**大模型 Agent 训练营**

我们坚信一个观点：AI Agent，不是下一个热点，而是下一个时代。它不会是ChatGPT那样的聊天工具，而是会思考、会执行、会协作的“数字员工”。未来3年，懂得构建和驾驭Agent的人，将成为最稀缺的资源。

很多学生看了那么多AI课，简历上还是没东西可写？因为学的都是玩具demo。真正的能力，来自于你能否掌握底层原理，能否解决真问题，能否复现顶会的核心思想。

因此我们推出了AgentAlpha训练营计划，逼你从第一性原理出发，不再是“学习一个工具”，而是在获取一种定义未来规则的核心能力。这条路线和后续的训练营，就是一张“造浪者”的船票。



**从零到一掌握大模型 Agent 核心技术**

|  |
| --- |
| **团队介绍**  😎原字节3-2，AI大模型算法方向大佬，有10多篇Agent、RL方向顶会论文和SOTA开源项目，90后年轻高潜，多段大厂跳槽成功涨薪80%经历，有语音多模态、NLP、大模型Agent方向丰富业界经验  😊MIT博士，985青年教授，担任腾讯、oppo等知名大厂AI lab产学研项目合作的资深研究科学家，多次斩获CVPR、ECCV等顶会Workshop challenge的全球冠军，12篇顶会，具备极强的学术届和工业界综合背景  😊香港科技大学计算机系博士后。研究聚焦高效 LLM/VLM 训练与推理、联邦学习与分布式系统、以及基于外部工具的智能体应用，代表性成果发表于/入选 NeurIPS (含 Spotlight)、ICML 、COLM、ACL、与 ICLR 2025（含 Blogpost），共发表论文80余篇，引用量上千。并获 NeurIPS 2024 Top Reviewer、FL@FM-NeurIPS’24 最佳学生论文奖等荣誉。当前他致力于以结构化训练范式与高效缓存/压缩等技术，推动可落地的 LLM/VLM 智能体。  😁25 年面试超 80 场 国内 LLM 大厂通过率 100% 拿到多个大厂顶尖人才计划 offer 的中科院 phd  社群还有 字节seed、Meta、Google、Apple、阿里p7..... （欢迎您的加入！） |

|  |
| --- |
| **为什么你值得加入？**  面向人群：  1️⃣ 零基础想转大模型，迷茫没有方向的——告别"学了就忘"，掌握系统化技术路径  2️⃣ 非科班出身，缺少大模型项目实战的——无需计算机背景，从基础到高阶完整闭环  3️⃣ 想找实习/校招/社招，简历含金量不够的——拥有可展示的、有深度的项目成果  4️⃣ 自学经常卡壳，需要手把手辅导的——全程实战指导，避免在 GitHub 示例中迷失 |

|  |
| --- |
| **不止于"学会"，而是"实用"**  你将掌握的核心能力：  ✅ 从单 Agent 到多 Agent 系统  基于 AutoGen，实现工具调用、Self-Reflection、多智能体协作（Planner/Researcher/Coder），构建企业级 Agent 应用  ✅ 深度搜索（DeepSearch）技术  复现 Search-o1，掌握推理中检索（Agentic Search In-Loop）与文档内推理，解决传统 RAG 的痛点  ✅ Code Agent 实战  精通前沿 SWE-agent 与 RepoMaster，实现仓库级代码理解与复用，完成真实 GitHub issue 修复  ✅ 自进化编码与强化学习  掌握 自进化agent与 Agentic RL，实现模型的持续进化 |

|  |
| --- |
| **为什么这个课程与众不同？**  🔥 真正实战：不是理论讲解，而是从 AutoGen 核心示例开始，每阶段都有可交付的代码项目  🔥 企业级技术栈：覆盖 vLLM、LangGraph、OpenEvolve 等工业界主流框架  🔥 解决真实痛点：针对"搜索时机不当"、"代码生成质量低"、"Agent 协作效率低"等实际问题 |

|  |
| --- |
| **学完能达到的水平**  ✅ 技术能力：能独立设计、实现、优化完整的 Agent 系统，从单 Agent 到多 Agent 协作，再到深度搜索与代码 Agent  ✅ 面试竞争力：实习/校招同学，学完可达到互联网大厂 LLM agentic 工程师的面试要求，拥有可展示的项目成果  ✅ 职业竞争力：社招转大模型的同学，学完可达到一到两年左右大模型工程师的能力水平，直接参与核心项目 |

