**腾讯云智能体开发平台**

**产品技术白皮书**

深圳市腾讯计算机系统有限公司

# 1 概述

腾讯云智能体开发平台（Tencent Cloud Agent Development Platform），是基于大模型的智能体构建平台。提供LLM+RAG、Workflow、Multi-agent等多种智能体开发框架，助力企业能够结合专属数据，更高效地搭建稳定、安全、符合业务需求的智能体。

本文档适用于「腾讯云智能体开发平台」产品的目标用户，以供相关交付、使用、维护和管理人员较 为全面地了解腾讯云智能体开发平台的整体产品和技术情况，可作为售前技术交流，及项目售中售后的技术参考资料。

本文档适用于腾讯云智能体开发平台2.9.5私有化版本，所有软件功能的实现细节以最终交付的产品为准。

未经许可，不得编辑、转载、节选或分享本文档的内容。

# 2 背景介绍与应用场景

在数字化转型的浪潮中，企业客户的需求和期望正在发生显著变化。从行业趋势来看，企业已经从单纯追求效率分流，转变为更加注重运营价值和服务体验，寻求降本增效与客户体验之间的平衡。这一转变不仅对企业的内部运营提出了新的挑战，也为企业提供了新的发展机遇。

AIGC（人工智能生成内容）技术，特别是大语言模型的应用，为满足这些新需求提供了强有力的支持。它不仅在C端（消费者端）提升了对话体验，也在B端（企业端）显著提高了运营效率，为企业带来了前所未有的想象空间。

“构建一个离产业最近的AI平台”是腾讯云的核心主张之一。在产业智能化转型的浪潮中，“构建离产业最近的AI平台”不仅是技术能力的比拼，更是对产业痛点的深度理解和价值闭环能力的考验。

面对企业智能体开发门槛高的挑战与问题，腾讯云将大模型和工具进行产品化，提供完整的平台开发框架及端到端服务链路，**助力企业改进沟通交互体验，增强服务能力，提升经营效率。**

○RAG（检索增强生成）：支持行业最全面的知识类型导入，在复杂图文混合排版的文档解析和文档问答方面保持行业领先的水准。

○工作流编排：除了业内共性的低代码拖拽方式，让企业可以轻松编排业务流程，实现标准化、稳定的操作；在实战中打磨了「工作流结合agent」的创新方案，支持智能的节点回退，对话响应更灵活。

○Agent设置： 首发支持零代码配置多智能体（Multi-Agent），并集成腾讯特色MCP插件及三方插件，如腾讯医典、腾讯地图、文档格式转换等，满足企业多样化需求。

**主要的应用场景体现在如下几个场景：**

**智能客服**

基于腾讯云行业大模型，通过腾讯云智能体开发平台，提供垂直领域的专业化智能客服能力，助力企业及开发者打造场景丰富、功能完善的智能客服解决方案，实现知识生产效率、问题覆盖率、回答准确率跨越式提升。

**企业专家问答助手**

支持企业级知识库搜索和智能问答，对知识库多模态知识，进行高效处理、分析、检索、问答，助力企业及开发者实现企业级知识库数智化服务升级。

**车载语音助手**

支持图文混排等复杂汽车操作手册、产品说明书的快速解析及检索，结合行业大模型生成拟人化答案，实现更加拟人化、个性化的车内人机交互。

**Agent模式-经营分析助手**

支持根据用户的数据查询或图表生成指令查询检索知识库中的经营数据，并调用代码解释器和科学计算等工具高效生成基于用户数据的专业经营图表。

# 3 产品框架与优势

24年，腾讯云智能体开发平台，在原有的LLM+RAG应用框架基础上，升级为基于大模型的应用开发平台，旨在通过LLM+RAG 、Workflow、 Multi-agent等多种应用框架以及原子能力（API），促进高效的大模型应用构建。产品着重于支持客户重塑和升级其企业级知识服务系统。





# 4 功能清单

#### [【新版】腾讯云智能体开发平台功能清单（公有云&私有化）](https://doc.weixin.qq.com/sheet/e3_AZAAYwY5AOcCNGj9L0ERmT6arZbfr?scode=AJEAIQdfAAoDTMw4KhAKIAOwZ8ACo&tab=acs0ic)

# 5 核心功能与优势

## 5.1 模型层

腾讯云智能体开发平台提供多种模型选项，您可根据您的需求进行选择。

平台按照模型用途，将模型分为两大类：

1. **思考模型**：用于意图识别，主要影响意图识别的效果。

2. **生成模型**：主要影响阅读理解与答案生成的效果。

腾讯云智能体开发平台目前已接入精调知识大模型、混元大模型、行业大模型、DeepSeek 等十余种模型，各模型详情及适用场景如下：

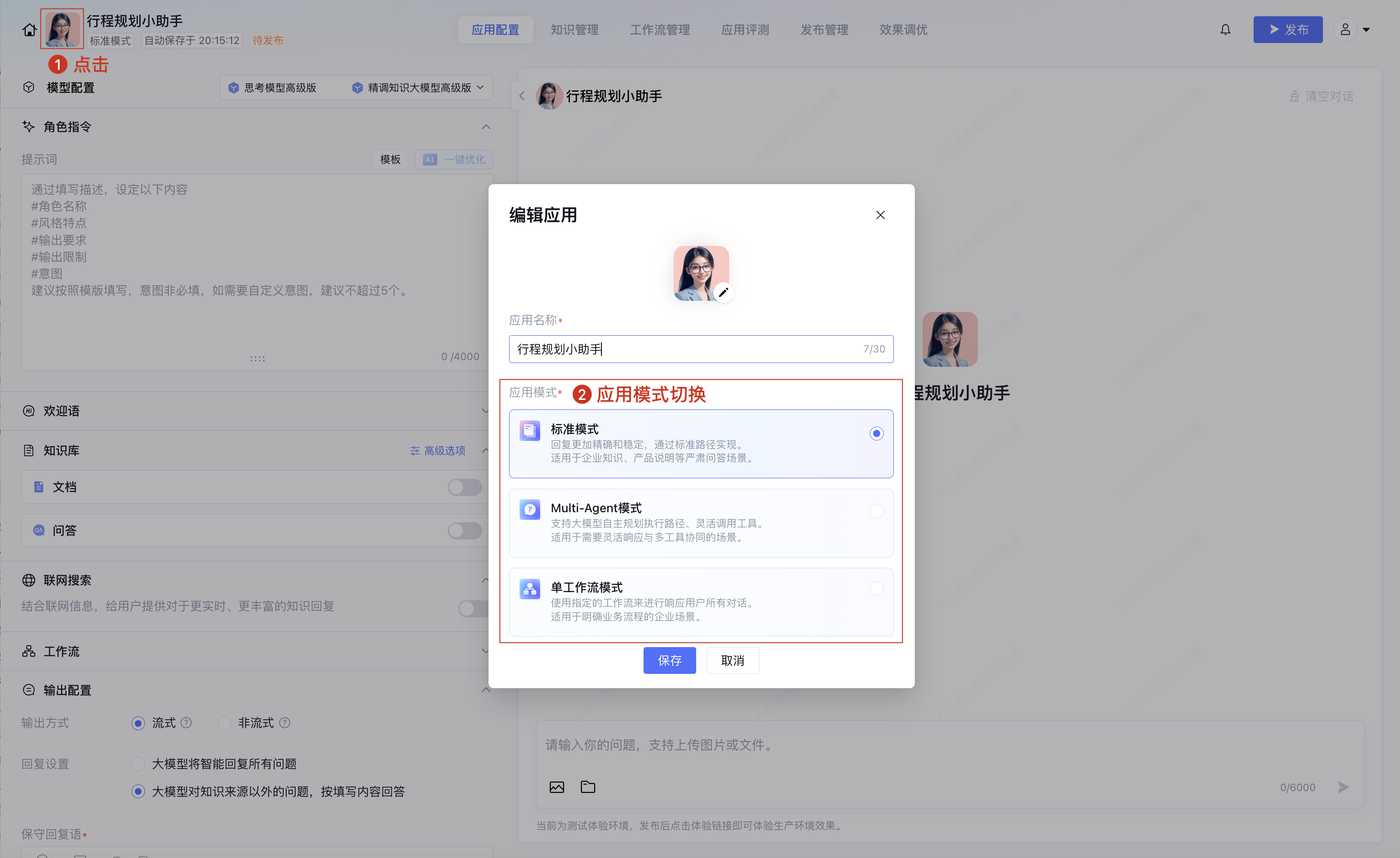
[腾讯云智能体开发平台 模型介绍\_腾讯云](https://cloud.tencent.com/document/product/1759/112876)

