

AI_REC 文档



【版权声明】

版权所有©百度在线网络技术（北京）有限公司、北京百度网讯科技有限公司。未经本公司书面许可，任何单位和个人不得擅自摘抄、复制、传播本文档内容，否则本公司有权依法追究法律责任。

【商标声明】



和其他百度系商标，均为百度在线网络技术（北京）有限公司、北京百度网讯科技有限公司的商标。本文档涉及的第三方商标，依法由相关权利人所有。未经商标权利人书面许可，不得擅自对其商标进行使用、复制、修改、传播等行为。

【免责声明】

由于产品版本升级或其他原因，本文档内容会不定期进行更新。除非另有约定，本文档仅作为使用指导。如您购买本文档介绍的产品、服务，您的权利与义务将依据百度智能云产品服务合同条款予以具体约定。本文档内容不作任何明示或暗示的保证。

目录	
目录	2
产品介绍	5
产品描述	5
购买指南	7
产品定价	9
版本选型建议	10
服务等级协议SLA	10
快速入门	12
快速入门	12
操作指南	16
数据仓库	16
数据管理	16
数据接入	19
数据探查	23
应用管理	28
创建应用	28
调用服务	29
兴趣构建（LLM增强版功能）	31
查看模拟用户画像	32
模拟体验PC端demo	34
场景管理	37
场景管理	37
推荐策略管理	38
强插策略	38
打散策略	39
排序调权策略	39
推荐物料选品	40
物料过滤策略	41
基于地理位置召回	43
负反馈与黑名单	44
流量调控策略	45
置顶策略	46
搜索策略管理	48
搜索query词召回	48
搜索精确度配置	49
索引配置	49
词典管理	50
热搜榜配置	51
相关搜索	53
拼音召回策略	56

Baidu 百度智能云文档	目录
联想词策略运营	57
猜你想搜策略运营	59
用户运营	60
搜索自定义召回	61
监控报表	64
服务监控	64
推荐链路	66
用户画像	68
业务指标监控	69
版本迭代	70
AB实验	70
版权内容	73
创建内容集	73
使用内容集	75
管理内容集	76
其他	77
多用户访问控制	78
模型超参配置	78
API文档	89
数据接入接口	89
行为同步接口	89
用户同步接口	93
用户数据详情接口	95
物料同步接口	96
物料简单查询接口	101
物料数据详情接口	104
物料条件批量调权接口	105
上下架删除接口	108
推荐服务接口	109
个性化推荐接口	109
相关推荐接口	112
热门推荐接口	114
新推荐接口	117
自定义筛选使用说明	120
自定义筛选使用说明	120
自定义召回支持第三方服务API	121
一、自定义召回支持第三方服务API功能说明	121
二、接口对接示例	121
1、输入规范样例	121
2、输出规范样例	122
3、请求示例	122

产品介绍

产品描述

移动时代下，流量的竞争、私域流量的运营、用户价值的挖掘成为业务成功的关键目标。越来越多的行业都在自建独立运营APP，但普遍缺乏流量运营的经验。百度将从内容、个性化推荐、全渠道营销三方面技术赋能，助力精细化、场景化、个性化运营自有流量，带来业务的提升。

一、产品介绍

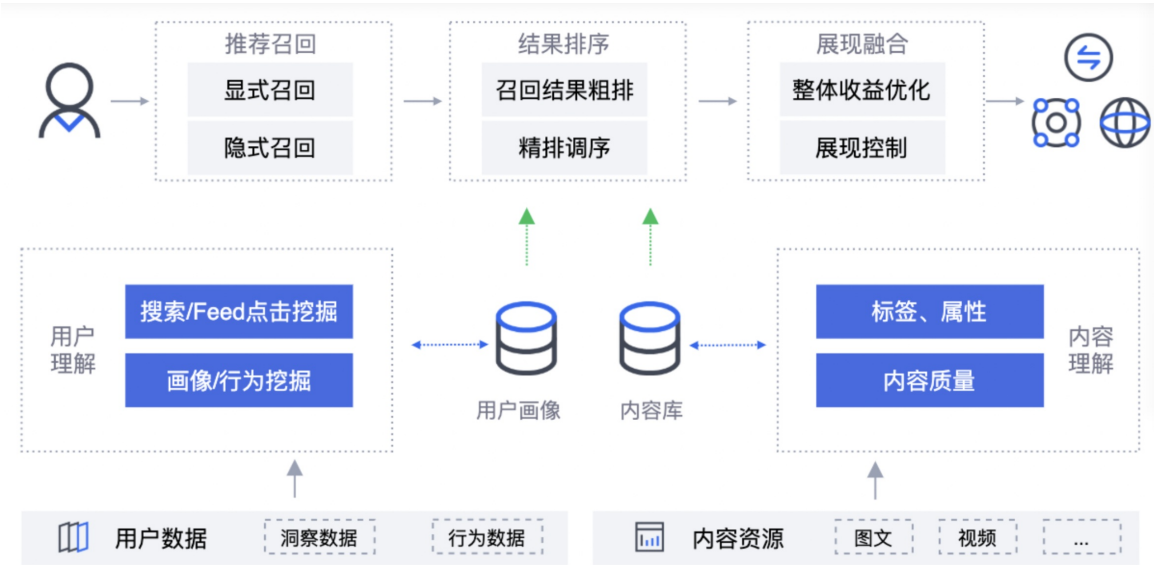
智能推荐引擎，依托百度开源深度学习平台，通过推荐、内容和搜索三位一体模式提高个性化水平，为开发者提供端到端的RESTful API 和 SDK，帮助业务系统提升私域流量用户活跃、留存和转化效果。

