

翟羽京

AI 产品经理 · 作品集

用「先定义什么算好，再动手」的方式做 AI 产品

数字媒体技术 · 本科 · 2026 年 7 月毕业 · 杭州 · 可周内到岗

18562269318 | 13188772585@163.com

在线版: zhaiyujing-portfolio.pages.dev

本版重建于 2026-10-17 | 口径取自可核验的工作文件

一句话定位

我不是「做过 AI 功能」的产品经理，而是能把「什么叫好」定义成可判定标准、并自己动手把判定跑起来的人。

读这份作品集只需要问三个问题：① 标准是不是他定义的？② 判定是不是真跑过？③ 他有没有如实说清边界？——每一项下面都给证据。

我证明三个能力

① 定义力 · 把模糊的「好」变成可执行的标准

把「内容合格」拆成 C1-C6 六条可判定约束，并想清楚哪两条该设为一票否决——因为它们的误放代价不可逆，而其余问题可靠重试与人工兜底收敛。

- C1-C6 判据表、三档判定方式（代码判 / LLM 初筛 / 人工终判）、评分与分流规则
- 31 篇真实语料实测出文风基准，用 15 个场景标定出净值放行线（详见「门槛标定实验」）

② 判断力 · 从证据反推产品决策

所有结论都标注证据强度，而不是当成事实陈述。

- 全文区分「硬证据 / 推断 / 无法验证」三档，逐条标注来源

③ 落地力 · 真跑通过，不是画图

用 Python 标准库独立实现 RAG 语料索引、自动判分器、HTTP/SSE 后端与可视化工作台，零第三方依赖、可离线运行；配套 73 项回归测试，近半数覆盖失败路径。

- 在桩模式下跑通完整链路 52 次、260 次节点执行、936 条 trace，终态交付 20 / 降级 10 / 转人工 22——异常分流与人工接管路径均被实际走通。

项目一 · 内容数据查询与分析 (SQL)

独立完成 | 基于自有抖音账号真实数据 | 2026.08-2026.09

为什么做

内容是「爆款驱动型」还是「稳定输出型」，不能靠感觉判断。我把 7 个月的后台数据入库，用 SQL 做归因，目标是把「优化什么」从直觉变成可验证的结论。

做了什么

- 抖音后台导出 68 条记录，清洗入库 57 条公开作品（总播放 357 万、10w+ 爆款率 19.3%、单条最高 87.4 万、播放中位数 9,146）
- 设计单表 23 字段，派生互动率 / 涨粉率，建 3 个索引
- 编写 22 条 SQL，含窗口函数 NTILE 计算爆款集中度、CTE 分层聚合

关键结论（反直觉的那一条）

指标	与播放量的相关性	含义
5s 完播率	+0.546	最强正相关——推流闸门在前 5 秒
整体完播率	+0.099	几乎不相关——看完不是推流关键

据此把优化动作从「压缩时长」改为「重做前 3 秒」。

进一步定位到的转化断点

单条作品	播放	主页访问	涨粉
最高的一条	87.4 万	2,425	103

结论：瓶颈不在内容供给，而在主页访问 → 关注 → 成交这一步。不是「涨粉慢」，而是转化链路在这一环断了，断点在内容没有给出关注理由。

诚实边界

- 相关系数是自有账号 57 条作品的样本，样本量小，相关性不等于因果
- 后台数据来自本人账号，可对照后台核验，但无法由外部第三方验证

项目二 · AI 内容生产 Agent (0→1, 含多智能体迭代)

独立完成 | 个人项目 | 2026.06-2026.09

为什么做

内容脚本创作稳定性差、耗时长、质量不可控、产能依赖个人状态、无法规模化。我要解决的不是「能不能生成」，而是「什么样的稿子算合格、谁来判、判不准怎么办」。

需求拆解

把全链路拆为选题、RAG 检索、骨架生成、脚本输出、质量校验五段，逐段定义交付物、判定标准、判定主体与兜底路径。

方案设计

- 以「一条主线 + 6 个小论点」叙事模板与 C1-C6 六条硬约束为内核，把「内容好坏」翻译成机器可判定的规则
- 类比只允许来自知识库（31 篇语料与类比清单），库外自创一律拦截

