果就可能时好时坏。

资料给得过多，AI 也可能误解哪些要求该执行。

麻辣酱希望 AI 按规定的位置存资料、按既定步骤做事。这样，每次交付都能依据相同的要求检查，而不必靠换一个工具重新碰运气。

对需要长期服务的学生，资料和步骤越能积累，下一次备课就越有依据。

题目有没有出处， 交付能不能查证

麻辣酱强调，找题不能只匹配几个相同的字。题目要符合学生正在学的内容和难度，也必须能回到原资料里核对。

遇到的问题

题目匹配错了

麻辣酱要找与校内题结构相近的题。

AI 给了三角形的中位线题，仍不符合需要。

规定的做法

按任务选择步骤

做讲义就读讲义要求，分析错题就读对应要求。

AI 应按资料分区查找，并说明依据。

交付时检查

结果可以核查

记录文件位置、大小、经手人、日期和所用步骤。

版本多了，还要注明哪一份最新。

先说明哪些事需要人批准

麻辣酱会用项目说明写明 AI 能做什么、不能做什么，以及何时必须停止。AI 要修改这份规则，也需要人批准。

麻辣酱后来安排两个角色交叉检查，一个负责制作，一个负责检验。因为只听制作方说「完成了」，还不足以判断文件是否存在、内容是否正确。

资料怎样保存 Markdown 是一种用纯文本记录标题、段落等格式的写法。麻辣酱看重这类文本和表格，因为人和 AI 都方便阅读、检索。

觉得稿子有 AI 味，就把问题写具体

麻辣酱把「AI 味」拆成可以检查的项目，例如反复出现的句式、互动引导和篇幅。AI 味指读起来生硬、套路化的表达；只凭感觉挑，容易漏掉问题。

麻辣酱怎样检查一篇稿子	
检查句式	麻辣酱把「不是……而是……」「先……别……」等高频句式列入检查项，让程序提示问题。
检查要求	麻辣酱把互动引导、字数上下限写入规则，避免稿子出现不合要求的表达或长度。
参考旧稿	AI 出稿前先读过往稿件和口吻要求，让新稿有可参照的写法。
人工检查	程序检查不通过就继续修改；通过后，人再读一遍。图片里的乱码和奇怪的字也要核查。

麻辣酱说，PPT 里仍会偶尔出现生硬句式，规则还在调整。检查程序能帮助发现固定问题，但不能代替人判断整篇内容是否自然、是否说得清楚。

麻辣酱认为，这套积累更适合反复发生的工作，例如管学生、做讲义和写稿。

助手越来越多， 为什么人反而更忙

「这一群 AI 有没有变成团队，我不知道。但我自己肯定成了不停复制粘贴、传话和派任务的项目经理。」

JKL · 课件第 3 页

各自开工

每个助手只看到一部分

文案助手拿到聊天记录，视觉助手看着旧版本，开发助手还不知道项目定位变了。它们的结果因此可能互相矛盾。

共同理解

开始前先读同一套资料

每个助手先了解项目目标、已确认决定、不能擅改的要求，以及其他角色的职责，再处理自己的任务。

JKL 最初用 Obsidian 整理教学经验，例如不同年龄和英语水平适合什么教材、一个知识点怎样举例。后来工作涉及活动、内容和开发，问题也随之改变。

以前需要解决的是「将来还能不能找到资料」；使用多个助手后，还要解决「不同助手是否理解了同一个项目」。

JKL 用准备活动宣传来说明这个问题。这是课件中的流程示意，用来解释协作关系。

长期要求和临时进度， 分别放在哪里

JKL 建议把反复使用的背景与每天变化的状态分开保存。知识库让助手了解项目，看板则让助手知道当前能做什么。

01 长期资料放进知识库

项目背景、品牌规范、已确认决定和工作方法，可以留在知识库里，供不同助手反复阅读。

02 当前进度放到任务看板

看板记录任务、负责人、状态和下一步。文案还没确认时，视觉助手就知道需要等待。

03 一类任务配一份说明

Skill 可以理解为执行某类任务的方法说明。内容工作要读口吻和标题要求；视觉工作要读颜色和标志规范。

同一份说明不必对所有任务讲完所有细节。助手接手哪一类工作，就读取相应要求。这样既有共同背景，也能保留不同工作的具体标准。

这些资料也需要备份和版本记录，并让有权限的协作者能够读取，才能持续用于共同工作。

改了信息之后， 怎样让大家都知道

JKL 举了活动时间变更的例子：如果聊天、项目首页和文案各写一版，助手就不知道该相信哪一份。每种信息需要有明确的修改和确认位置。

为信息指定一个确认位置

活动信息看项目首页，品牌要求看对应工作说明，任务进度看看板。其他地方可以引用，更新时回到约定位置修改。

把做完的事记录到项目里

助手完成任务后，要记录结果位置、关键决定和任务状态，并说明接下来交给谁、还需要谁处理，方便下一位接手。

「下一位有没有拿到继续工作需要的信息？文件数量不是关键，交接能否继续才是关键。」

JKL · 课件第 20 页

JKL 把这种做法称为「写回」，也就是把工作结果和变化更新到项目记录。记录无需很长，但接手的人要能据此判断从哪里继续。

什么时候需要 JKL 认为，简单任务用一个 AI 就能完成。等项目角色和交接增加，再逐步补上相应安排。

一块白板怎么用，现场朋友追问了什么

树语演示时，现场朋友追问了白板和插件的具体用法。这些问题发生在分享过程中。

现场朋友 · 提问

白板是不是 AI 自动从 Markdown 转出来的？

树语 · 回答

树语解释，可以把笔记拖进画板；新建卡片也可以不进入主库，避免主库越来越臃肿。

现场朋友 · 继续问

刚才那个英文插件叫什么？

树语 · 回答

树语说刚才确实没报名字，随后演示了第三方插件的查找入口。相关功能拆成多个扩展，可以按需加载。

以上根据现场问答整理 · V2 29:56–32:08

树语还演示划词解释 GTD。GTD 是一种把待办梳理成具体行动的方法。树语也提到可以用下一步清单，或 SOP（标准操作步骤）来帮助自己行动。

这一段的具體收获是：白板可以帮助展开思考，卡片是否进入主库，也可以按需要安排。

从哪件事开始， 由你的工作决定

树语建议在 72 小时内开始使用。树语是文科背景，技术基础不强，因此希望大家先从实际使用开始。

01 如果你总在重新找材料

可以试着把一次任务用过的资料留好。树语关心的是，未来再遇到相似问题时，这些准备还能不能帮上忙。

02 如果你的交付总有遗漏

可以参照麻辣酱的做法，把要求和检查依据写清，再用真实交付检验这些规则是否有效。

03 如果你总在替助手解释

可以借 JKL 课件中的问题想一想：谁需要理解你的项目？这个人或助手，此刻缺少哪一部分资料？

JKL 在开场还谈到，执行可以交给 AI，但人仍需要想清问题、定义标准，也听见不同声音。选什么工具，要回到你自己正在面对的工作。

先试着完成一次任务，
再看这套安排应该怎样调整。

摘自 AI 神思会第 10 期（广州）
视频号 / 小红书：JKL神思记