二、核心概念

- 1、应用：应用定义了一个API调用者的身份（服务消费者），可以将多个API授权给同一个应用，APP被授权后才能调用API。
- 2、Token：访问令牌（Access Token）是调用智能推荐引擎服务的凭证，防止被窃取恶意调用。
- 3、服务：计算机中的服务是一种应用程序类型，在后台运行，每个服务都有特定的功能。服务通常以API的方式供应用使用。
- 4、物料：物料是推荐系统可推荐对象的统称。例如在电商栏位下，物料表示商品；在资讯栏位下，物料表示文章。
- 5、内容集：该参数出现需要满足的条件：用户在物料数据来源选择了百家号内容资讯服务。
- 6、内容资讯服务：百度内容资讯池由百家号等内容平台提供资源支持，垂类内容和创作者都可根据需求灵活调整与订制。
- 7、兴趣点强度：感兴趣（有过行为）的物料按照标签粒度聚合的数量。
- 8、调权策略：在重排序阶段通过调整物料权重干预从而影响最终的排序结果。
- 9、QPS（每秒查询数）：表示每秒钟系统处理的查询或请求的数量。 $QPS = PV / \text{平均页面响应时间}$ 。
- 10、打散策略：原始推荐结果按照相似度打散为多个请求中。
- 11、强插策略：强制插入最近一个正反馈相关的几个相关推荐内容相似资源
- 12、召回策略：根据召回候选物品，用户和物品相关性、物品和物品相关性、用户和用户相关性、用户和标签相关性或者标签和物品相关性等计算方式召回物料。

三、技术架构

基于用户数据、行为数据、物料数据多方数据源训练三段式“召回-排序-融合”推荐算法，依托百度高性能高可靠的工程架构保障，助力构建全链路推荐系统。



四、产品优势

- 1、简单可依赖的端到端的智能推荐引擎，接入简单、使用简单和维护简单。
- 2、提供多场景推荐api，含猜你喜欢、相关推荐、热门推荐开发接口。
- 3、内置多种机器学习、深度学习模型，通过简单的无代码化配置完成ai模型调用，屏蔽复杂性。
- 4、全方位的服务监控，含接口调用、接口异常、服务健康度、用户交互等统计，监控无死角。
- 5、推荐、内容和搜索三位一体模式提高个性化水平，搜索推荐行为融合相互补强。
- 6、预置百家号内容物料数据，无需接入开箱即用。

五、产品特性

1、高效接入数据

自适应对接：支持自定义数据源灵活对接，实时同步，确保数据使用质量并快速上线。

推荐物料管理：图文资讯、权益商品、定制活动等各种业务均支持数据管理和应用推荐。

2、优质推荐模型

模型自主学习：模型每天基于增量数据自学习，模型自动迭代优化自学习至最优状态。

灵活配置算法：支持无代码编排召回算法、排序算法和综合展示策略。

3、多维效果调优

适配多种工具：智能推荐引擎支持流量、行为、漏斗等数据分析工具。

提高运营效率：以数据分析触达业务指标，调整运营策略，提升运营效率。

4、稳健系统监控

监控能力稳定：支持通过性能监控、业务监控，提升异常问题排查与解决的效率。

平台化运维：形成平台化运维，全局监控保障服务稳定。

六、应用场景

(1) 信息流推荐

在全渠道内容中，结合用户标签快速优化内容精准分发用户期待的内容，提高终端用户在站内的活跃度和粘性。

场景优势： 推荐策略适配度高，减少企业开发与试错成本； 精准获取用户心智，提升用户粘性与活跃度。

(2) 商品推荐

根据用户属性、历史行为、兴趣偏好等指标为用户在APP首页、相似商品页等场景进行个性化推荐。

场景优势： 用户浏览时，推荐相关商品、兴趣商品，提升商品销售转化； 在相关页面，通过猜你喜欢与相关推荐增加产品曝光，挖掘购物需求。

购买指南

概述

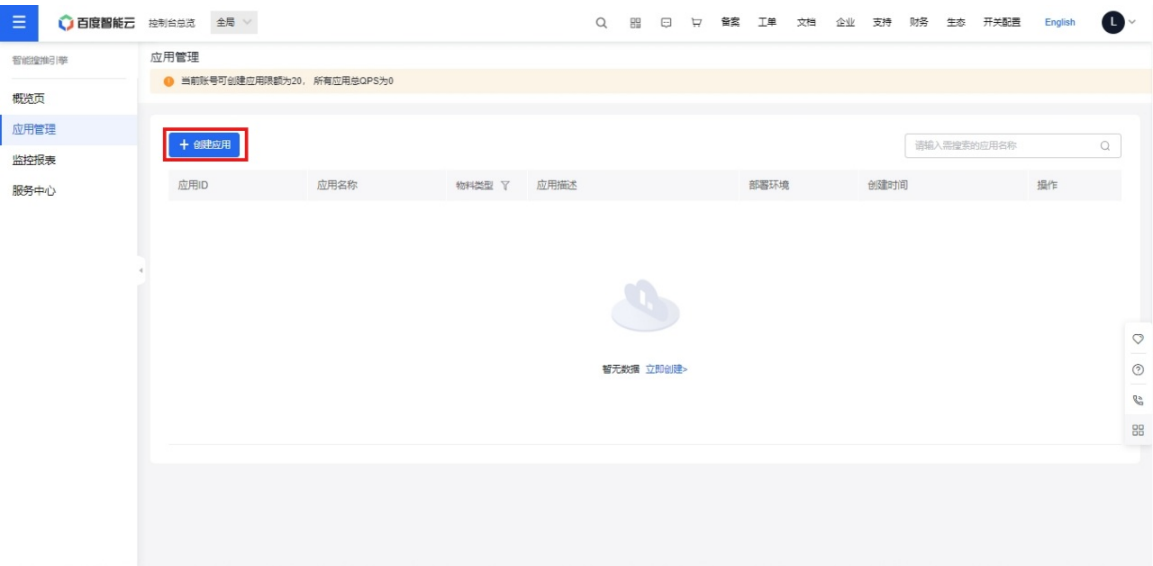
在应用管理模块新增下单流程，当应用首次创建或权益余额不足时支持配置应用付费信息。