其中，开箱即用的行业大模型，利用无监督数据二次训练行业大模型； 利用行业模型挖掘领域数据中的知识；人工对知识进行审核，确保正确合规；利用人工审核过的数据，调优问答模型；通过行业LLM扩大了知识边界，在“预训练”阶段积累海量的行业知识，帮助企业开箱即用。另一方面，基于行业大模型之上，提供企业专有模型的Fine Tune。

## 5.2 多种应用框架

在腾讯云智能体开发平台应用构建平台，支持无代码基础的企业运营人员基于不同应用模式，快速配置大模型应用，简单修改配置项后即可发布应用。





### 标准模式

平台内置最佳实践流程，只需导入文档/问答对，即可即可让大模型对接企业知识，达到更稳定和精确的知识问答效果。适用于企业知识服务、产品咨询等严肃问答场景。

面向企业服务严谨场景，提供自研RAG算法和最佳实践，同时提供问答对生成/校验、保守回复等多种运营工具，适用于严肃严谨场景，如政务服务、政策咨询、智能客服等。

### 单工作流模式

单工作流模式是一种以 统一业务流程 为核心的工作流设计范式，适用于业务目标明确、流程固定且需高度标准化的场景（如审批、订单处理、客户服务等）。其核心在于通过 单一流程实例 覆盖所有用户交互与任务执行，确保业务逻辑的集中管理和执行一致性。

* 规则内置化 ：业务逻辑（如条件判断、权限控制）通过预置代码或配置实现，避免运行时动态决策。
* 业务标准化 ：强制统一执行路径，减少人为操作差异，适用于合规性要求高的场景（如金融风控）。

### Multi-Agent模式

Agent模式由大模型进行任务自主规划和工具调用（function-call）。根据用户的输入，由模型来进行任务的拆解和主动工具调用，可实现高效应用搭建，适用于有灵活回复或快速搭建需求的服务问答场景。

Agent模式的核心优势是由大模型进行自主规划和工具调用，其核心工作流程包括：

* 思考规划：制定整体任务达成的规划思路，并把整体的复杂任务拆解为细分的子任务。
* 主动选择和调用工具：根据拆解后的子任务和工具的描述说明，选择一个或多个合适的工具来解决问题。
* 主动纠错反思：模型自主改进优化过去的行为决策，并对行为进行纠正

以上三类工作由思考模型执行完成，并可能存在多次循环执行，最终输出回复的答案。



Agent模式现有代码解释器、知识库问答和天气等6种可用工具，助力用户实现图表生成分析、知识库文档检索和天气数据查询分析等功能，扩大大模型应用能力边界的同时支持提升交互操作的简便性。

## 5.3 产品功能概述

## 可视化构建Agent界面示例 - 主界面



## 可视化构建Agent界面示例 - 知识库

知识库管理：

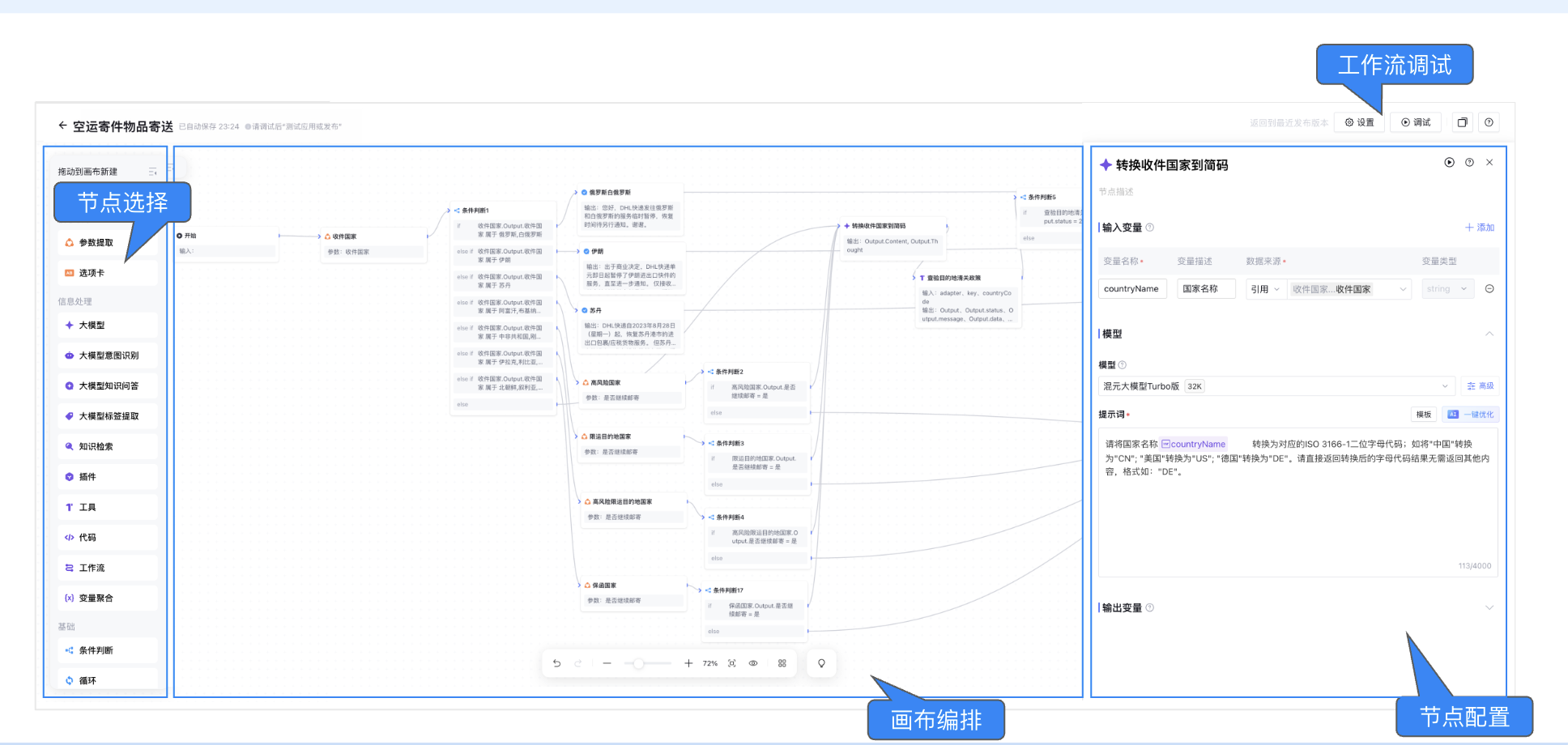
支持多类别知识上传，文档、网页、PPT、表格、图片等，同时包括支持问答对和数据库对接

格式类型pdf、doc、docx、ppt、mhtml、pptx、wps、ppsx、xlsx、xls、md、txt、csv、html、jpg、png、jpeg、tiff、bmp、gif



