



QQ 游戏开放生态 · 作品 Demo

AI 频道智能匹配

理解玩家需求，匹配更合适的游戏频道 —— 产品说明文档

- 意图理解
- 可解释匹配
- AI Discovery
- Bad Case 闭环

文档版本 v2 · 纯前端离线 Demo · 零依赖可运行
定位：面向「QQ 游戏开放生态产品经理」岗位，演示一个 AI 产品的最小完整闭环

QQ 游戏开放生态中沉淀了大量游戏频道 —— 新手组队、攻略讨论、开黑大厅、BUG 反馈等。这些频道承载了玩家的交流与组队需求，但存在一个明显的供需错配：**频道很多，玩家却常常不知道「我该去哪个频道」**。

Demo 定位：验证一个 AI 产品方案 —— 让玩家用一句自然语言描述需求，AI 自动理解意图、生成结构化画像，并推荐最合适的频道。整个 Demo 不追求算法复杂度，而是完整跑通「问题定义 → 模型链路 → 评估 → 数据反馈 → Bad Case → 迭代」的 **AI 产品闭环**，而非单纯调用一次大模型。

一句话价值主张

一句话输入 → 一次点击 → 得到一个「可解释」的频道推荐

Demo 的三个关键特征

- **零门槛：**纯前端静态页面，双击 **index.html** 即可运行，无需服务器、无需依赖、无需 API Key。
- **双层架构：**模型「理解层」可替换，产品「匹配决策层」稳定，换模型不改匹配逻辑。
- **真实反馈闭环：**用户 👍 / 🗣️ 反馈进入 Bad Case 池，并真实地改变下一次推荐结果。

玩家侧

想找队友、求攻略、问搭配，却要面对几十个频道；不知道该进哪个，发帖石沉大海，需求无法被及时响应。

生态侧

频道流量分配粗放，优质内容与目标用户错配；
频道活跃率受损，长尾内容得不到有效分发。

更深一层的问题：频道库的「覆盖」与「供给」缺口

问题类型	具体表现	Demo 中的应对
有频道，找不到	玩家不会表达 / 频道命名无法被理解，不知道该进哪个	自然语言意图理解 + 可解释推荐
有频道，不匹配	游戏有频道，但主题覆盖不足（如异环只有讨论/攻略/BUG，缺「新手互助」）	AI Discovery 触发频道扩充
没频道	长尾 / 新游戏在频道库中完全缺失（如「幻塔」）	AI Discovery 从公开信息生成频道画像