前提条件

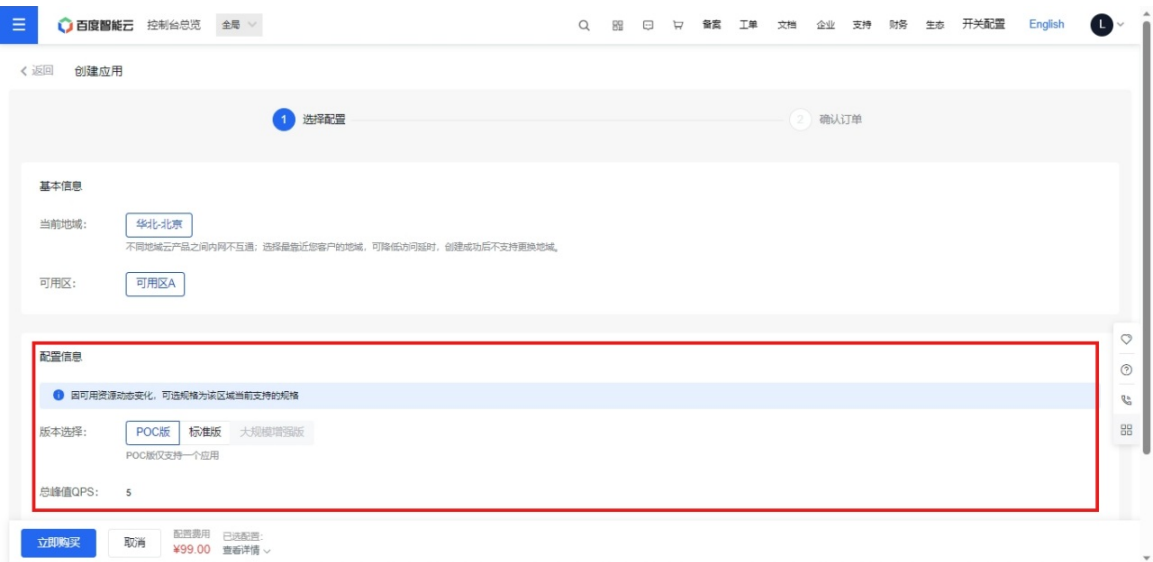
有权限登录进入百度智能搜推引擎，同时具有应用创建权限。

操作步骤

1点击【创建应用】按钮，如果是首次创建则跳转至套餐购买页面，如果不是则弹出创建应用弹框。另外在创建应用弹框中填写完应用创建信息后如果权益余额充足则创建成功，如果不充足则跳转至套餐购买页面。

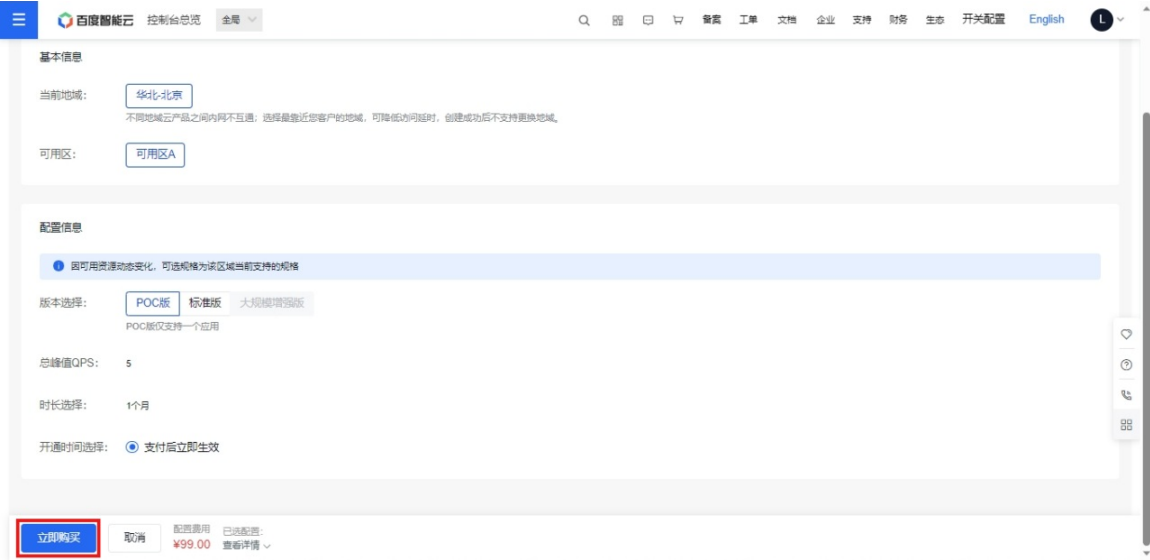


2进入到套餐购买页面，第一步是选购,选购需要填写地域信息、配置信息、购买信息。【地域信息】需要填写可用区域，选择好的可用区域显示在当前地域中。【配置信息】需要选择请求总QPS、物料总数、用户总数和是否使用大模型增强服务。填写【购买信息】需要选择购买时长、是否自动续费，如果自动续费则需要选择续费周期。