## 可视化构建Agent界面示例 - 工作流

工作流：提供可视化低代码编排能力，可在画布中拖拽节点，快速搭建标准、稳定的执行流程。



## 可视化构建Agent界面示例 - Multi-Agent

Multi-Agent：提供可视化配置界面，支持添加多个Agent并定义其间的转交逻辑，实现高效协同。

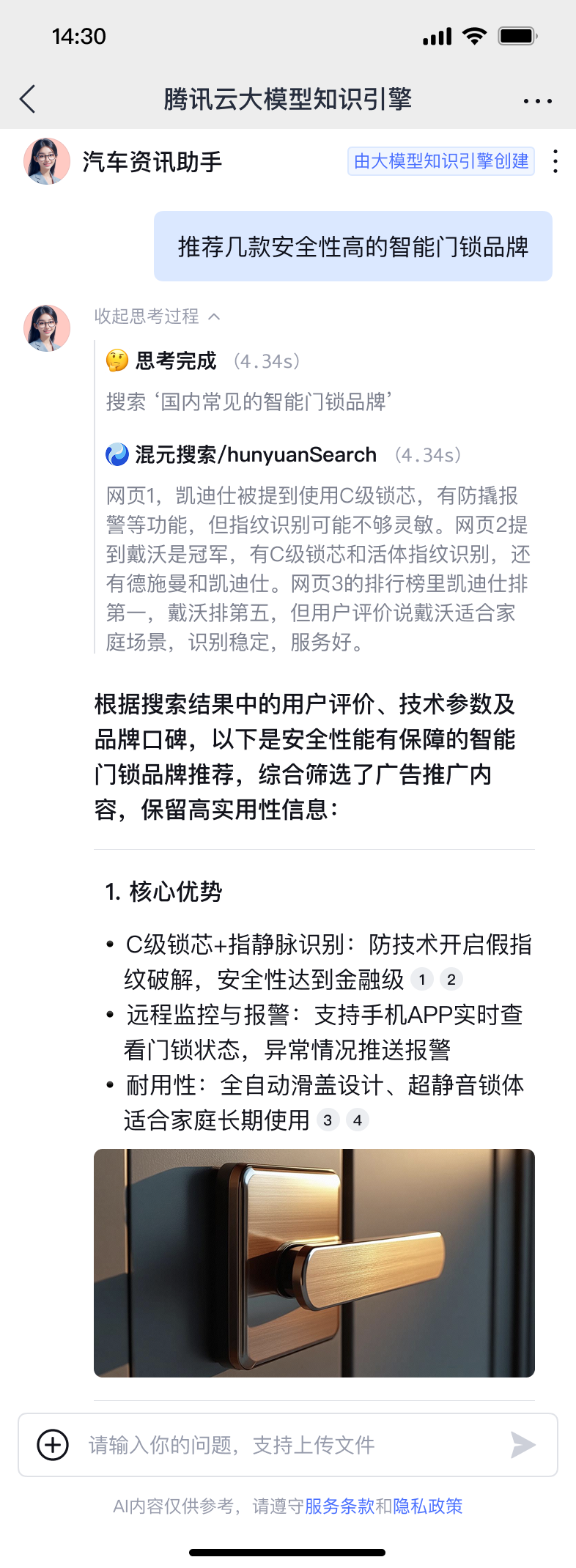
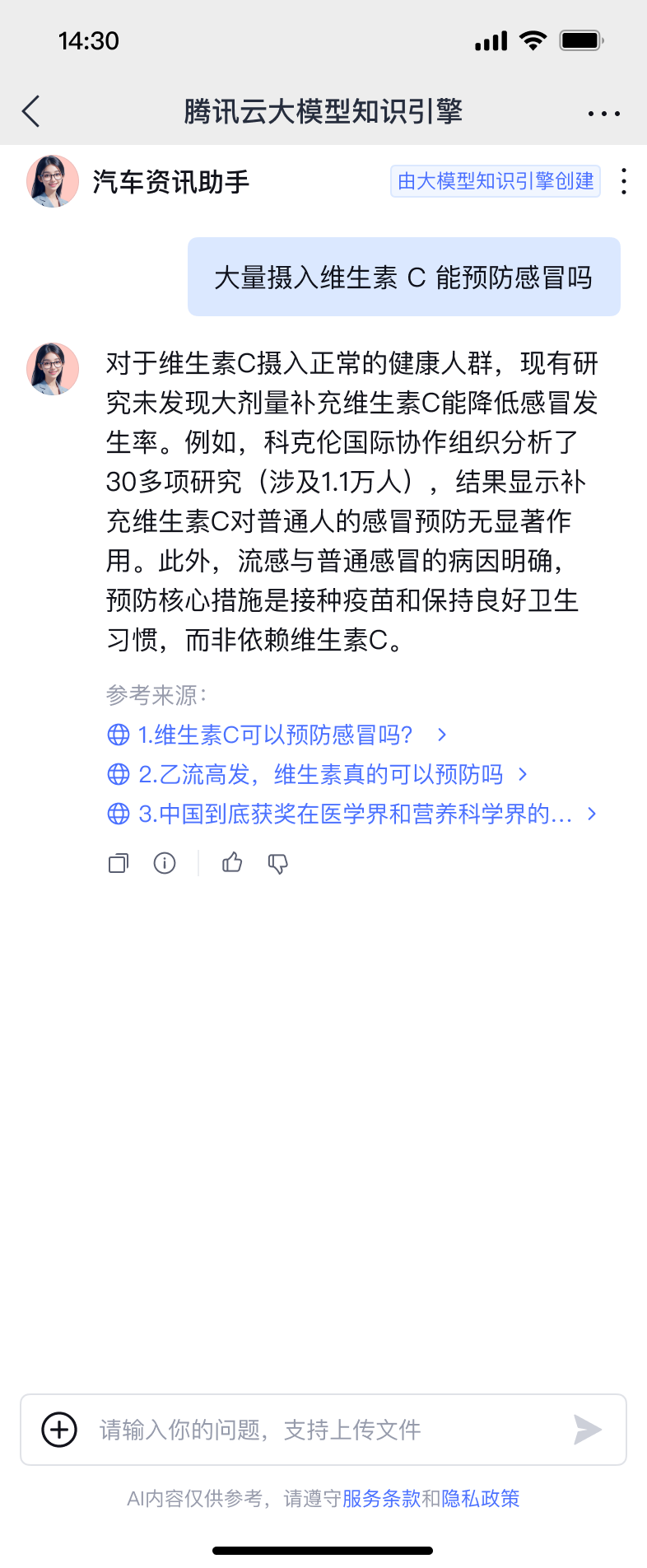