核心思路：把「选频道」从**用户自己翻找**，变成**AI 理解需求后主动推荐**，并且每一步都可解释、可反馈、可迭代。

四大能力模块

① 结构化意图理解

从一句话中提取**游戏 / 意图 / 阶段 / 标签**四要素，形成结构化用户画像，而非笼统的文本匹配。

② 两阶段可解释匹配

先按游戏实体**硬过滤**，再按「意图 35% + 标签 30% + 阶段 15% + 主题 20%」加权排序，逐项给出匹配依据。

③ AI Discovery 频道扩充

频道覆盖不足或游戏缺失时，自动发起 Web 检索 → 结果过滤 → **生成频道画像** → 人工审核后入库，不推荐其他游戏的频道。

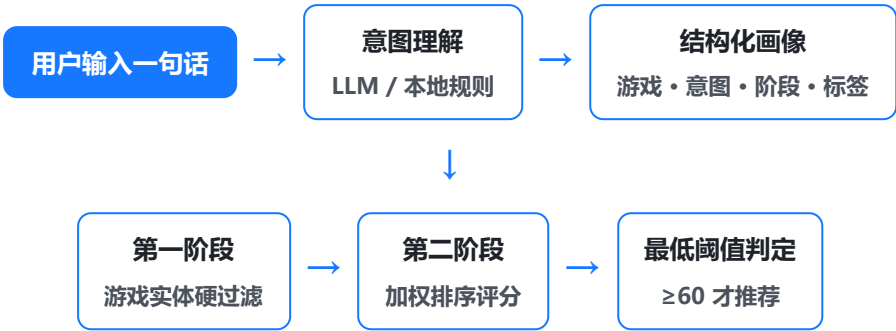
④ Bad Case 反馈闭环

👍 / 🗣️ 反馈落库；「推荐不准确」会生成**排除 / 降权规则**，直接影响下一次推荐，形成数据飞轮的雏形。

可演示场景（覆盖四种产品路径）

场景	输入示例	预期行为
A • 本地高匹配	刚玩逆水寒手游，想找几个新手一起打副本。	直接推荐「逆水寒手游 新手组队」约 90%，日志提示「无需扩充」
B • 主题覆盖不足	异环有没有新手群？刚入坑求带	本地最高约 28% → 触发 Discovery → 生成「异环 新手互助」（待审核）→ 重新匹配约 90%
C • 游戏不存在	最近在玩一个叫幻塔的游戏，想交流一下	本地候选 0 → Discovery 生成「幻塔 玩家讨论区」→ 推荐约 75%
兜底 • 拒绝生成	XX 是什么游戏？	公开信息不足 → 明确拒绝生成频道画像（不编造，避免 AI 幻觉）