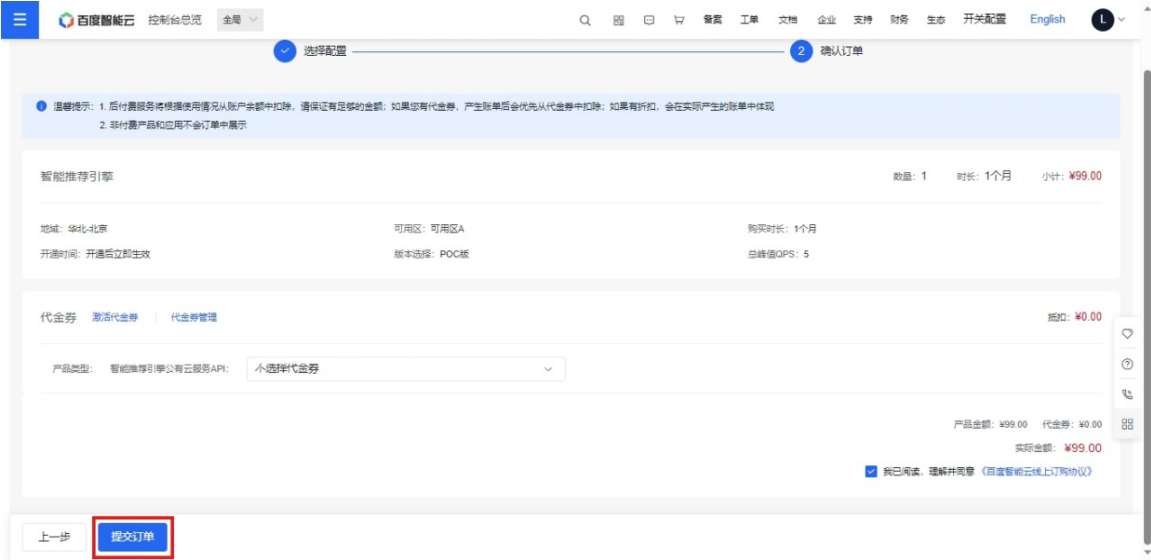


3地域信息、配置信息、购买信息都填写完成后会显示在所选配置区域内。在该区域内点击【立即购买】会进入到订单确认

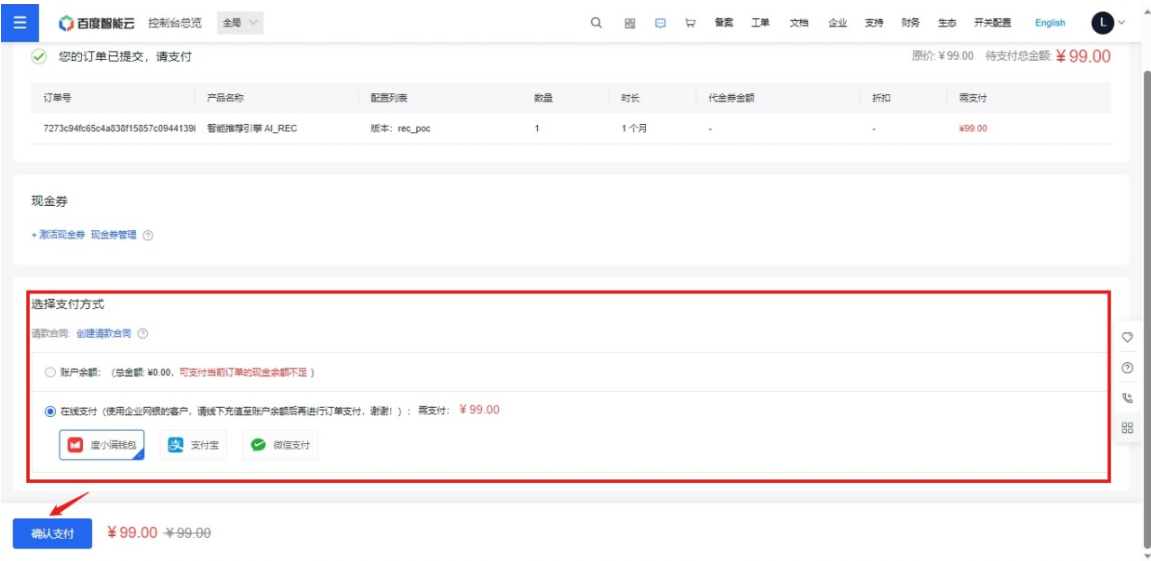
页。



4在订单确认页中可查看配置的套餐信息。查看后如果对套餐信息确认无误则点击【提交订单】按钮，会进入到订单支付页。如果订单有误则点击【上一步】按钮，会返回到选购页面。



5在订单支付页选择任意支付方式完成支付后，点击【确认支付】按钮系统会弹出开通成功提示。



6在开通提示页中可查看【订单明细】和跳转至【管理控制台】。



产品定价

智能推荐引擎（SaaS）报价=推荐引擎基础套餐包（必选，任选其一）+推荐引擎增值服务（非必选），具体如下所示：

一、推荐引擎基础套餐包（必选，含搜索推荐能力） 收费方式：按照QPS指标包年包月，分阶梯计价。

最小 QPS总数	最大 QPS总数	标准版单价/月	专业版（待上线）单价/月
5	20	1000元/qps	待上线刷新
21	80	超出部分700元/qps	待上线刷新
81	200	超出部分400元/qps	待上线刷新
201	1000	超出部分300元/qps	待上线刷新