|  |
| --- |
| **课程特色**  💡 从"跑通示例"到"解决实际问题"：不只是跑通 AutoGen 官方示例，而是通过阶段式任务，逐步构建能解决真实问题的 Agent 系统  💡 深度对比分析：在每个阶段都要求进行性能对比（如 ReAct vs. Reflection， 仓库复用 vs. 从零生成），培养工程思维  💡 前沿技术全覆盖：涵盖 Agentic Search、DeepSearch、Code Agent、自进化编码、Agentic RL 等最新技术方向 |

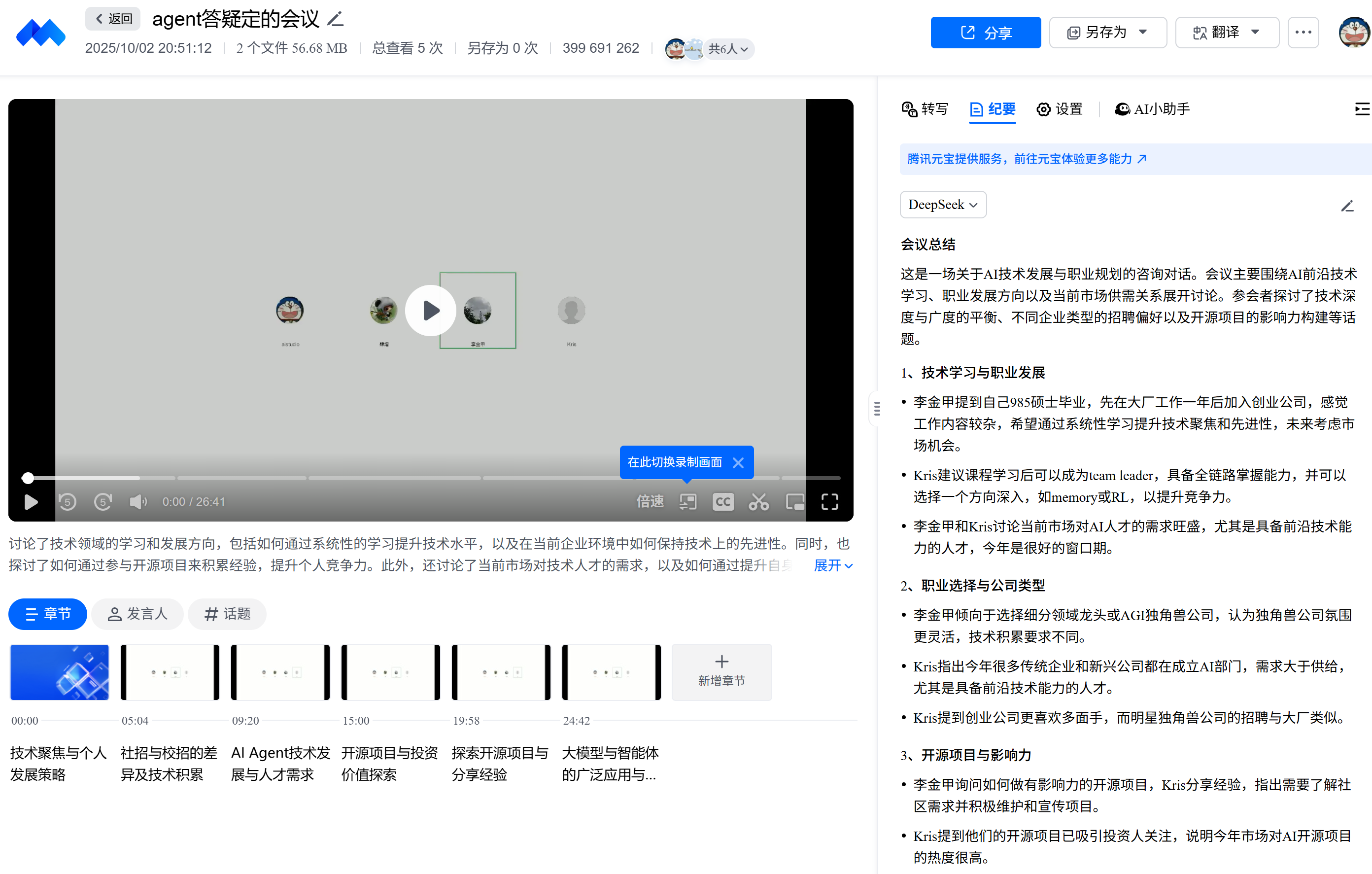
|  |
| --- |
| **适合谁？**  👉 想进入大模型/Agent 领域但不知从何入手的开发者  👉 想提升简历质量，增加大厂面试筹码的求职者  👉 非科班但对大模型有热情，需要系统化指导的学习者  👉 自学遇到瓶颈，需要实战指导与项目驱动的学习者  别再让"理论"与"实战"之间隔着一条鸿沟——加入我们，从今天开始构建你的大模型 Agent 工程师能力！ |