工程实现

- Python 标准库实现 RAG 语料索引（语义检索 + 行号级溯源）、自动判分器与前端工作台，零第三方依赖、可离线运行
- 配套 73 项回归测试，近半数覆盖失败路径

护栏机制

分流	触发条件	结果
自动校验	规则分 ≥ 85	放行
失败重试	$60 \leq \text{规则分} < 85$	重试, ≤ 2 次
人工接管	命中红线 / 得分 < 60 / 引擎不可用	转人工, 不静默放行

实测运行结果

指标	数值
完整链路运行次数（桩模式）	52 次
节点执行 / trace 事件	260 次 / 936 条
终态分布	交付 20 降级 10 转人工 22

降级 10 + 转人工 22 共 32 次，意味着异常分流与人工接管路径被实际走通——这比一条漂亮的生成结果更能说明系统在真实业务里可用。

诚实边界（这一节我不含糊）

- 链路运行是桩模式 (provider=mock) 验证：验证的是编排、降级与人工接管路径，不是真实模型的生成效果
- 评测只跑出 12 条已组装用例，不是 30 条；30 条目前是「规格」
- 对外的「一致率」目前是自洽率——人工真值列仍为空，验证表待标注
- C1 的抽取召回率是最大盲区：已从 33.8% 标定到 43.6%，精确 74.1%、加权精确仅 34.5%；抽取依赖人工维护的标记词表，无标记隐喻仍会漏

深入材料 ① 判据怎么定的

C1-C6 六条约束

编号	约束	判定方式	等级
C1	类比必须来自知识库	代码判（三档）	确认违规=一票否决
C3	不引入未经验证的数据或事实	代码判 + 人工复核	一票否决
C4	结构完整性（主线 + 6 小论点）	机器抽取 + 人工判服务度	回归检查
C5	前后话题一致，不跑题	人工 + LLM 双评	扣分
C6	口语通俗、术语必配解释	人工 + LLM 双评	扣分
C2	类比与概念结构一一对应	人工 + LLM 双评	扣分

为什么只有两条配一票否决：它们的误放代价不可逆——库外自创类比等于对外输出未验证的知识，编造事实是事故。其余是质量问题，可靠重试与人工兜底收敛。

三条铁律

- 能代码判的绝不用模型判——C1/C3/C4 走规则，保证可复现、可解释、能定向重试
- LLM 判必须先验证——人工标 10 条，一致率 ≥80% 才放量
- 主观项写 rubric，不写形容词——「类比要贴切」不可判，「覆盖概念 3 个关键属性」可判

判分器校准：实测发现并修掉的 7 个问题

#	问题	类型与危害
1	出处识别把段落标题【原理展开】当成引用	假阴性——所有无出处数字被豁免，C3 一票否决完全失效
2	数字识别把口语量词「一个 App」当成数据	假阳性——告警被淹没，久了没人看
3	严格子串匹配把类比的合规改写判成违规	字面与语义混淆——把好稿拦下
4	C4 拿「7 段骨架」当判定对象	判定对象错误——测了生产中不存在的问题
5	C3 把「时长 40 秒」「竖版 9:16」当成事实数据	判定范围错误——正常脚本判出 46 条假违规
6	知识库规模：PRD 设计值 1,284 篇 vs 实测 31 篇	口径不一致——40 倍差距会让 C1 判定失效
7	库内真实类比被判「待复核」甚至「确认违规」	整句与半句匹配错位——库里有的是拦

由此得出两条结论：一票否决项的判分器必须双向验证；判定对象与范围必须来自生产事实，不能来自设计文档的假设。

深入材料 ② 门槛该定在哪（标定实验）

判据解决「什么叫好」，这一节解决更难的一问：门槛该定在哪，凭什么这么定。

第一步：用 31 篇真实语料实测出文风基准

指标	实测中位数	为什么它是好指标
企业套话密度（赋能 / 闭环 / 抓手）	0 / 千字（均值 0.09）	语料里几乎没有——最强判别指标
过渡词密度（首先 / 其次 / 总之）	0 / 千字（p75 0.83）	一半语料为 0
短句占比（≤6 字）	20.7%（下限 12.3%）	口语节奏指标
句长标准差	19.5	长短句交错是真人特征