二、推荐引擎增值服务（非必选）

- 1、技术专家支持，上线投产支持：免费1次，后续每次收取5000元/次
- 2、百家号及第三方的内容引入服务：单独报价

三、报价说明

- 1、可根据接口数、接口调用量、物料数量、用户数量自由选择套餐类型，如有特殊需求请联系商务按需打包计算。
- 2、推荐接口当前支持：性化推荐：常以“推荐”、“猜你喜欢”等形式出现，千人千面；相关推荐：根据当前内容计算相关内容，常以“相关推荐”、“买了还买”等形式出现，一般放在内容详情页；热门推荐：基于上报的各种数据进行计算，常以“热点”等形式出现。
- 3、推荐效果指标可线下商议具体的效果指标，支持定制。
- 4、个性化推荐的saas版本支持poc测试1个月。
- 5、QPS估算方法

以 DAU = 1000w 为例：
PV: 按照日访问量为日活的5倍计算，PV = 1000w * 5 = 5000w
均值QPS: 均值QPS = 访问量/时长 = 5000w/(86400) = 578
峰值QPS: 峰值 QPS 按照均值的10倍预估 = 5780

6、举例：客户购买50qps——20000+（50-20）*700=41000/月

按照qps计算总价说明					
单价(元/qps*月)					总价=各个阶梯价格之和， 即落到橙色区域面积
1000					
700					
400					
300					
QPS:	5	20	80	200	1000

版本选型建议

一、版本介绍与优势

版本	介绍	版本优势
POC版	POC版（Proof of Concept，概念验证）的目标主要是验证搜索推荐系统的核心功能和技术可行性。 仅用于测试试用为目的不可作为生产环境，最多使用1个月。	·功能简单上手快。 ·价格更具经济性。
标准版	标准版具有更多的功能和更好的性能，可以满足大多数用户的搜索和推荐需求。对于更高阶的配置需求和性能要求小。	·开箱即用，无需运维，可一键完成智能搜索和推荐全链路服务部署 ·支持搜索和推荐一体化引擎，支持组合查询，含强大的全文检索和语义召回能力。
专业版 (即将上线)	专业版是针对特定行业或领域的高级用户而开发的版本。 支持模型参数配置和大模型增强等更高级的功能和更强的性能，可以满足专业用户在搜索和推荐方面的特殊需求。	·支持大模型增强能力，如用户兴趣理解、大模型自动化测评、问答等能力。 ·多模态，支持图搜、音视频搜索、个性化搜推。 ·支持多种深度模型管理，支持参数配置等更多自定义能力。 ·性能更强、支持文档和数据存储空间更大，权益更多。

二、权益对比

权益	qps区间	poc版	标准版	专业版(即将上线)	说明
最大应用数	1个	4个	4个	10个	超出后：无法创建应用。
最大物料总数	1~5	5w	无	无	较早的数据被清除。
	5~500	无	5w条/QPS	7w条/QPS	
最大用户总数	1~5	5w	无	无	较早的数据被清除。
	5~500	无	5w条/QPS	7w条/QPS	
最大行为总数	1~5	10w	无	无	较早的数据被清除。
	5~500	无	15w条/QPS	20w条/QPS	
大模型调用数	1~5	无	无	无	超出后：访问被拒绝。
	5~500	无	无	2000次/月/qps	

服务等级协议SLA

本服务等级协议（Service Level Agreement，以下简称 "SLA"）规定了百度智能云向客户提供的智能推荐引擎（AI_REC）的服务可用性等级指标及赔偿方案。

1.定义

服务周期：一个服务周期为一个自然月。

服务周期总分钟数：按照服务周期内的总天数 * 24（小时） * 60（分钟）计算。

服务不可用：因百度智能云故障导致了推荐接口报错或者不可用。

服务不可用分钟数：在一个服务周期内智能推荐引擎（AI_REC）的服务不可用分钟数之和。

月度服务费：为用户在一个服务周期（即自然月）内所支付的服务费用总额，如用户一次性支付多个月的服务费，则将按照所购买的月份分摊计算月度服务费。

2. 服务可用性

2.1 服务可用性计算方式

服务可用性将根据服务周期统计服务的可用性：

服务可用性 = $\left(\frac{\text{服务周期总分钟数} - \text{服务不可用分钟数}}{\text{服务周期总分钟数}} \right) \times 100\%$

2.2 服务可用性承诺

- 1.百度智能云承诺一个服务周期内AI_REC的服务可用性不低于99.9%；
- 2.如服务未达到上述服务可用性承诺，客户可以根据本协议第三条约定获得赔偿。赔偿范围不包括以下原因所导致的不可用：