**课程介绍视频和答疑 meeting**



**概要**：细介绍了课程的十个阶段，包括 RAG、Memory、Single Agent、Multi Agent、Deep-Research、Coding Agent、自进化 Agent、Agentic RL 等

录制文件：https://meeting.tencent.com/crm/2G4ZrQVAd0



**录制： 答疑的会议**

**概要**：主要围绕 AI 前沿技术学习、职业发展方向以及当前市场供需关系展开讨论。参会者探讨了技术深度与广度的平衡、不同企业类型的招聘偏好以及开源项目的影响力构建等话题。

录制文件：https://meeting.tencent.com/crm/KwBp84aQ74



**录制： 一对一辅导的会议**

**概要**：强化学习与深度学习的理论与实践探索

录制文件：https://meeting.tencent.com/crm/N8XDyErX4b

以下是包含全部 10 个阶段、精简版课程介绍：



|  |  |
| --- | --- |
|  |  |

**agentic 开发实战课程（精简版）**

**阶段 1｜RAG：从理论到实践**

**掌握内容**：

* RAG 基本架构：检索器 + 生成器协同机制
* 文档预处理策略：切分粒度、元数据注入、嵌入模型选择
* 向量数据库选型与使用：FAISS、Chroma、Pinecone 等对比
* 检索质量评估指标：召回率、相关性、延迟、幻觉控制
* 混合检索策略：关键词 + 向量 + 重排序（re-ranker）

**核心实践任务**：

* 搭建端到端 RAG 系统，支持 PDF/网页等多源文档摄入
* 实现检索结果过滤、上下文拼接与引用标注
* 对比不同切分策略与嵌入模型对最终答案质量的影响

**阶段考核**：

* 完成一个可回答指定领域问题（如技术文档问答）的 RAG 系统
* 输出答案需包含引用来源，并通过人工或自动指标评估其准确性与幻觉水平

**实战产出**：（北美金融独角兽定制项目）搭建一个通用型智能知识引擎系统，并完成全链路评测报告，支持任何企业将非结构化文档转为可对话的知识资产。

**阶段 2｜记忆系统：赋予 Agent 长短期记忆**

**掌握内容**：

* 记忆类型区分：
* 短期记忆（working memory / message buffer）
* 长期记忆（persistent memory）：语义事实（semantic）、事件记忆（episodic）、程序性技能（procedural）
* 记忆内容策略（What to store）：用户偏好、关键事实、行为模式、交互事件
* 写入策略（When / Where）：实时 vs 异步写入；内存、向量库、图数据库、KV 存储
* 提取机制（How to retrieve）：相似度搜索、metadata 过滤、时间衰减、top-k 召回
* 摘要与压缩：滚动摘要、周期性压缩、自动/手动触发
* 记忆持续性与淘汰：跨会话记忆保持、记忆衰减（decay）、过期删除策略

**核心实践任务**：

* 使用 LangChain 的 ConversationBufferMemory 管理短期对话上下文
* 基于 mem0 框架构建长期记忆层，支持混合存储（向量 + KV + 图）
* 设计记忆写入规则：区分噪声与关键信息