顺带发现一个假指标：我本想用段长方差，统计后发现语料是一句一行加空行分隔（490 行里 359 行是空行），段落长度恒等于句子长度，该指标不成立。

第二步：判据设计——为什么不能每条统一扣 5 分

第一版规则是「加分项 + 每项扣 5 分，净值 > +5 放行」。跑完场景库发现它误杀 3 个真产出。

根因：门槛只有 +5，而每条小瑕疵扣 -5，任何一条轻微瑕疵就吃掉全部门槛余量，规则退化成零瑕疵要求。一个术语没解释就被拦下——这不是质量标准，是洁癖。

据此改成三条：

- 加分改按比例（骨架 10 分按论点命中数 ÷ 6 计），取消二元判定的悬崖效应
- 扣分分三级：轻微 1 / 中等 3 / 严重 10
- 一致性乘数：文风类指标命中 ≥3 个维度时，该五项扣分 ×3——单项瑕疵只是小问题，同时命中三项以上说明整篇文风已崩

第三步：15 个场景 × 阈值扫描

阈值	误杀	漏放	一致
+5	0	4	11/15
+10	0	2	13/15
+14	0	1	14/15
+15（采用）	0	0	15/15
+16	1	0	14/15
+18	5	0	10/15

+15 是全场景库上唯一误杀 0、漏放 0 的点。代价也写清楚：满分稿只能带 7 分瑕疵，骨架齐加主线齐只有 3 分预算——这是刻意的严格。

诚实边界

- 15 个场景是构造的，不是真实产出；规则要在生产里站住，必须拿历史稿子重跑净值分布
- 加分项 B2/B3/B4 依赖 LLM 判定，需先人工标 10 条验证一致率 ≥80% 才放量

- 语料只有 31 篇且同一作者，覆盖面窄；且是公众号长文，与口播脚本文体不同，直接对比有系统性偏差

深入材料 ③ 我怎么发现并修正自己的问题

把判分器当成被测对象，反过来审它自己。结果发现 4 类问题，其中一个安全红线在特定输入下静默失效。

验证项	修复前	修复后
短稿违规是否触发否决	否（红线被旁路）	是
规则分取值数	3 个	7 个
严重度能否区分	踩 1 条 = 踩 2 条	1 条 59 / 2 条 54
C1 召回盲区	无度量	盲区可诊断，已标定 33.8%→43.6%
改写与无关的区分余量	-0.018（无法区分）	+0.288
回归测试	无	3 组

发现 1 · 安全红线被静默关闭（最严重）

判分器有个护栏：输入明显不是纯口播稿时不给分数，条件是违规比例大于 60%。但样本量很小时比例极易达到 100%——一篇 300 字正常短稿只要出现 1 个无出处数字，比例就是 100%，触发护栏，一票否决被关掉。

危害：安全机制旁路。修复方式是把置信度与违规结论解耦，并加样本量下限，固化为回归测试。

发现 2 · 召回率当时从未被测量

C1 的判定是串联两级：先抽取候选，再判定。我测的一致率只覆盖第二级。端到端 = 抽取召回率 × 判定准确率——抽取只捞到 43.6%，判定再准也管不了漏掉的。

- 已标定：抽取召回率 33.8% → 43.6%，区间 30.0%–60.7%
- 精确率 74.1%，但加权精确率只有 34.5%——这个难看的数字我主动留着，它比好看的召回率更能说明抽取层是瓶颈
- 仍有 7 句未解决

发现 3 · 规则分退化成二值，没法做版本对比

扣分逻辑只对 C2/C4/C5/C6 生效，但 C2/C5 压根没被调用，C6 当时没有计数字段，于是恒不扣分。结果规则分只有 59、90、100 三个取值，无法支撑用于版本对比的设计用途。