(1) 百度智能云预先通知用户后进行系统维护所引起的，包括割接、维修、升级和模拟故障演练；

(2) 用户实际的流量超出所购买的服务规格参数，导致流量激增、资源不足引起的服务不可用。如购买了5qps规格参数实际流量为10qps。

(3) 用户的应用程序或数据信息受到黑客攻击而引起的；

(4) 用户维护不当或保密不当致使数据、口令、密码等丢失或泄漏所引起的；

(5) 用户的疏忽或由用户授权的操作所引起的；

(6) 不可抗力以及意外事件引起的；

(7) 使用本地盘存储的云服务器器的本地磁盘中数据有丢失的风险（比如本地磁盘损坏等），依赖本地盘及本地盘中数据作为启动依赖项而导致的不可用；

(8) AI_REC服务只赔付AI_REC自身，不赔付AI_REC实例所关联的其他云服务；

(9) 任何百度智能云所属设备以外的网络、设备故障或配置调整引起的；

(10)其他非百度智能云原因所造成的AI_REC服务无法正常使用。
3. 服务赔偿条款
- 3.1 赔偿标准
- 服务可用性低于99.9%，可按照下表中的标准获得赔偿，赔偿方式仅限于用于购买AI_REC产品的代金券，且赔偿总额不超过未达到服务可用性承诺当月客户就该AI_REC服务支付的月度服务费。
- | 服务可用性 | 赔付代金券金额 |
|-------------------|-----------|
| 低于99.9%但是等于或高于99% | 月度服务费的10% |
| 低于99%但等于或高于95% | 月度服务费的20% |
| 低于95% | 月度服务费的30% |
- 3.2 赔偿申请时限
- 12

用户可以在每月的第五个工作日后对上个月发生故障的AI_REC提出赔偿申请，赔偿申请必须限于在AI_REC故障的相关月份结束后的两个月内提出。超出申请时限的赔偿申请将不被受理。

4. 其他说明

- (1) 在法律法规允许的范围内，百度智能云负责对本协议进行解释说明。
- (2) 本协议一经公布立即生效，百度智能云有权对本SLA条款作出修改。如本SLA条款有任何修改，百度智能云将以网站公示或发送邮件的方式通知您。如您不同意百度智能云对SLA所做的修改，您有权停止使用AI_REC服务，如您继续使用AI_REC服务，则视为您接受修改后的SLA。
- (3) 本协议项下百度智能云对于用户所有的通知均可通过网页公告、站内信、电子邮件、手机短信或其他形式等方式进行；该等通知于发送之日视为已送达收件人。因用户未及时获知百度智能云的服务变更或终止条款遭受损失的，百度智能云不承担任何责任。
- (4) 本协议的订立、执行和解释及争议的解决均应适用中国法律并受中国法院管辖。如双方就本协议内容或其执行发生任何争议，双方应尽量友好协商解决；协商不成时，任何一方均可向北京市海淀区人民法院提起诉讼。
- (5) 本协议构成双方对本协议之约定事项及其他有关事宜的完整协议，除本协议规定的之外，未赋予本协议各方其他权利。
- (6) 如本协议中的任何协议无论因何种原因完全或部分无效或不具有执行力，本协议的其余协议仍应有效并且有约束力。
- (7) 关于用户约束条款，详见《用户服务协议》中的"用户的权利与义务"相关条款内容。
- (8) 关于服务商免责条款，详见《用户服务协议》中的"免责声明"相关条款内容。

