**姓名**：李之南  
**邮箱**：zhinan.li@email.com  
**地点**：辽宁 铁岭，现居杭州  
**语言**：中文（母语）、英语（流利）

**职业目标**  
以AI之力服务于真实世界的不完美与复杂，建设具备社会责任感的企业智能系统。

**技能专长**

* AI建模：Transformer、GNN、XGBoost、因果推断
* 编程/数据：Python、SQL、Scala、Airflow、Databricks
* SAP系统：SAP HANA ML、SAP BTP、SAP AI Core
* 工程实践：微服务架构、CI/CD、MLOps
* 其他：Prompt Engineering、多语言NLP、Explainable AI

**工作经历**

**AI科学家 | 独立顾问 & 项目合作 | 杭州 / 远程**  
2024–至今

* 为西欧一初创公司设计客户交互预测系统
* 为国内中型电商设计多语言情感识别系统，支持南方方言
* 撰写模型评估与公平性审计文档

**职业空档期 | 照顾家庭成员 & 深造学习 | 铁岭**  
2021–2023

* 全职照顾长期患病家人
* 期间自学SAP BTP，参与Kaggle竞赛获Top 5%
* 博客撰写AI系统设计中的公平性与可达性思考系列

**数据科学家 | 某大型零售集团（SAP生态） | 杭州**  
2017–2020

* 主导客户智能分层模型建设
* 推动SAP AI Core与本地CRM集成
* 发起“包容性AI”学习小组

**教育背景**  
MSc 人工智能 – 香港科技大学（奖学金）  
BSc 应用数学 – 湖南师范大学

**对话 1｜SAP AI Core 项目挑战与取舍**

**面试官**：你在使用 SAP AI Core 的过程中，有遇到什么技术上的挑战吗？

**李之南**：有，最大挑战其实是资源调度和模型回滚机制。我们之前把一个微调的情绪识别模型部署到 SAP AI Core 上，初期没设定好并发限制，导致多个预测请求同时写日志，系统负载飙高，错误日志也混乱不清。  
我后来设计了一个轻量的请求缓冲模块，用 SAP BTP 的函数服务处理前置清洗，还加了异步日志归类，成功把故障率降到了 0.7%。最有趣的是，我们还加了一套 AB 测试的 shadow 部署机制，在不影响现网的情况下测试更新模型，这一点 SAP 的 API 设计其实挺灵活的。

**对话 2｜职业空档期的成长**

**面试官**：这两年你没有全职工作，是因为什么原因？

**李之南**：我在老家照顾母亲，她那时身体状况不太好。我选择了放慢职涯节奏，但也没真的“停”下来。我一边照顾她，一边系统学了 SAP BTP 和 AI Core 的底层机制，还参加了两个远程实战项目，其中一个是做多语言情绪分析。我还试着用语音技术做了个定时提醒App，主要是想让母亲能自主吃药。这段时间让我真正意识到，技术的“可达性”意味着什么。

**对话 3｜技术短板与学习方式**

**面试官**：你觉得自己目前还有哪些技术短板？

**李之南**：我目前还在攻克两个点：一个是 SAP DI 的 pipeline 复用设计，另一个是复杂事件驱动架构下的模型版本管理。我不是那种一口气全通的选手，我更喜欢边踩坑边搭文档，边跑测试边总结模式。我的 github 上现在就有一份 SAP AI Core + Airflow 的联动模板，我打算年底前发布成教程。

**对话 4｜包容性产品设计**

**面试官**：你有没有在设计模型时考虑过包容性的问题？

**李之南**：当然有。我参与过一个 HR 面试评分系统项目，我们发现模型对“非标准简历”得分偏低，比如学历断档、工作经历换行业的候选人。我提出将“时间序列特征”改为“语义事件链”，比如“主动转行+创业失败+重返职场”被当成有韧性的信号，而不是负面打分。上线后我们模型通过率提升了 12%，而且 diversity hire 的比例也稳步上升。我始终认为，包容不是“修补某些人”，而是“重新定义谁被看见”。

**对话 5｜跨文化协作与适应**

**面试官**：你有没有和外国客户合作的经历？

**李之南**：有，我在去年和一家西班牙SaaS公司远程合作，做过一个客户行为预测系统。他们的沟通风格非常“口语化”，很多技术细节都是在WhatsApp语音里说出来的（笑）。我就主动做了中英文对照文档、系统设计图，定期发给他们审阅。我们后来用这个版本推进到了测试上线。我觉得跨文化合作最难的是你要主动去“做翻译”，不仅是语言，更是沟通节奏和理解方式。

**对话 6｜技术领导力与架构意识**

**面试官**：你最近设计的系统架构能讲讲吗？

**李之南**：可以，我在为一家电商平台做实时情绪识别系统。我们从客服对话中实时抽取情绪信号，最初模型延迟在 1.5 秒左右，影响响应节奏。我后来把模型拆分为轻量预测+精度补偿两个阶段，轻量模型实时运行，重模型定时纠偏，最终延迟降到 600ms。部署用的是 SAP AI Core，前端调用通过 BTP Functions 暴露 API，还配了一层错误识别+自动回退机制，确保线上稳定。

**对话 7｜AI 与社会责任的思考**

**面试官**：你怎么看 AI 和社会责任之间的关系？

**李之南**：我其实一直在想这个问题。可能是因为我个人身体上有点小状况，从小也不太符合大家对“理工男”的标准。我走得慢，敲字有时没那么快，但思考和逻辑并不慢。这种“被误判”的经历让我特别警惕技术可能制造的系统性偏见。

比如我们建模时总想找“最优解”，但真实世界从来都不是最优的。那些看起来“效率低”的人、经历有点“混乱”的人，其实往往最值得被听见。我希望我参与的 AI 系统，不是筛掉他们的那一套，而是能学会“看懂他们”的那一套。技术不是拿来贴标签的，是拿来拆掉偏见的。