## 5.4 应用端体验

此功能仅支持用户端体验测试，不作为用户端功能交付。创建好的应用支持通过PC端和移动端进行体验。

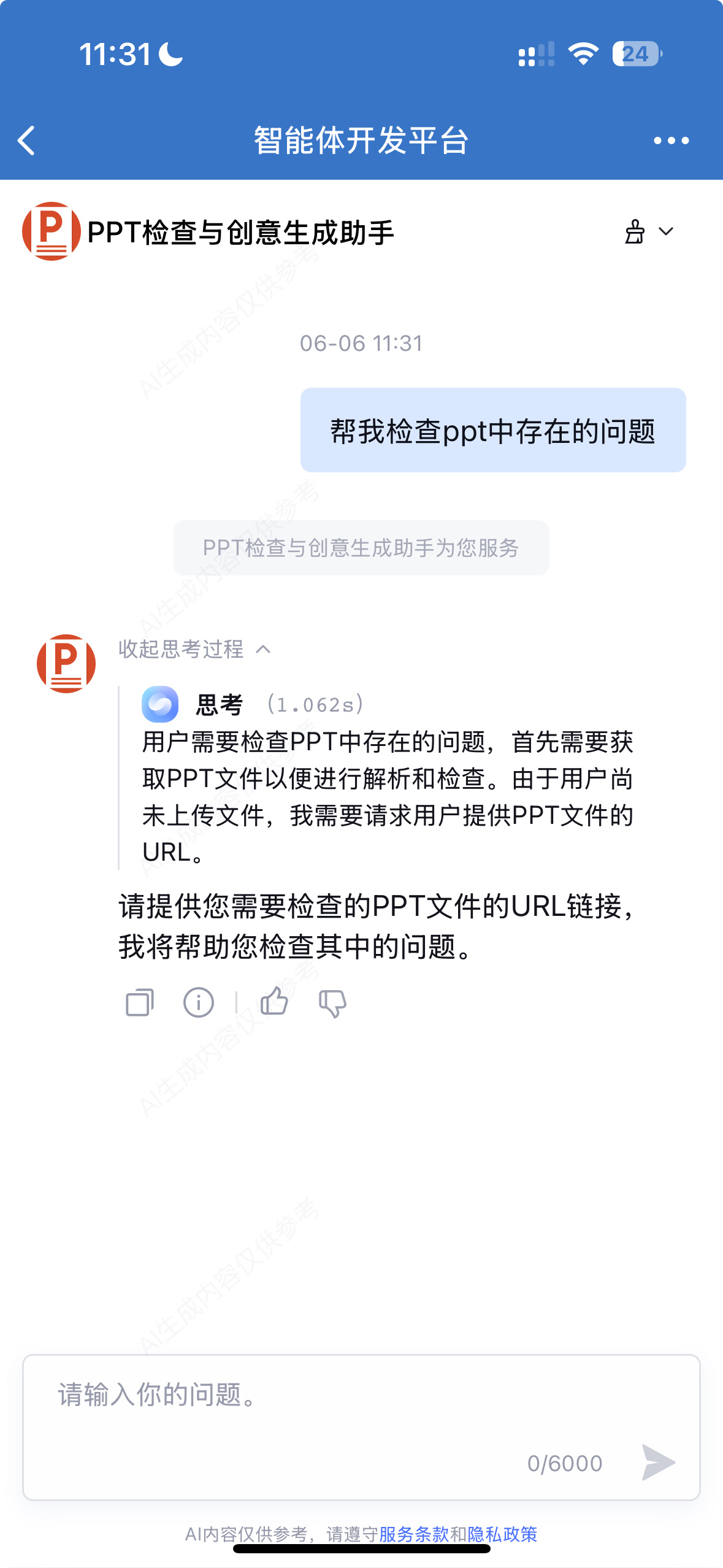
**支持展示消息类型：**

纯文本、图文列表、文本+图片、文本+链接、文本+图片+链接（markdown格式)。

支持思考过程的展示：

应用端支持将大模型思考过程进行呈现，给用户提供更完整的信息参考。

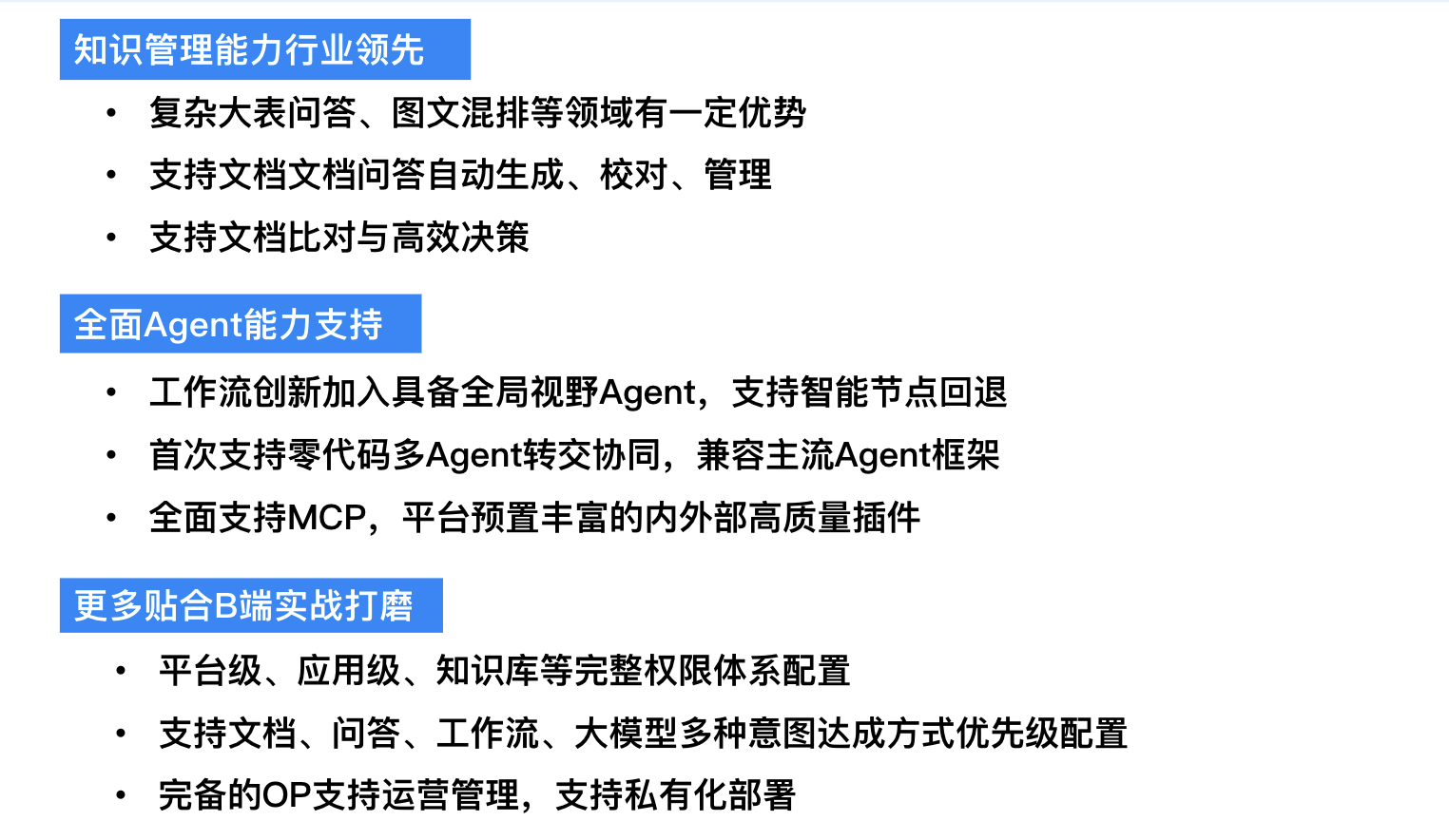


支持用户进行问题反馈：

应用端也支持用户对答案进行问题的反馈



## 5.5 ToB侧实战打磨优势



# 6 核心技术

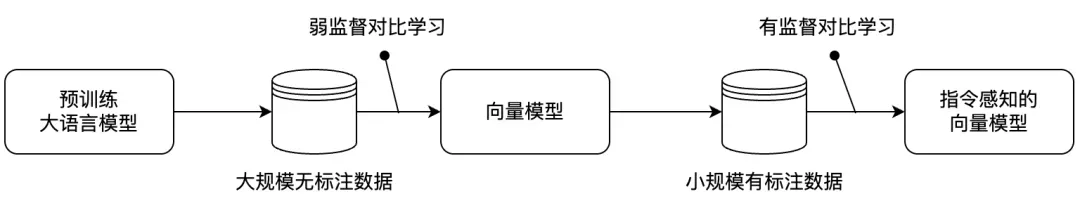
# 6.1RAG 技术优势

在信息爆炸的时代，如何从海量数据中精准获取知识并生成智能回答，已成为AI落地的核心挑战。腾讯云智能体开发平台，依托腾讯优图实验室前沿的RAG体系，突破传统检索与生成的局限，打造了一套覆盖语义检索、结构化表检索、图检索的全栈解决方案 ：从多阶段训练的2B级Embedding模型、Reranker分层蒸馏，到结构化表的智能解析与查询，再到自研GraphRAG框架在构图效率与复杂推理上的突破。目前，优图实验室自研的RAG技术已应用在多个领域和产品，未来，我们更将着力于迈向Agentic RAG与低成本精细化方向，推动产业智能化升级。

## **6.1.1 Embedding模型**

### **（1）多阶段训练管线**

为了提升基于大语言模型（LLM）的向量模型的检索能力，采用多阶段训练策略，逐步增强向量模型的泛化能力和检索效果。

图1.1. 训练管线概览图

* 弱监督对比学习训练。通过批次内负样本共享和跨设备负样本共享技术，每个查询文本对应多达6万个负样本，来极大增强向量模型的判别能力。
* 有监督对比学习训练。通过优化数据采样方法，使跨设备共享的负样本来源于同一个子数据集，来保证难负样本的质量和难度一致性，提升对比学习的有效性。在输入文本中加入特定任务的指令词，进行指令感知的对比学习，使模型能够根据不同任务调整语义匹配策略，来提升向量模型指令遵循的动态检索能力。