发现 4 · 换词不换义，字符比对区分不了

字符二元组 Jaccard 实测：轻度改写 0.74、核心片段 0.53、中度改写 0.29、重度改写 0.27、无关类比 0.00–0.02。改用语义匹配后，改写与无关的区分余量从 -0.018 提升到 +0.288。

深入材料 ④ 真实用起来会出什么问题

这套系统我自己连续用了几个月做内容生产。这一节不写它能跑，写它一直没修好的三个问题，以及它们分别对应评测体系的哪个盲区。

现象	性质	对应盲区
稿件合格率低，时好时坏	指标结构看不见	判定层准不准，不影响抽取层漏掉的
库外比喻听感很奇怪	已知盲区的线上表现	C1 抽取召回只有 43.6%
文风过于 AI 味	判据维度缺失	C1-C6 没有一条管文风

问题一 · 合格率低，时好时坏

一开始我以为是模型发挥不稳定。归因后发现：链路两级串联，我一直盯着的一致率只测了第二级。端到端 = 抽取召回率 × 判定准确率——抽取只捞到 43.6%，漏掉的 56.4% 根本不进入判定。所以时好时坏不是模型随机，是抽取层的漏抽在随机发生。

- 新增召回诊断输出：覆盖度、含标记未参与判定的句子、无标记但形态像类比的句子
- 补正则覆盖无标记隐喻，召回 33.8% → 43.6%

残留：没有根治。抽取依赖人工维护的标记词表，无标记隐喻天生抓不到，且漏掉的不进入任何指标的分母。

问题二 · 库外比喻听感奇怪

这正是 C1 要拦的东西，但它漏了。三个原因叠加：

原因	说明
抽取召回只有 43.6%	库外自创的比喻没被抓进判定环节，直接被当正常内容放行
语义阈值 0.55 偏向不漏放	为避免误拦合规改写而调低阈值，导致假阳性增加
知识库只有 31 篇示范语料	完整语料在另一台设备；库越小，库内与库外的边界越模糊

关键点：我在写分层说明文档时已经推导出这个盲区会在线上表现成假阳性，真实使用反馈把它验证了。预测与实测对上，比事后解释有力得多。

问题三 · 文风带 AI 味

稿子规则上都过，但读起来有股机器味——套话多、句式整齐、没有口语的错落感。翻遍 C1-C6：C1 管类比来源、C3 管事实、C4 管结构、C6 管口语通俗与术语解释，没有一条管文风像不像人写的。这不是 bug，是判据维度缺失。

补法见「门槛标定实验」：用 31 篇语料实测出 4 条可代码判的风味指标，再加一致性乘数，构成 C6 的参照层。

残留：规则层只能抓表面特征（套话、过渡词、句长分布），抓不到信息密度低、每段都在铺垫这类结构性 AI 味，那部分要靠参照层与人工。

这一节的诚实边界

- 使用者是我自己，没有外部用户。它证明的是真实长期使用与迭代，不是有他人采纳
- 这些是定性反馈，没有日志埋点，合格率低没有精确失败率数字

深入材料 ⑤ 成本与延迟预算（文字链路）

AI PM 必须算得清成本。这一节把所有数字摊开，包括三处与设计文档不一致的实测发现。

实测调用结构

依据 n8n WF1 工作流 JSON（16 节点）逐节点核对，只有 2 个节点调用 LLM：

环节	每轮次数	输入 tokens	输出 tokens
选题生成	1	140	约 400
脚本生成	5	183	约 600
合计	6 次	1,055	3,400（4,455 tokens / 轮）

三处与设计文档不一致（实测发现）

PRD 里的假设	工作流实测	影响
有动态评分与自动重试 ≤2 次	WF1 没有评分与重试节点	成本模型不含重试倍数
走 RAG 召回知识库原文	脚本节点只注入选题四字段	无检索 token 成本
脚本可能按段多次调用	每选题恰好 1 次	调用次数可确定