主流程



分支判定（三种结局）



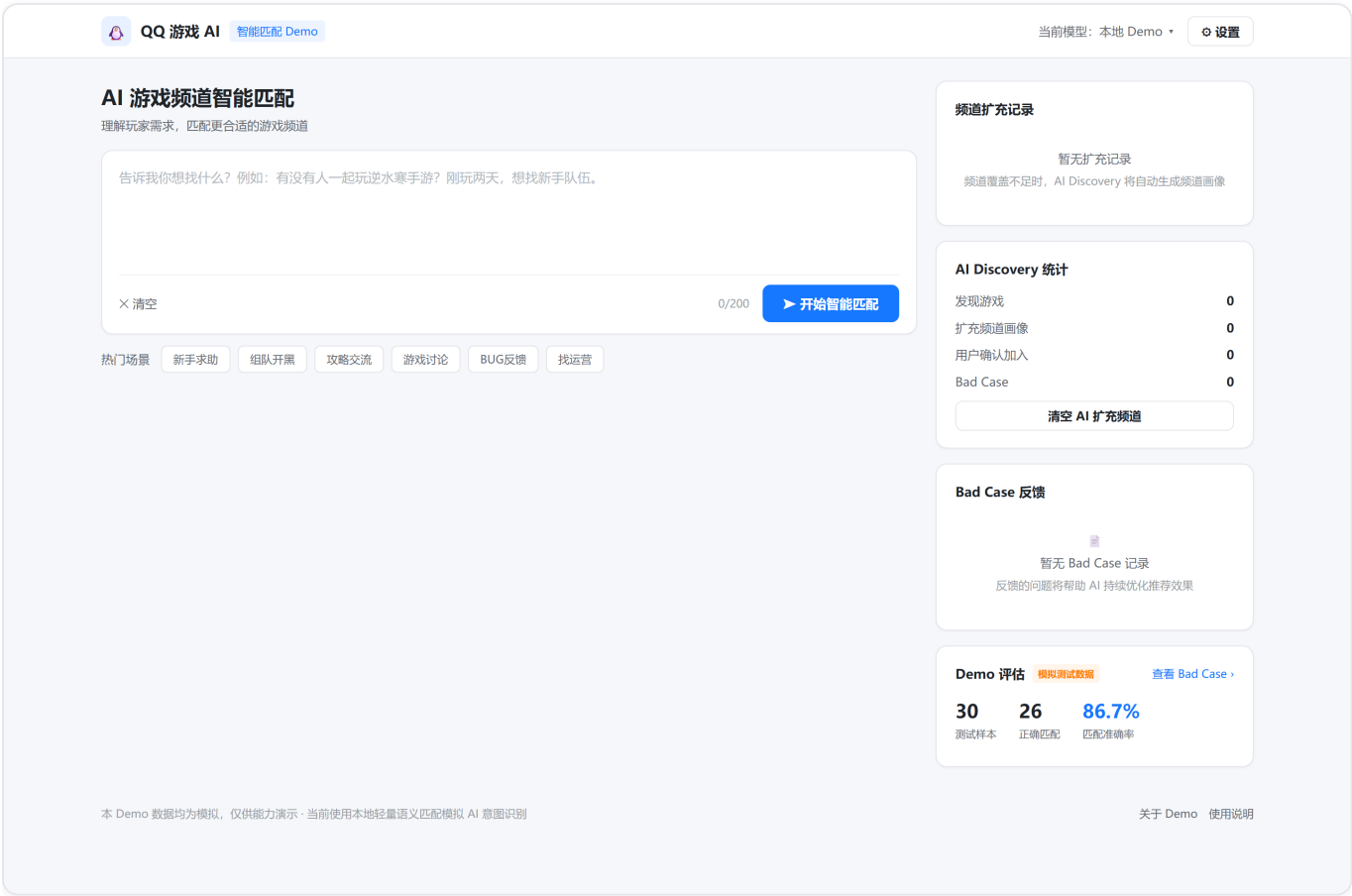
AI Discovery 子流程



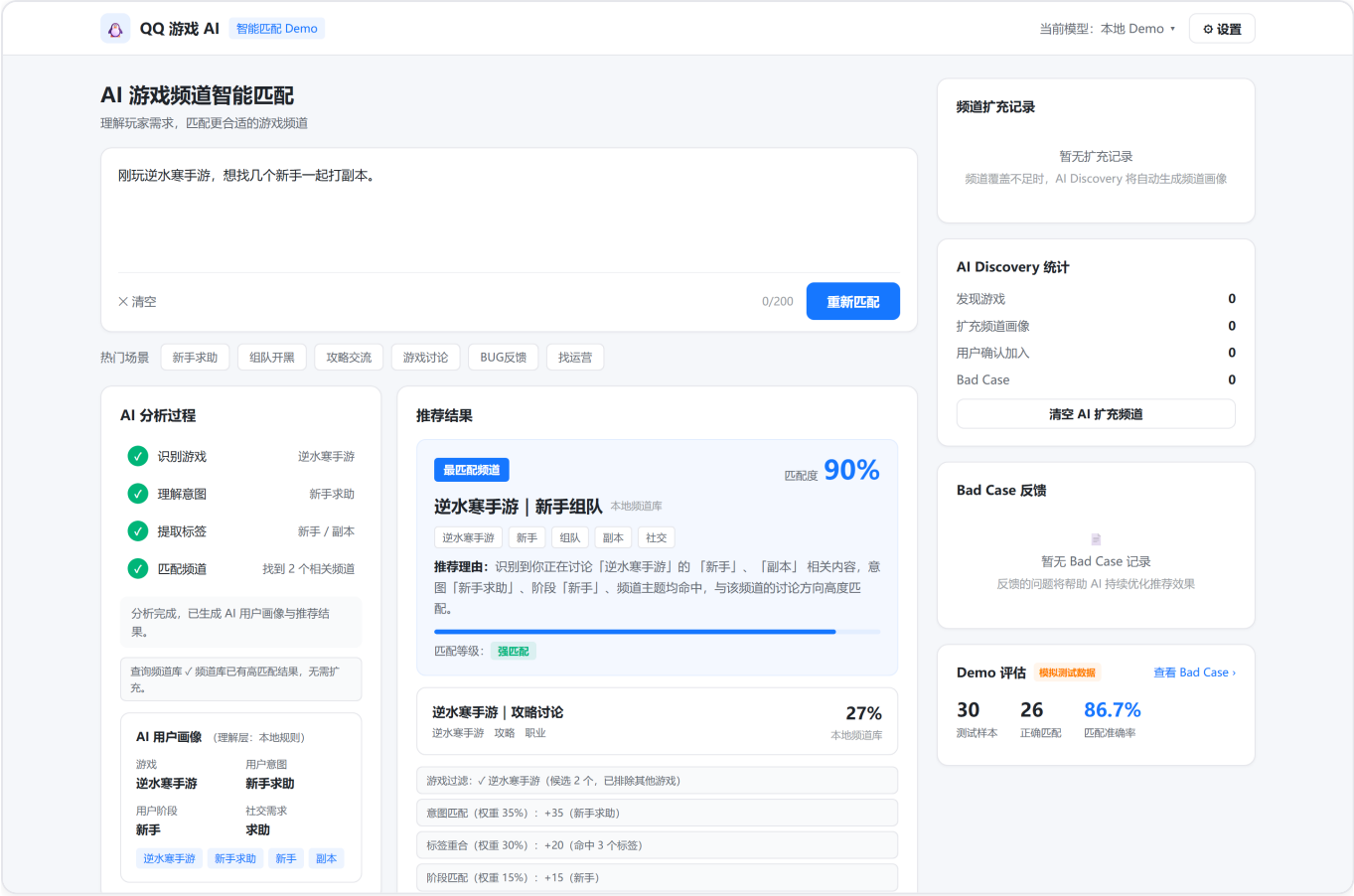