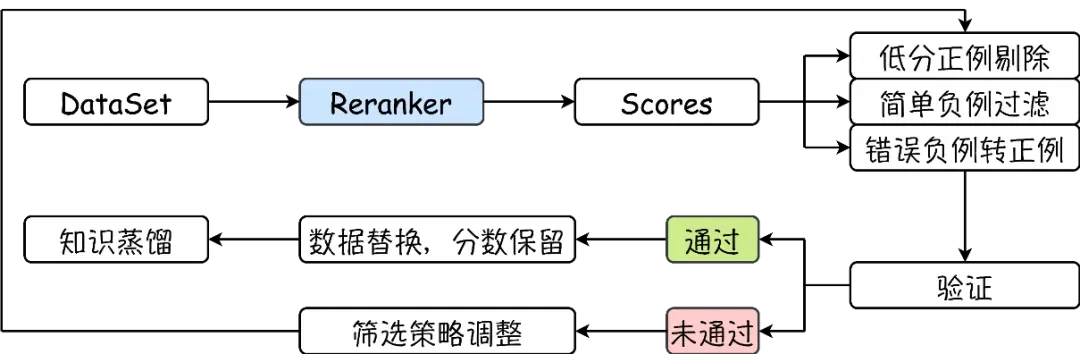
### （2）精细化数据工程

#### -数据构造流程

训练数据的规模和质量对向量模型的效果至关重要，一般地，构建对比学习训练数据的流程如下：

* **构建（问题，相关文档）的文本对**。通常有两种方式，一是在网络上收集已经构建好的开源的问答对数据；二是利用大语言模型杰出的文本生成能力，为文档生成高质量的问题数据。通过收集开源数据和利用大语言模型合成数据，扩充了训练数据的规模，增加训练数据的多样性和丰富性，有助于提高向量模型的泛化能力。
* **挖掘难负样本，构建（问题，正样本，负样本）三元组**。构建两千万规模的文本语料库用于难负样本挖掘，通过扩大语料库规模、构建特定行业语料库、利用大语言模型识别过滤假负样本的方法，优化了挖掘负样本的质量和效果。

#### -数据质量控制

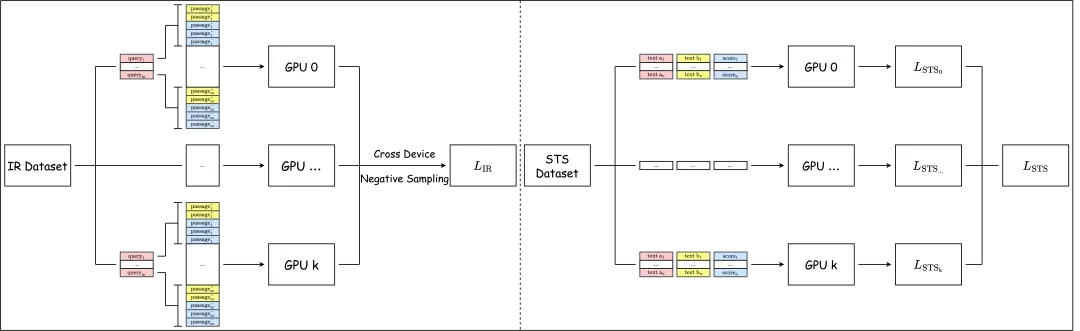
图1.2. 质量控制逻辑示意图

在上述内容基础上，借助 Reranker 模型对训练语料进行筛选及重组，以进一步提升数据质量。大致的处理逻辑包含以下三项：

* 剔除相关性分数极低的伪正例
* 基于相关性分布，过滤简单负样本
* 识别强负例挖掘过程中的潜在正样本，并进行替换

Reranker 模型的评分在通过验证后，会应用于编码器的更新过程，实现label层面的知识蒸馏。

### **（3）多任务均衡配置**

图3. 多任务跨GPU联合训练示意图

为充分发挥 Embedding 模型的潜力，解决不同任务属性、不同领域数据之间的冲突问题，我们设计了一套精密的联合训练方案：

* **数据统一化**：依据数据在组织形式等方面的差异，将整体语料划分为 IR 和 STS 两大类，并采用统一的联调格式同时囊括二者，从而实现混合加载。
* **动态采样器**：跨设备负采样是编码器微调过程中的常用技巧，但在多任务、多领域、多节点混合训练时，跨域数据的引入会为对比学习带来噪音，影响模型表现。对此，通过重构采样器和加载器，保证一次 iteration 中，多个 GPU 获取的样本严格出自同一数据集，并支持为它们设置差异化的 batch size 以充分平衡更新次数。
* **任务特定指令及损失**：不同的检索及匹配任务拥有不同的领域特点及评价标准。相较于不加区分地对待全体数据，差异性的设置可以在最大程度上为参数更新过程注入先验知识。经过分析，我们针对 STS 和 IR 这两大类任务设计了不同的损失函数，同时支持配置个性化指令以灵活应对下游任务。在这种方式下，通过与采样器的联合作用，每个批次将提供纯粹的任务梯度，从而极大地避免强制适配时的潜在性能损失。
* **模型融合策略**：以ModelSoups为代表的权重融合技术此前已被证实可以为CLIP等多模态模型带来提升，而这一方案同样适用于文本嵌入领域。在精调阶段结束后，通过选取不同训练轨迹得到的模型，并精心设置它们的融合方式及权重，进一步增强了网络在各项任务的表现。

### **（4）任务定制损失**

损失函数是模型优化过程的目标及主要参照，对于神经网络的性能具有重要影响。良好的损失函数应充分贴近任务的评价指标，从而为模型提供有效指导。

具体到编码模型最主要的两类应用场景——**文本语义相似性（STS）及信息检索（IR）**。STS任务采用Spearman相关系数作为根本指标，该指标通过计算样本的预测排位与真实排位之差来衡量顺序一致性。IR任务的核心指标nDCG同样是list-wise式的，但它更强调高位优先性。鉴于在大部分IR任务中，与给定query相关的文档其实非常稀少，因此将这些正样本有效突出出来是提升模型表现的关键。

基于这两类任务的差异性和共通性，我们为STS任务引入了**多种顺序性损失，希望模型从逆序对、分数差异性等角度对Embedding分布进行调整，以捕获细粒度的语义区别**。对于IR任务，则会在采集充分多的负样本同时尽可能地扩大query和所有正样本之间的相似度分数，从而增强模型的判别能力。

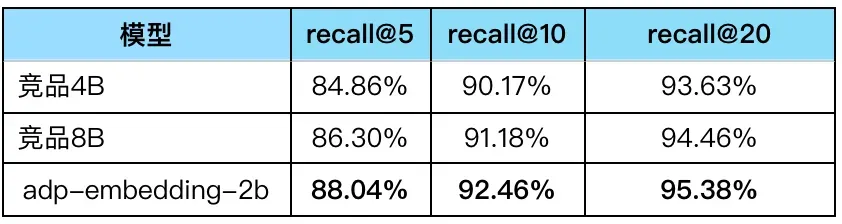
### **（5）模型效果**

目前很多开源的Embedding模型在开源榜单测试集和业务侧测试集上的效果没法很好的平衡，往往顾此失彼。我们的apd-embedding-2b模型能够在这两种测试集上都达到比较好的效果。