**阶段考核**：

* 为阶段 1 的 RAG 系统集成记忆模块
* 重启对话后，Agent 能准确回忆并使用用户指定的“项目名称/代码风格/输出格式”等长期信息

**进阶挑战**：

* 设计记忆提取器，从长期记忆与短期摘要中召回最相关条目
* 通过实验初步验证该机制是否能有效降低模型幻觉

**实战产出**：能记忆的智能科研文献管理与知识图谱系统，专门为研究生和科研工作者提供文献管理、关系挖掘和科研辅助功能。

**阶段 3｜单 Agent 架构与强化：基于 AutoGen 的实践**

**掌握内容**：

* Agent 基础结构：推理、工具调用、输出生成
* 工具调用机制：函数注册、参数解析、执行环境隔离
* Self-Reflection（自我反思）：输出后自检、改写、错误修正
* 单 Agent 的局限性：上下文长度、任务复杂度、容错能力

**核心实践任务**：

* 跑通 Microsoft AutoGen 官方核心示例（AgentChat、工具调用、OpenAI/本地大模型适配）
* 熟悉经典 Agent 范式：ReAct、Reflexion、Tool Use，理解其推理链与工具调用逻辑
* 实现一个简单 ReAct Agent（支持 Web 搜索 / 本地计算）
* 为 Agent 增加工具（如计算器、代码执行沙箱）
* 加入简单 Self-Reflection 机制（输出后自我检查并改写）

**阶段考核**：

* 构建一个「增强版单 Agent」：能完成小型任务（如查找信息 + 计算结果），并在出错时自动反思并修复

**阶段 4｜多智能体协作：基于 AutoGen 的实践**

**掌握内容**：

* 多 Agent 系统设计理念：为何需要协作？单 Agent 的瓶颈
* 角色分工：Planner（规划）、Researcher（检索）、Coder（编码）、Critic（评审）
* 多轮对话与任务分解：谁规划、谁执行、谁验证
* 消息流与对话管理：GroupChat、轮次控制、终止条件
* 协作机制：多 Agent 版 ReAct、Peer Critique（互相纠错）
* 编排思想：有限状态机（FSM）与图驱动流程（如 LangGraph）
* 状态（state）、条件（condition）、转移（transition）
* 支持回滚、重试、人工干预
* 对比框架：LangGraph、CogFSM 等设计思想

**核心实践任务**：

* 跑通 AutoGen GroupChat 示例，体验 Planner + Executor + Critic 协作
* 用 AutoGen 实现多 Agent ReAct 流程
* 加入 Reflection Agent 进行批判性审查
* 学习 LangGraph 基本语法，实现小型 FSM（如三 Agent 编程任务，含失败回滚）

**阶段考核**：

* 构建「三人协作小组」：Planner + Researcher + Coder
* 要求：
* 至少一次反思/批判性反馈
* 至少一个 FSM 驱动的流程控制（含回滚/条件跳转）
* 接入记忆层，实现跨任务上下文延续

**阶段 5｜DeepSearch 路线**

**掌握内容**：

* 推理中检索（Agentic Search In-Loop）：动态触发检索（何时/多少/停止）
* 文档内推理（Reason-in-Documents， RiD）：对长文档结构化预处理（切分/压缩/证据对齐），先在文档内推理再注入主链
* Agentic RAG vs 传统 RAG：差异、优势、适用场景

**核心实践任务**：

* 复现最小 Search-o1 闭环
* 开放领域任务实战（如论文综述、产品调研）
* 与“纯 RAG”做对照实验

**阶段考核**：

* 针对开放领域问题集，实现完整“思考→搜索→整合”链路
* 输出含“引用来源”和“证据热区”的报告
* 与阶段 1 的纯 RAG 系统对比，从答案质量、引用准确性、推理链完整性三维度分析

**阶段 6｜Code Agent 准备阶段：高效推理与大规模服务**

**掌握内容**：

* 解码与缓存优化：KV cache、PagedAttention、连续批处理
* 推测解码：Speculative Decoding / Medusa 基本原理
* 量化技术：GPTQ、AWQ、SpQR、SmoothQuant；INT4 / FP8 权衡
* 服务化部署：vLLM / sglang 的并发控制、限流、流式接口
* 可观测性：吞吐、延迟、显存监控