反馈闭环



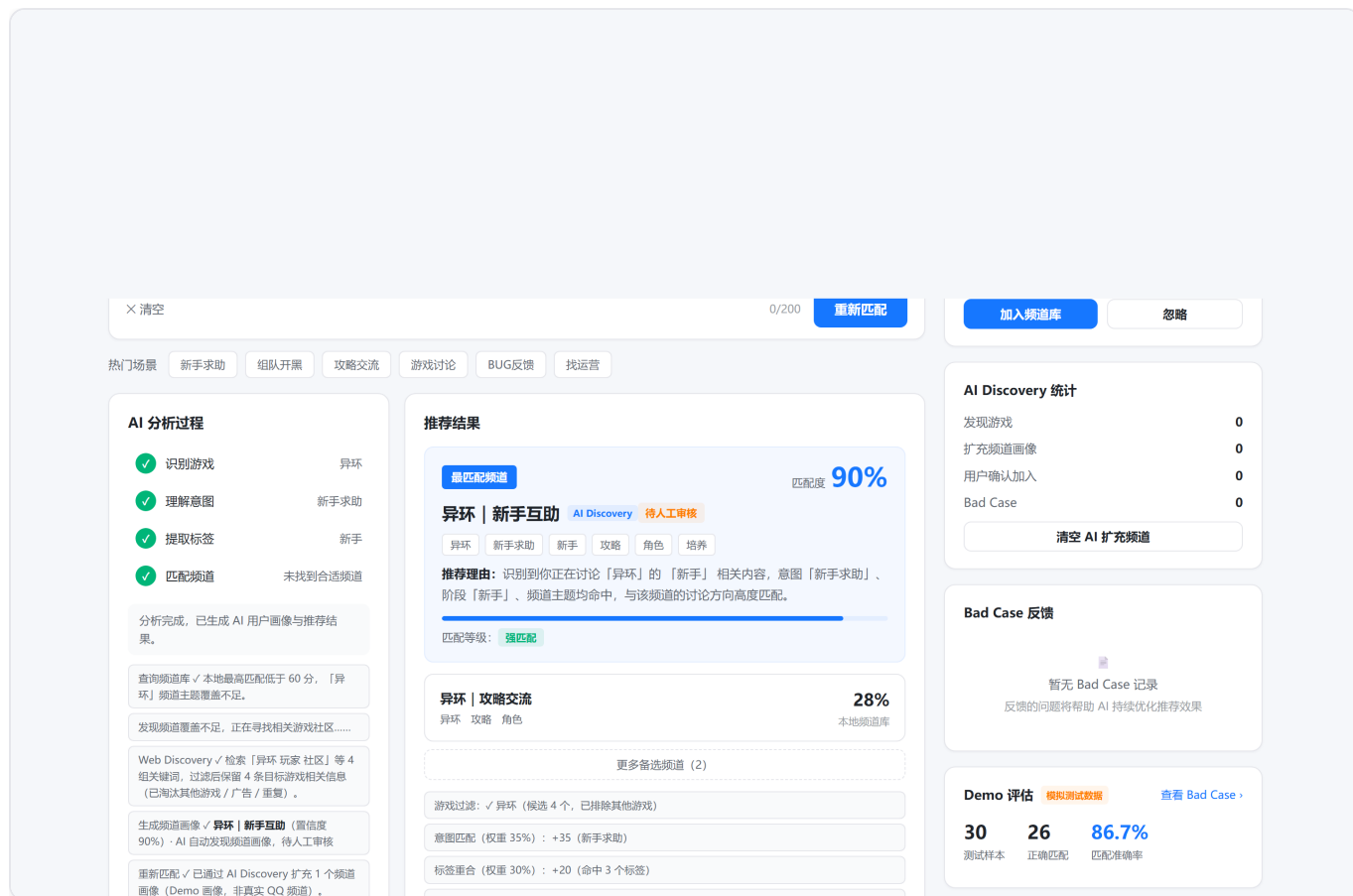
以下为 Demo 实际运行界面（本地模式，双栏布局：左侧「输入 + 分析 + 推荐」，右侧「运营面板」）。