我们验证了apd-embedding-2b模型在C-MTEB基准测试中的表现，在中文IR任务和中文STS任务上均取得了SOTA的结果。

* **中文IR任务**
* **中文STS任务**

同时我们也在业务集上进行了实际测试，apd-embedding-2b以2B的参数量超越竞品4B、8B模型的效果，具体结果见下表：



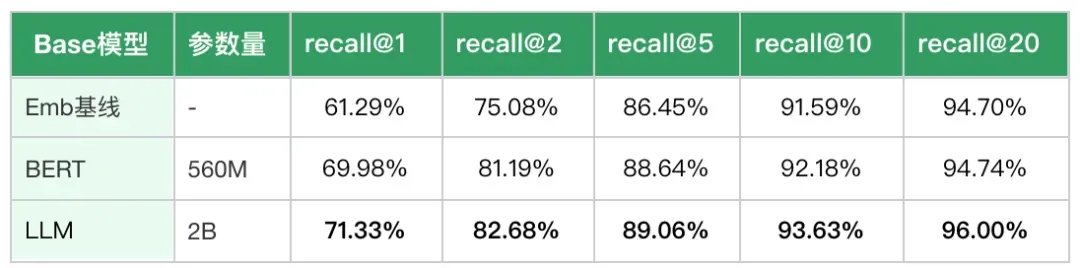
## Reranker模型

尽管向量模型的双编码器架构在实际的检索场景中计算效率高，耗时短，但它却无法直接捕捉查询文本和文档文本之间的微妙关联。为了提升检索环节召回文档的准确性，需要采用基于LLM的Reranker模型对向量模型的检索结果进行重排序。这种方式可以有效捕捉到查询文本和文档文本之间深层次的语义关联，从而给出更准确的检索结果。

### （1）Reranker模型升级为LLM模型

传统的Reranker模型通常基于BERT、RoBERTa等模型进行训练，包括BGE-Reranker-large、Jina-Reranker等，其模型参数量相对较小（110M～400M），输入长度有限（512个token），对自然语言的理解能力远不及LLM。

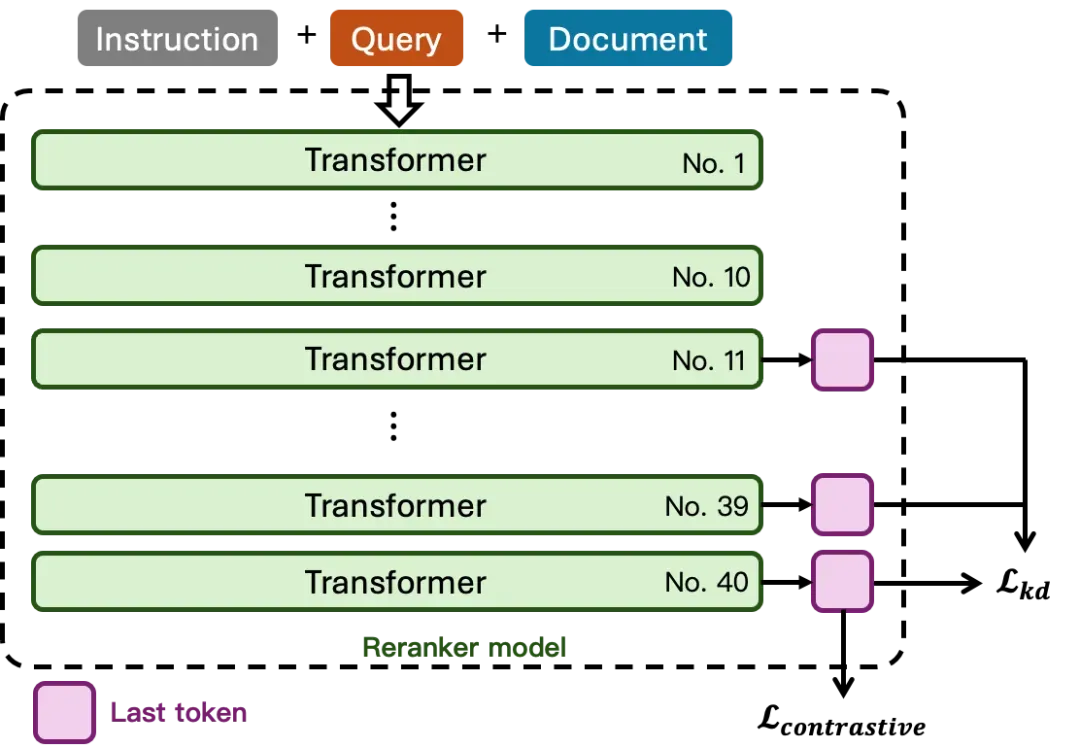
为了提升Reranker模型在实际复杂场景中的表现，使用LLM训练Reranker模型成为必要方案。该方案能够有效发挥LLM对复杂问题和文档的理解能力，从而提供更高质量的文档检索结果，并且其所能支持的文本长度更长（达到8k甚至更长）。同时，通过对特殊任务添加指令，模型也能够适应不同场景的重排序需求。下表是在某业务数据上进行的评测：



### （2）分层知识蒸馏损失

对比学习损失是的Reranker模型训练时常用的损失函数，它的核心作用是帮助模型学习到**区分相关和不相关查询-文档对的能力**，从而有效地提升文档的排序质量。除此之外，知识蒸馏也是一种可用的训练策略。使用更强大的LLM作为教师模型，为查询-文档对给出更精确的相似度分数，然后约束Reranker模型输出和教师模型尽可能保持一致。这两种损失均有助于模型提升文档检索能力，通常可以两者搭配一起使用。

为了进一步发挥知识蒸馏的优势，我们对Reranker模型多个层级的Transformer的输出添加约束，构建**分层（Layerwise）知识蒸馏损失**。这种策略能够强化模型在不同深度层给出较一致的查询-文档相似度分数的能力，也称**层级输出能力**。如果训练数据中未提供教师模型给出的相似度分数，则可以用模型最后一层的输出状态作为知识蒸馏的监督信号，来约束之前的部分层输出和最后一层一致的状态，同样可以实现分层知识蒸馏。

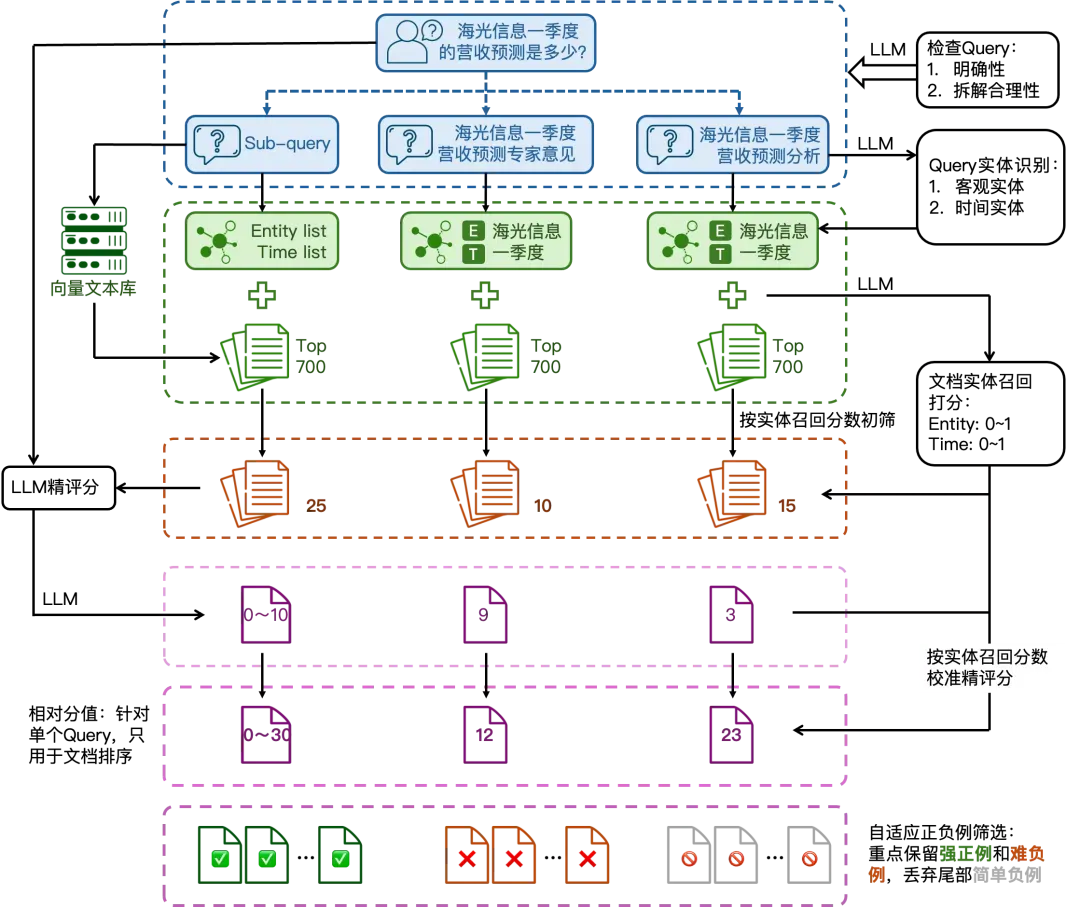
图1.4. 分层知识蒸馏损失策略

使用该策略训练的Reranker模型具备层级输出能力，允许用户选择模型不同层的输出来计算最终的相关性分数。这意味着用户可以选择使用模型较浅层或较深层的输出来进行重排序，这为检索效率和性能提供了更大的灵活性。通过选择合适的层，可以在性能和推理速度之间进行权衡。