**核心实践任务**：

* 使用 vLLM 部署一个开源模型，开启连续批处理与 KV 复用
* 提供流式生成接口，支持多并发请求

**阶段考核**：

* 提交部署与推理实验结果，展示吞吐/延迟对比，并演示流式输出效果

**阶段 7｜Code Agent：深入 SWE-agent 等框架**

**掌握内容**：

* 端到端代码任务流程：接收 issue → 检索/理解代码 → 编辑/生成 → 测试/验证 → 提交
* 仓库级代码理解技术：函数调用图、依赖图、层级代码树
* 评测基准：SWE-bench、GitTaskBench
* 前沿框架：SWE-agent、OpenHands、RepoMaster

**核心实践任务**：

* SWE-agent：在真实 GitHub 仓库中完成 issue 修复、CTF 挑战或工具流
* RepoMaster：掌握通过函数调用图/依赖图进行仓库级探索与复用
* 对比“从零生成” vs “仓库复用”在任务成功率与 Token 消耗上的差异

**阶段考核**：

* 成功运行 SWE-agent 修复真实 issue
* 撰写简短复现报告，包含性能对比与失败分析

**阶段 8｜AlphaEvolve 系列：掌握自进化编码**

**掌握内容**：

* 自进化闭环：生成 → 评估 → 筛选 → 迭代
* 多种群与多样性维护：MAP-Elites、岛屿模型
* 并行评估策略：提升候选解评估吞吐
* 前沿方向：SE-Agent 的轨迹级自进化思想

**核心实践任务**：

* OpenEvolve：理解其进化循环、MAP-Elites/岛屿模型、并行评估机制
* OpenAlpha\_Evolve：学习其工作流，与 OpenEvolve 对比
* 关注社区讨论（如数据库共享、MCP 客户端扩展）

**阶段考核**：

* 在可量化小任务（如算法优化、脚本提速）上跑通完整自进化流程
* 提交实验记录：性能曲线、多代种群演化、筛选策略、失败案例分析
* 进阶：结合 SE-Agent 思想，思考如何迁移到代码/检索策略进化

**阶段 9｜Agentic RL：基于 Search-R1 的强化学习实践**

**掌握内容**：

* 将“搜索/工具调用”建模为 RL 动作：何时 search，何时 reasoning
* 交错式推理 + 多轮搜索的实现与训练流程
* 奖励函数设计：答案正确性、格式合规性、工具使用效率、搜索次数代价
* RL 方法选择：PPO、GRPO、Reinforce 对比
* 训练稳定性技巧：retrieved token masking、防捷径策略

**核心实践任务**：

* 复现 Search-R1 论文与代码，训练 3B–7B 模型学习搜索时机
* 对比训练前后：检索次数、搜索触发位置分布、答案质量（F1/EM）

**阶段考核**：

* 在小规模问答集上完成 RL 微型实验
* 报告必须包含：
* 检索次数 / 触发位置分布变化
* 答案质量与“引用有效率”提升
* 1–2 个失败样本 + 改进思路

**阶段 10｜综合项目考核（1–2 周）**

**三选一项目（或与导师协商自定义）**：

1. **DepResearch × CodeAgent**：以 Search-o1 为检索-推理前端，后接 RepoMaster 进行仓库级代码复用，解决真实工程任务（如跨仓构建数据处理/评测工具链）
2. **自进化 × 代码/推理**：用 OpenEvolve 完成代码优化任务，并将 SE-Agent 的轨迹进化思想引入“检索-推理”链，对比两类进化在效率/质量上的贡献
3. **Agentic RL 应用**：在知识密集型领域训练微型 Search-R1 代理，与“纯 Prompt 工程的 Search-o1”对照，量化 RL 带来的性能增益

**交付成果**：

* 一份完整技术报告：含方法设计、实现细节、实验日志、核心指标、失效分析、复现脚本
* 一个可完整复现的代码仓库：含详细 README、一键执行脚本、小规模示例数据