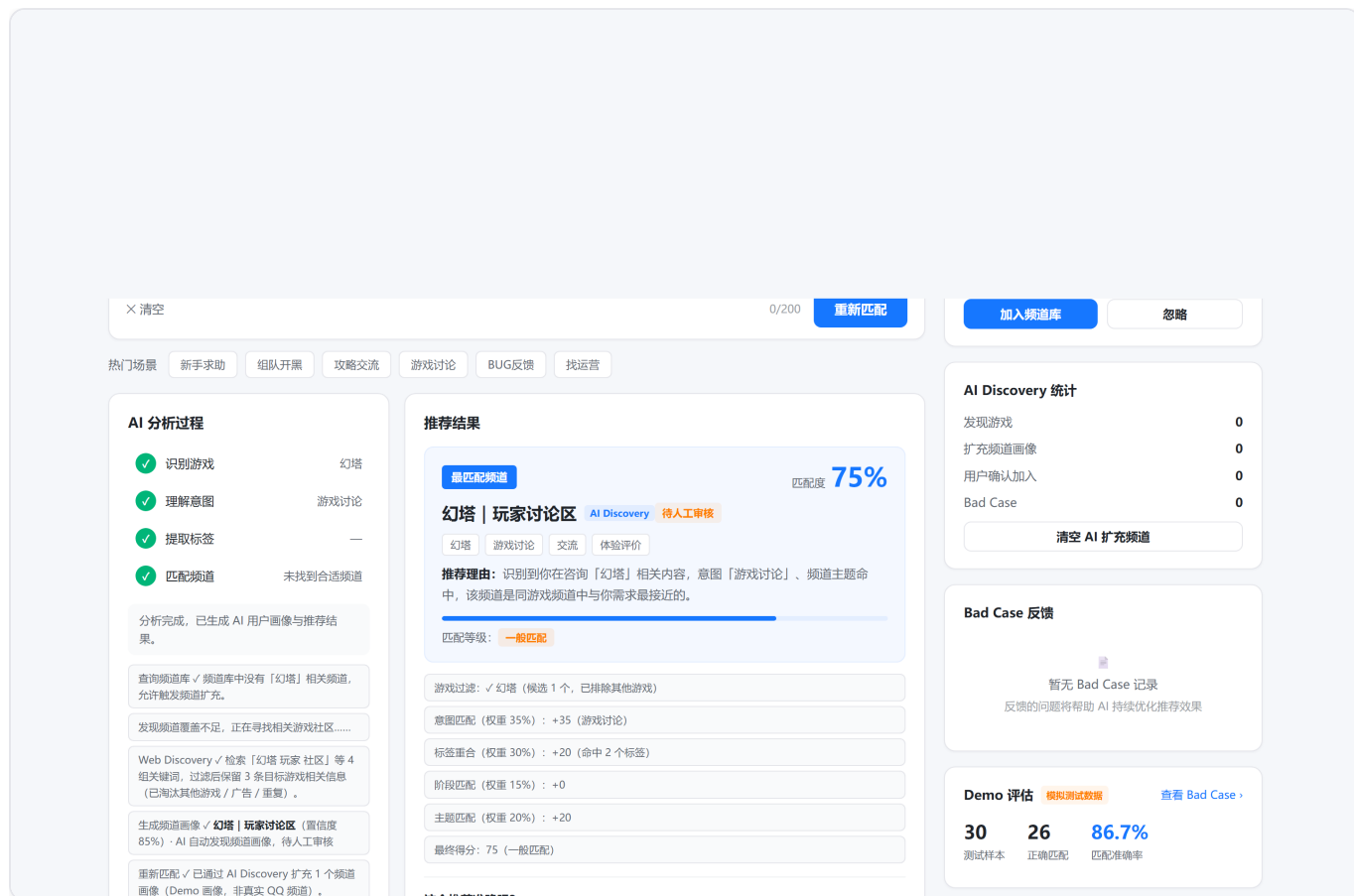
① 初始界面 —— 顶部导航 + 输入框 + 热门场景快捷入口；右侧为「频道扩充记录 / AI Discovery 统计 / Bad Case 反馈 / Demo 评估」运营面板。



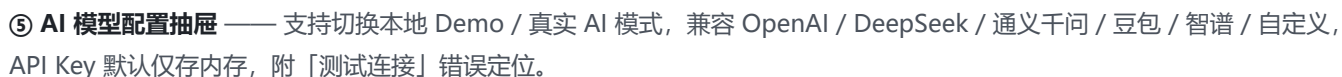
② 场景 A • 本地高匹配 —— 输入「刚玩逆水寒手游...」，AI 识别出「游戏=逆水寒手游 / 意图=新手求助 / 阶段=新手」，直接推荐「逆水寒手游 | 新手组队」约 90%，并逐项展示匹配依据。