快速入门

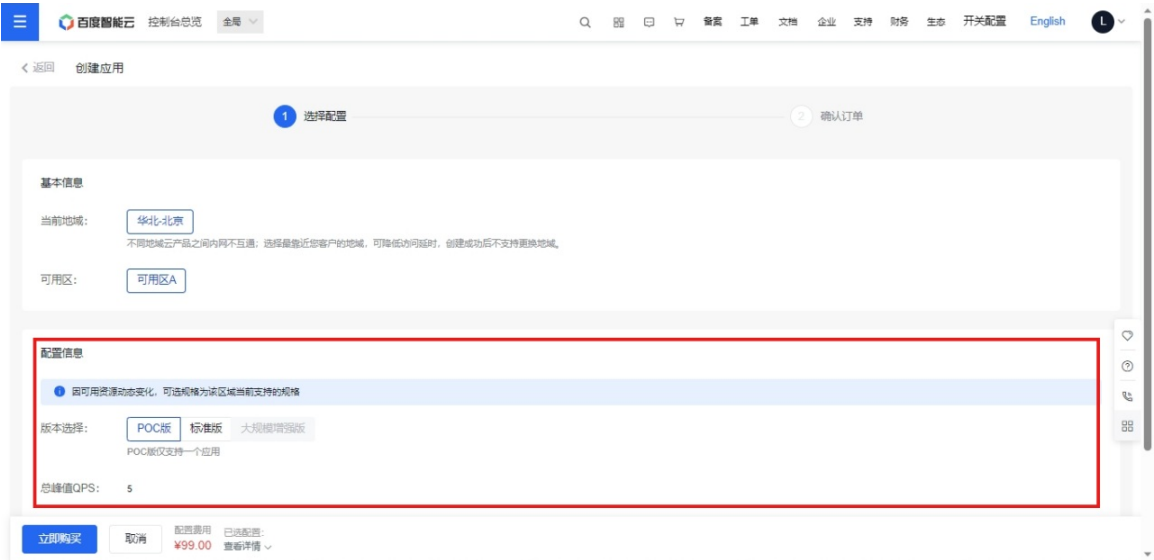
快速入门

核心操作流程



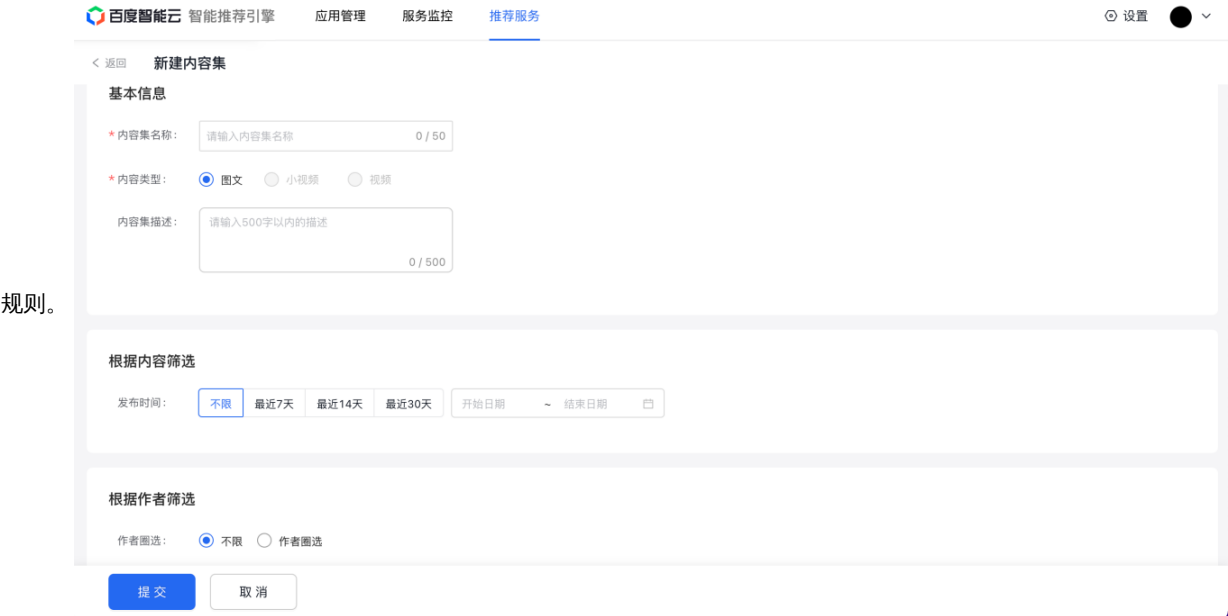
0 下单

如果是首次创建创建应用时则跳转至套餐购买页面，在套餐购买页中需要填写地域信息、配置信息、购买信息。所有信息填写后确认并支付订单，点击确认支付按钮系统会弹出开通成功提示。



1准备阶段

如果您选择百家号内容资讯服务作为推荐内容，请关注该步骤，如果没有跳过该步骤即可。通过设定筛选条件圈选内容资讯，并可以将其定义为某类内容集，供推进应用绑定作为物料数据。用户可以根据实际需求按照内容和作者两个视角自主设定筛选



2创建应用

通过新建应用定义API调用者身份的方式绑定推荐场景，配置应用数据接入和输出服务参数实现服务快速启动。

百度智能云 智能推荐引擎

应用管理 服务监控 推荐服务

设置

< 返回

创建应用

基本信息

* 应用名称:

请输入50字以内名称

0 / 50

* 应用描述:

请输入500字以内的描述

0 / 500

* 推荐服务:

请选择推荐服务

▼

创建

3 接入数据

接入数据是使用推荐服务的前提，开发者需要在数据接入环节接入三种数据：物料数据、行为数据和用户数据。每种数据都规定了需要传入的字段，开发者根据系统定义的字段名称和要求上传数据，否则将无法正常使用推荐服务。

百度智能云 智能推荐引擎

应用管理 服务监控 推荐服务

帮助文档

< 返回

配置应用

发布到生产环境 模拟接口DEMO

基本信息

应用名称: wangnuecong02

物料类型: 商品推荐1.3.0版本

创建时间: 2023-01-17 11:24:35

物料来源: 自有数据

唯一标识: 77ef83aea46c994ff5b7b6665d91a002

环境: 开发测试环境

应用描述: 商品

1 数据接入

2 服务调用

数据API

应用服务

物料数据接入

已接入183条数据

服务监控 物料管理

智能推荐引擎CMS数据接入服务，接收外部请求传入物料数据。物料数据为必需数据，最少传入100条，否则将导致无推荐结果。

接入数据规范:

字段名称	中文名	类型	是否必填	数据样例	备注
id	商品唯一标识	string	是	NO_1001	一条商品的唯一标识。商品id不能重复
title	商品标题	string	是		商品标题。主要用来进行语义分析提取物料特征
publish_time	商品上架时间	long	是	1648561924764	商品上架时间。可以用于最新商品召回的策略。13位时间戳
ts	上传时间	long	是	1648561924764	商品上传时间。格式: 13位时间戳
status	商品状态	int	是	1	商品上下架状态。0 (下架) 、1 (上架) ，只有上架状态的商品可能被推荐。建议上架状态的商品占比不低于60%

展开非必填字段

数据接入方式:

请求地址

CURL

Python

Java

数据接入服务调用指南

更多接口使用说明，请查阅 [API接口文档](#)