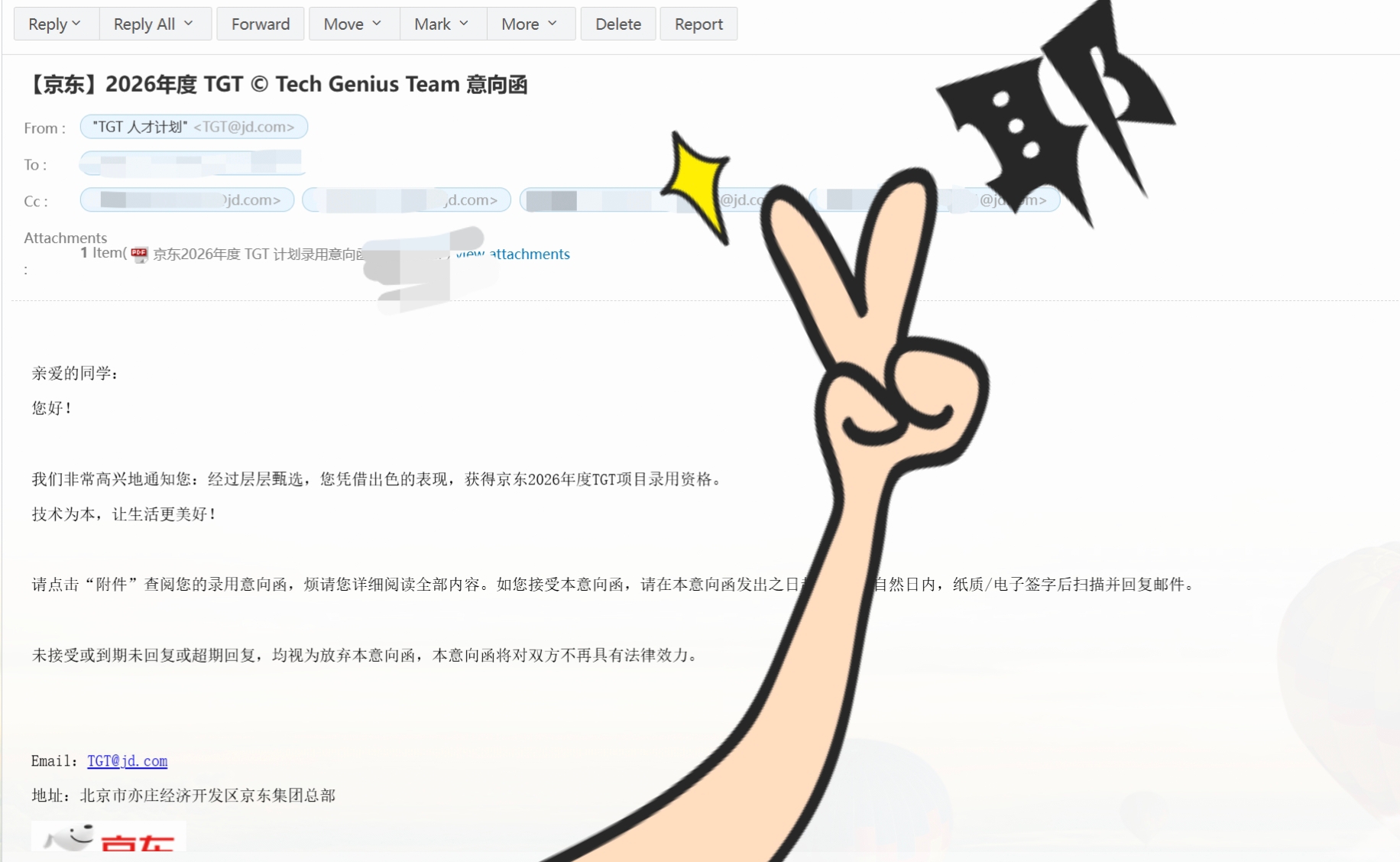
**往期课程**

|  |
| --- |
| [**跟着我们，两个月就搞定LLM~ （去年的课）**](https://vcneiljwtkzh.feishu.cn/docx/VjP3djEdCoJgjtxTLAPcmm1Lntc?from=from_copylink)  [**多模态大模型小班课 + 从零开始的具身智能课 + 基础课**](https://appjtakvrjf8935.h5.xiaoeknow.com/p/decorate/homepage) **（欢迎咨询）** |