单篇成本

模型	输入（元/百万）	输出（元/百万）	单篇
Doubao 1.5 Lite 32k	0.3	0.6	¥0.0005
Doubao 1.5 Pro 32k	0.8	2.0	¥0.0015
Doubao Seed 1.6 Flash	0.15	1.5	¥0.0011
DeepSeek flash（低谷）	1.07	4.26	¥0.0031

结论：文字链路成本可忽略——单篇约 ¥0.0015，日更 10 篇月成本约 ¥0.46。

延迟结构：与成本结构相反

LLM 只占 16 个节点中的少数，人工审批（Wait）才是耗时主体。所以所谓周期压缩 70%，压缩的是人工等待与返工，不是机器耗时。

诚实边界：单位成本是实测用量乘公开单价的推算，不是实际账单数字；端到端耗时与人工审批占比尚未实测。

附 · 我的贡献边界与 AI 使用说明

为什么要写这一页：作品集完整度较高，面试官自然会问这些是不是你做的。与其被追问，不如先把自己的边界说清楚。本文只讲事实，不讲感受。

总览：我做了哪些判断，工具做了哪些执行

类别	我做的	工具或 AI 做的
问题定义	判断什么问题值得做、边界在哪	—
方案取舍	选哪条路、放弃哪条、为什么	—
标准制定	什么叫好、什么叫不合格、什么算通过	—
判据设计	哪些能机器判、哪些必须人判、阈值定多少	—
代码与文档产出	逻辑与结构	排版、润色、代码初稿
素材生成	提示词策略、验收标准	图像与视频的实际生成
工程脚手架	业务逻辑、页面、数据模型	项目模板与 UI 组件库

我是怎么用 AI 的（方法论，不是辩解）

用 AI 做什么	不用 AI 做什么
生成文档初稿与排版	定义问题与非目标
写代码初稿、调试报错	决定判据与阈值
生成图像与视频素材	判断素材能不能用、什么算通过
整理与归纳信息	做取舍、决定放弃什么

一条原则：AI 可以把执行做得很快，但什么叫好必须我自己定义。所以我在这个项目里花时间最多的地方不是写代码，而是把内容合格拆成 6 条可判定约束、想清楚哪两条该设一票否决、反复验证判分器在该拦的和不该拦的两个方向上是否都准。

几个我主动否决的方案（这比做了什么更能说明判断）

- 让模型自由打分——否决。模型自评会漂移、不可复现。改为能代码判的绝不用模型判
- v1 直接上多智能体——否决。生产未稳时前后置智能体没有可靠输入，只会放大不确定性与成本
- C4 拿 7 段骨架当判定对象——已否决并修正。真实结构是一条主线加 6 个小论点，且生产中根本没有缺段问题
- 发布环节全自动——否决。批量秒发会触发平台风控，改为人工错峰发布
- 用大动作镜头做 AI 口播——否决。动作越小越像真人，大转头大挥手最容易暴露 AI

诚实边界

项	说明
本文不含任何虚构数据	所有无法验证项均如实标注
表格为自评	分栏依据是实际记忆与文件证据，欢迎对照作品集核验

会持续更新	随着跑分补齐与素材迁移，无法验证栏会逐项减少
-------	------------------------

全份材料的诚实边界（汇总）

- 评测只跑出 12 条已组装用例，不是 30 条；对外的一致率目前是自治率，不是人工一致率
- C1 抽取召回率是最大盲区：33.8% → 43.6%，加权精确仅 34.5%，仍有 7 句未解决
- 多智能体链路是桩模式验证：52 次运行验证编排、降级与人工接管，不是真实模型生成效果
- 知识库语料是 31 篇示范样本，完整语料在另一台设备；C1 判定有效性以类比库完整度为前提
- 数字人流水线的后两段（配音、发布）仍是规划，WF1 已跑通
- 成本为实测用量乘公开单价的推算，非实际账单
- 相关性不等于因果：+0.546 与 +0.099 是 57 条作品的相关系数，样本量小

以上所有内容均可在对应材料中核验 · 不含无法溯源的数字