③ 场景 B • 覆盖不足触发 Discovery —— 输入「异环有没有新手群？刚入坑求带」，本地异环频道最高仅约 28%，自动触发频道扩充，生成「异环 | 新手互助」（置信度 90%·待人工审核）后重新匹配到约 90%。



④ 场景 C • 游戏不存在触发 Discovery —— 输入「最近在玩一个叫幻塔的游戏」，频道库无幻塔频道，AI Discovery 从公开信息生成「幻塔 | 玩家讨论区」并推荐。



Demo 采用**双层架构**，核心原则是「**模型层可替换，产品决策层稳定**」：

用户输入 → LLM 理解层 → 结构化用户意图 → 本地频道匹配策略 → 推荐结果 → 用户反馈 → Bad Case

理解层（可替换）

模式	能力	特点
本地 Demo 模式 (默认)	游戏别名标准化 + 11 类意图触发词 + 阶段规则 + 标签词表	离线可运行，无需 API Key
真实 AI 模式	模型只输出结构化 JSON（游戏 / 意图 / 阶段 / 标签）， 不直接生成推荐	统一接口 <code>callLLM()</code> ，换模型不改匹配逻辑

匹配层（稳定 · 两阶段可解释匹配）

阶段	动作	说明
第一阶段	游戏实体硬过滤	识别出 game=异环 时，候选只能是异环频道；跨游戏频道禁止进入排序；叠加 Bad Case 排除规则
第二阶段	候选排序 (0~100)	按加权得分排序，低于 60 不推荐；未识别游戏时要求 ≥80

评分维度	权重	判定方式
意图匹配	35%	频道标签命中意图核心标签（11 类意图各有映射）
标签重合	30%	用户标签与频道标签的 Dice 重合系数
用户阶段匹配	15%	频道标签命中阶段标签（新手 / 进阶 / 长期玩家倾向）
频道主题匹配	20%	频道名「 」后缀主题与意图的亲密度

80~100

强匹配

60~79

一般匹配

<60

不推荐

安全与兜底设计

- **API Key 安全**：默认仅存页面内存，勾选「记住配置」才写入 localStorage，并有明确提示。
- **测试连接**：真实请求一次并校验 JSON 结构，失败给出可定位原因（401=Key 错 / 404=Base URL 错 / 400=模型名错 / 跨域）。
- **任何情况兜底**：无 Key / 配置不全 / API 失败 / 网络异常时，自动回退本地模式，Demo 永远可完整演示。
- **拒绝幻觉**：AI Discovery 公开信息不足时明确拒绝生成，不编造频道。

Bad Case 不是「展示用的列表」，而是**真实参与优化**的反馈机制——它改变下一次推荐。

反馈 → 策略联动

反馈原因	对本地频道的影响	对 AI Discovery 频道的影响
游戏识别错误	该游戏下被标记频道 直接排除	按次 降权 -15 分/次 ；≥2 次标记「低置信频道」并降低优先级
推荐频道 不匹配	该游戏下被标记频道 直接排除	
意图识别错误 / 其他	记录为 Bad Case，用于后续模型与匹配策略调优	

闭环链路



数据与评估

模拟评估数据（已标注）

测试样本 30 · 正确匹配 26 · 准确率 86.7%

游戏识别错误 1 · 意图识别错误 1 · 频道匹配错误 2

本机真实反馈

👍 / 🗨️ 反馈与 Bad Case 记录持久化于浏览器 localStorage，刷新不丢失，并在「Bad Case 反馈」面板与「本机累计反馈」中实时展示。

版本	方向	内容
V2	接入真实大模型 API	当前已内置配置支持；后续优化 Prompt、增加 few-shot 示例、按 Bad Case 类型定向调优
V3	建立真实频道画像	引入频道主题 / 活跃度 / 用户画像 / 历史匹配效果，匹配从「标签重合」升级为「供需匹配」
V4	建立数据飞轮	打通「用户反馈 → Bad Case → 模型评估 → Prompt 优化 → 匹配策略优化」，匹配准确率随数据积累持续提升

最终闭环

用户需求 → AI 理解 → 频道匹配 → 用户反馈 → Bad Case → 策略迭代 → （数据回流）→ 更准的推荐

QQ 游戏 AI · 频道智能匹配 Demo —— 产品说明文档
纯前端离线 Demo · 模拟数据仅供能力演示 · 非真实线上数据