**🎯 往期学员成果（真实案例）**

|  |
| --- |
| ▸ 某 985 研究生：课程期间完成「多 Agent 代码助手」，GitHub 获 120+star  ▸ 前 NLP 工程师：转行 agent 架构师，薪资涨幅 80%，现就职某大厂智能客服系统  ▸ 优秀学员内推案例：**字节 seed、腾讯青云计划、京东 TGT 人才计划、蚂蚁星计划 PLAN-A...**  ........ |

|  |  |
| --- | --- |
|  |  |





|  |  |
| --- | --- |
|  |  |

|  |
| --- |
| **📌 课程形式与资源**  ▸ 直播+录播+代码库（提供企业级 Agent 项目源码）  ▸ 字节 3-2， 顶会 nips spotlight 得主，大厂 P7-P8 导师每周固定答疑  ▸ 结业的优秀项目获内推机会 |

**正在辅导的例子：**

细致的认真的指导，确保学员能够掌握并启发新想法:

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 大厂面试 | 项目实战 | 启发思考 | 每周答疑会 |

**学习进度跟踪**

✅ 既然花了钱报名，就不许躺平，我会要求你每周提交学习周报，及时反馈学习成果。还会有阶段性考核，对于优秀的学生，我们会毫不吝啬进行强推（已有学生入职seed，kimi 等团队）

以下是学员周报和笔记：（含以往的辅导）

|  |  |
| --- | --- |
|  |  |

**论文辅导**

|  |
| --- |
| **题外话， 这两年参与论文课题的部分记录：** |



|  |  |
| --- | --- |
| nips 合作后推荐港大读博 | 211学弟拿到5个985的直博offer |

**一对一辅导老师详细介绍**

|  |
| --- |
| Ben老师  MIT博士，985青年教授，担任腾讯、oppo、字节、阿里等大厂AI lab产学研项目合作的资深研究科学家，多次斩获CVPR、ECCV等顶会Workshop challenge的全球冠军，具备极强的学术届和工业界综合背景。研究领域主要包括大模型可解释性、多模态大模型、CodeAgent、等。共发表十余篇CCF-A类国际顶级会议和SCI国际高区期刊，包括一区期刊IEEE Transactions on Pattern Analysis and Machine Intelligence (影响因子24.31)，IEEE Transactions on Neural Networks and learning systems (影响因子10.45)，及二区期刊IEEE Signal processing letters (影响因子3.20)等；发表CCF-A类顶级会议论文CVPR, ECCV, ACM MM等, CCF-B类会议PRICAI等。  长期担任计算机视觉、人工智能、多媒体领域顶级会议CVPR, ECCV, ACM MM、NeurIPS等诸多审稿人， IEEE Transactions on Neural Networks and Learning Systems，IEEE Transactions on Neural Networks and learning systems 等国际顶级期刊审稿人。  指导博士研究生、硕士研究生、本科生、公司实习生10余人，曾获**腾讯、华为、oppo等产学研项目合作支持（项目资金支持数百万级）**。 |

|  |
| --- |
| Jack老师  香港Top3博士后研究员，曾赴新加坡国立大学交流访学，亦有知名科技公司研究经历，学术与产业背景兼备。多次在 ICLR、NeurIPS、ICML、ACL、AAAI、ASPLOS 等国际顶会发表80多篇论文，获 **NeurIPS Spotlight、Outstanding Student Paper Award** 等殊荣。研究方向涵盖Agentic RL、Self Evolve Agent、大模型推理加速、联邦学习、分布式训练与智能体（Agent）系统，致力于推动 LLM 在实际场景中的落地与优化。长期担任 NeurIPS、ICML、ICLR、ACL、COLM 等国际顶会审稿人，亦为 TPAMI、TMLR、TNNS 等国际高水平期刊评审。指导博士、硕士、本科生及业界实习生多名，具有丰富科研辅导与项目指导经验。 |

|  |
| --- |
| Nicky老师，中科院博士，已拿到Top大厂 LLM Agent方向人才计划offer，在ICML, NIPS, ACL等顶级期刊会议发表论文20余篇，研究方向涵盖Agentic reasoning, CodeAgent, Self-evolve等如今业界最前瞻方向，指导博士、硕士、本科生及业界实习生多名，具有丰富科研辅导与项目指导经验。 |

**感兴趣可联系wechat：**