### （3）高质量业务训练数据构造

对于特定业务场景，通常缺乏领域适应的高质量训练数据用于Reranker模型的精调。对此，我们构建了一套高效的数据自动化构造流程，能够批量的清洗和构造高质量的训练数据。具体步骤如下：

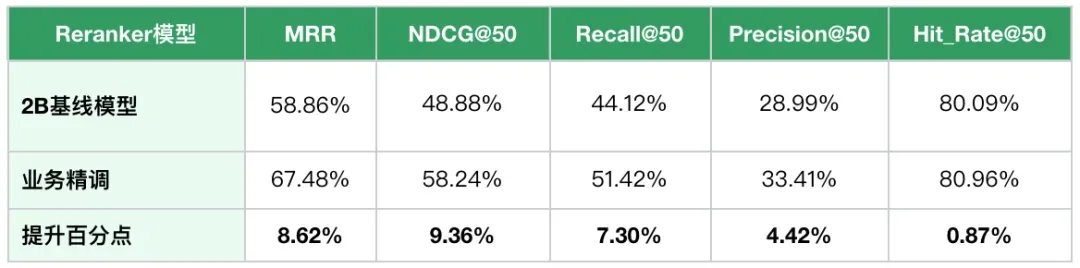
* **Query预处理（可选）**：对于复杂问题，可以优先对问题进行拆解，用子问题（或原问题）借助向量模型进行第一阶段文档检索，同时检查Query的明确性和拆解的合理性，去除无效的Query
* **Query实体识别**：对Query或子问题进行分析，识别其中所包含的有效实体，包括客观实体和时间实体，以此作为文档初筛的参考依据。
* **文档实体召回**：对于步骤1中检索到的文档，使用LLM判断其中是否包含Query中存在的实体，并给出实体召回打分；客观实体和时间实体需要分别打分，0为无召回，1为全部召回。
* **文档初筛**：根据实体召回结果，筛除实体召回打分均为0的文档，不参与下一阶段处理（这些文档可视为简单负例）
* **文档精评分**：使用LLM对初筛后的文档结合Query一起给出相关性打分（这一步的文档数量将大幅度减少，提升精评分速度）
* **分数校准**：对于打分后的文档，根据实体召回的评分重新校准分数；这一步能有效缓解模型在评分时产生的幻觉，纠正一些LLM的不合理判断。校准后的分数仅是针对单个Query的相对评分，只用于文档排序
* **自适应正负例筛选**：
* 按照单个Query的分数分布选取正例，遵循“高分突出的情况下固定正例数量 <=10”和“高分均衡的情况下保持最大分均为正例”两个原则；
* 根据正例数量按固定比例确定负例数量，按分数从高到低依次补齐负例，尽可能保留难负例。

图1.5. 高相关性数据筛选流程

这套数据构造流程的优点在于：

* 通过实体召回对文档进行粗筛，能够**有效降低精评分步骤需要处理的文档数量**
* 通过实体召回打分对精评分进行矫正，能够**有效避免LLM因为幻觉打出错误的高分或低分**
* **自适应正负例采样策略**保证了每个Query所构造的正负例文档都是高质量且分布比例均衡

借助该数据构造流程，目前已针对业务场景进行了精调验证。根据业务评测报告，精调后的版本显著优于线上版本：



## 结构化信息检索

在数据呈指数级增长的今天，企业内部积累了海量的信息数据，其中，结构化数据因其格式规整、语义明确，蕴含着巨大的商业价值。然而，如何让非技术人员也能轻松访问和分析这些数据，一直是业界的难题。

* **结构化数据**：具有固定格式和明确语义，如数据库表格，便于计算机快速查询和处理。
* **非结构化数据**：如文本文档、图片，无固定格式，语义理解难度大。

为应对结构化数据查询的挑战，我们基于经典RAG框架融合Text2SQL技术，通过“理解-检索-生成”的模式，将用户的自然语言问题高效转化为精准的数据结果。

### （1）多源数据检索

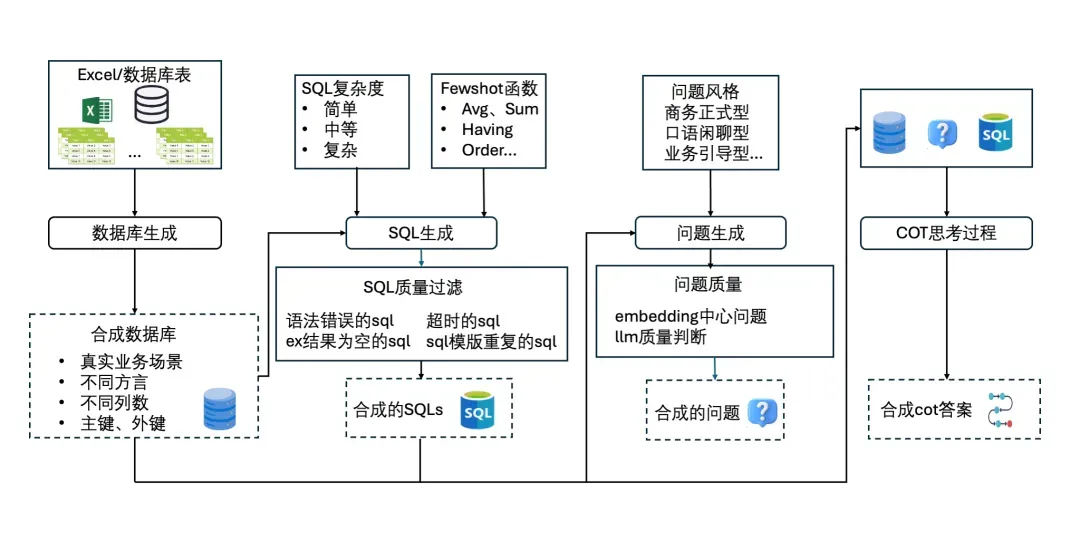
结构化数据常见数据源形态包括DB数据库表、表格文件等，业务上通过支持不同数据源的载入，设计了基于文本切片检索的RAG与Text2SQL融合的方案，将文本切片与text2sql查询结果送给下游阅读理解模型。阅读理解模型会综合两类信息，生成更准确、更全面的回答——既包含基于统计或字段的精确数据，也包含相关文本切片提供的上下文解释或补充信息。 整体检索问答方案如下：

图2.1. 不同数据源载入问答系统

### **（2—— Text2SQL核心技术**

（1）自动化数据合成和增强

数据合成对Text2SQL任务具有重要价值，主要体现在快速适配新场景和提升模型泛化能力两方面。通过自动化生成多语言的数据库表结构、自然语言问题及带推理过程的SQL答案对，系统能快速构建适配不同数据库方言（如SQLite、MySQL等）的训练数据。这种能力不仅显著降低人工标注成本，更重要的是使模型能预先学习到多样化的schema结构和查询逻辑，当面对真实业务中新出现的数据库范式或查询需求时，模型凭借合成数据训练获得的"经验"能更快实现性能收敛。特别是合成的"带思考过程的SQL答案"通过显式展现查询逻辑的构建路径，有效增强了模型对复杂查询的语义解析能力。

图2.2. 数据合成方法

通过数据合成加训，对新场景提升效果如下：

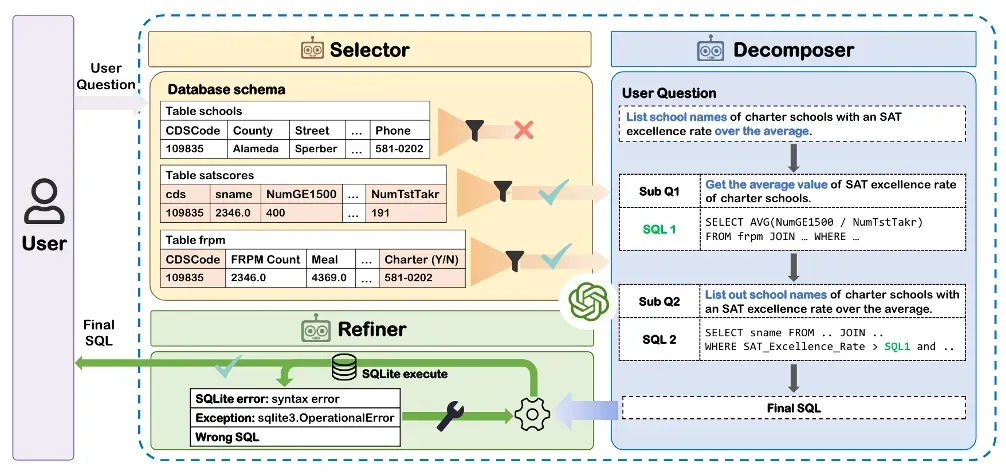


（2）基于Agent的Text2SQL框架

Text2SQL 是一项将自然语言转换为SQL的技术，它允许用户通过日常语言与数据库交互，而不需要掌握专业的SQL语法。在实际业务中落地应用仍面临诸多挑战。例如领域知识的泛化能力，自然语言表达的多样性与复杂性，语义不明确、不完整等。

我们提出基于大语言模型的多智能体（Multi-Agent）协作框架，该框架由三个Agent组成：

* **筛选器（Selector）**：从众多表中选择相关表和列，减轻不相关信息的干扰；
* **分解器（Decomposer）**：将复杂的问题分解为子问题并逐步解决它们；
* **优化器（Refiner）**：使用外部工具执行SQL并获取反馈，根据反馈信息优化错误的SQL。

图2.3. MAC-SQL技术架构概览（中稿COLING 2025 [1]）

基于开源 **BIRD** 和 **Spider** 数据集，本框架配合自研的7B模型，执行准确率超过ChatGPT-3.5等。本框架的方法配合 GPT-4 使用，能够达到SOTA的水平，远超单独直接使用GPT-4的效果。

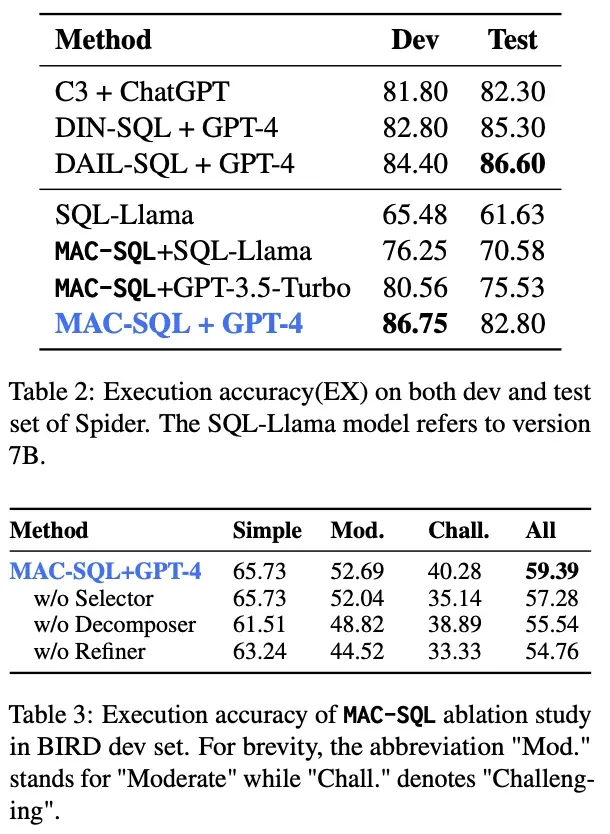


图2.4. 效果对比

# 6.2 Agent技术优势

1. **支持Multi-agent架构，对齐openAI，支持主agent和子agent协同响应对话；**
2. **采用业界创新的PDL语言描述工作流提供给Agent使用**，让对话执行兼顾可靠性与灵活性。
3. **分层解耦：**微服务化提升迭代效率；通用能力下沉，把插件服务，代码执行，大模型服务下沉。
4. **高性能低延时**：轻量协程实现，事件驱动，进程内的异步队列通讯

|  |  |  |
| --- | --- | --- |
| **序号** | **通用能力/独有能力/优势能力** | **腾讯云智能体开发平台** |
| 1 | 优势 | 支持模型量化和多batch推理 |
| 2 | 优势 | 问答提取，答案支持输出图片、表格、表情等多模态信息 |
| 3 | 优势 | 阅读理解支持图片、表格、表情等多模态信息 |
| 4 | 优势 | 文档切分集成语义切分模型，根据语义进行文档段落切分，效果比规则切分更加准确 |
| 5 | 优势 | 支持复杂表格分析 |
| 6 | 优势 | 支持不同长度的embedding模型和reranker模型，效果超过开源模型 |
| 7 | 优势 | 支持通过text2sql、混合检索、小找大策略等解决复杂表格的检索 |
| 8 | 优势 | 支持200+角色风格和角色指令遵循对话，支持角色设定一键优化，角色扮演知识问答 |
| 9 | 优势 | 模型本身可以解决95%以上的有害问题 |
| 10 | 优势 | 复杂意图识别与参数提取能力 |
| 11 | 优势 | 模型支持将数据图（如：柱状图、折线图、饼状图等）转换为标准markdown表格形式，结合LLM进行问答 |
| 12 | 优势 | 模型支持将UML类图像（如：流程图、架构图等）转为mermaid形式，结合LLM进行问答 |
| 13 | 独有 | Agent-支持软性的流程控制 |
| 14 | 独有 | 支持图文的多模态检索和跨模态检索，提升检索的效果 |
| 15 | 优势 | 多模态阅读理解-支持根据用户问题，输出知识库或C端文档中的图片回答问题。 |
| 16 | 优势 | 多模态阅读理解-支持根据用户问题，理解知识库或C端文档中“无文字”的图片内容，并基于理解回答问题。 |
| 17 | 优势 | 多模态阅读理解-支持用户输入与知识库或C端文档相关的图片作为问题的一部分，结合对应文档内容输出文字和图片回答问题 |
| 18 | 独有 | 文本切分-支持多级文档结构化切分，实现更准确更灵活的文本切片检索 |
| 19 | 优势 | Agent模式-精调function-call模型自主规划和工具调用能力，工具调用成功率领先。 |
| 20 | 优势 | Agent模式-精调function-call模型的反思纠错能力效果领先。 |
| 21 | 优势 | 工作流-参数提取效果、意图识别效果领先 |
| 22 | 优势 | 工作流-模型根据对话内容实现工作流节点间的灵活跳转 |

# 7 产品对接形式

交付物形态：智能体开发平台端+应用端对话运行时API接口

|  |  |  |  |
| --- | --- | --- | --- |
| 输出层级 | | 场景 | 对开发者要求 |
| C端 | 配置端 |
| API | 标准平台 | 有C端定制需求，配置端可接受独立系统跳转维护/单点登录 | 1. 有前端开发资源，可对C端返回数据做排列组合，组装为新的样式 2. 有后台资源，配置端可完成与客户系统单点登录的适配工作 |
| API | 均有定制需求 | 同上+  3、对对话服务场景有较深理解，有前端开发资源可基于API对管理平台页面做定制开发 |