第1步: 应用服务

点击左键按钮一键开启3个数据接入服务。

第2步: 编写/获取代码

选择适合您的数据接入方式: 您可以使用CURL快速接入验证服务，也可以选择相应语言复制代码示例或者编写代码。

第3步: 替换代码中的参数

复制上方基本信息中数据接入和数据API的URL，并参考代码示例进行替换。

第4步: 执行接入数据

返回"success":true即表示数据接入成功，数据接入成功。接入数据成功。接入成功数据即可

测试数据样例

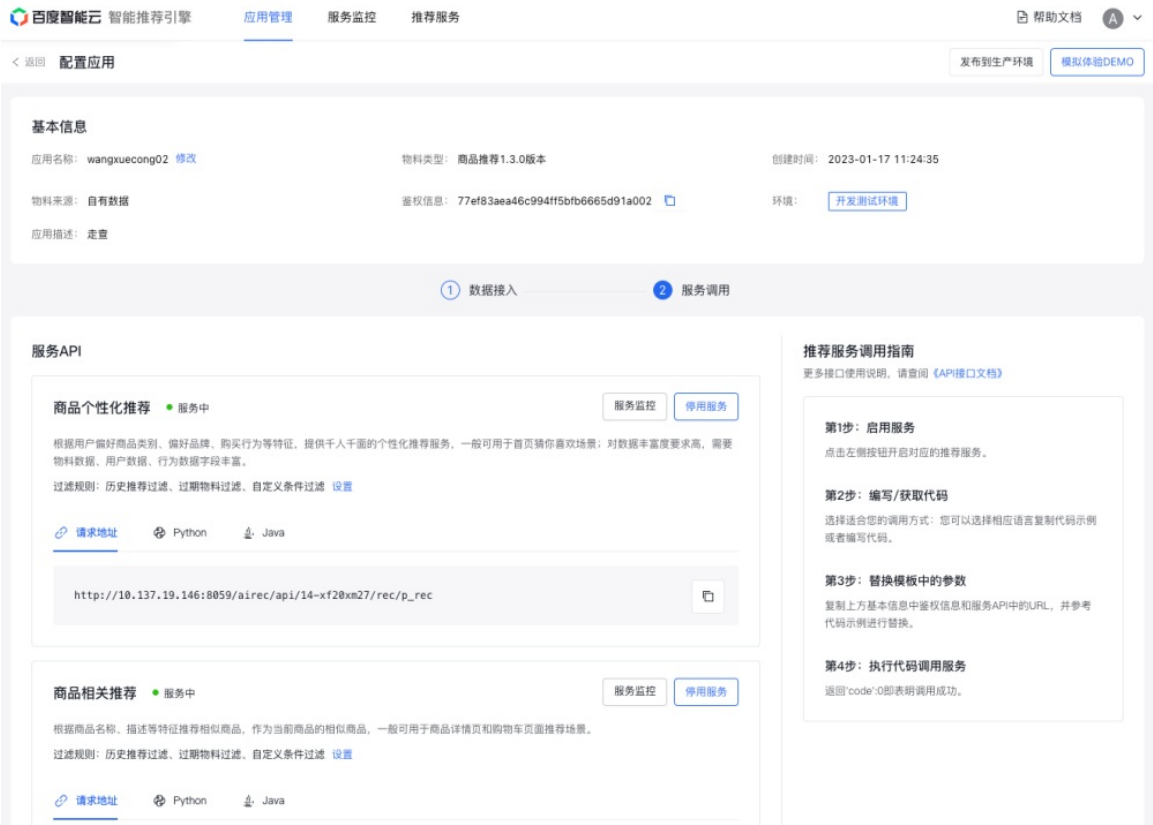
您可以参考如下数据样例，测试构建推荐应用。

1

A_CMSdemo.csv

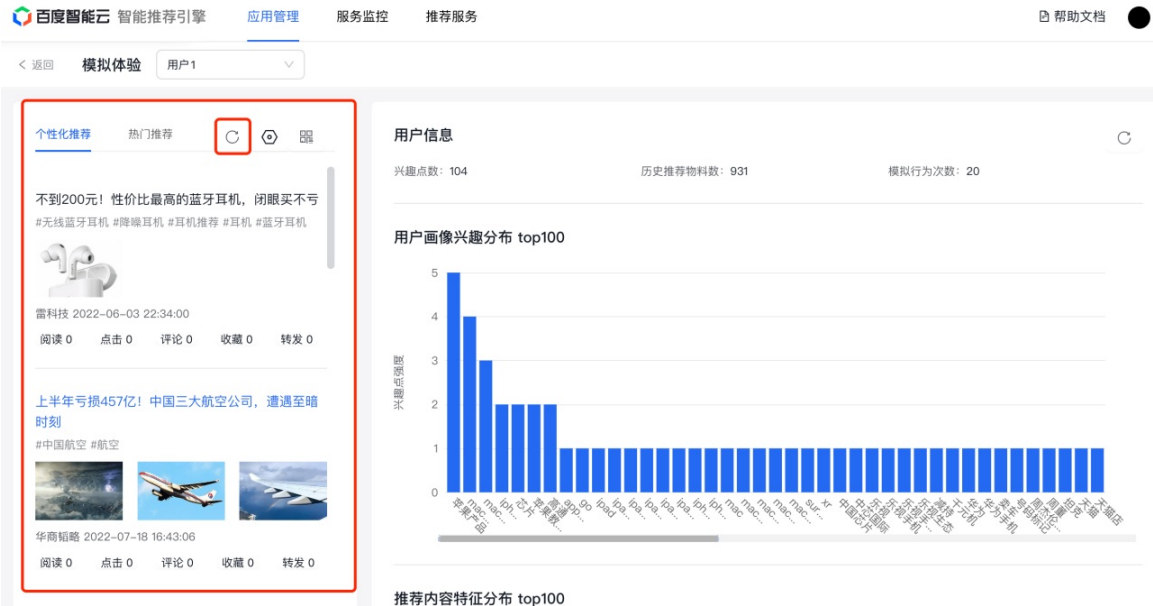
4 调用服务

数据接入后，开发者启动推荐服务，通过配置流量控制参数确定所需资源，一键启动推荐服务，启动服务后开发者可根据系统使用指南调用接口，并根据接口文档说明对API进行调试。同时，开发者可以在此节点查看监控服务运行状态。



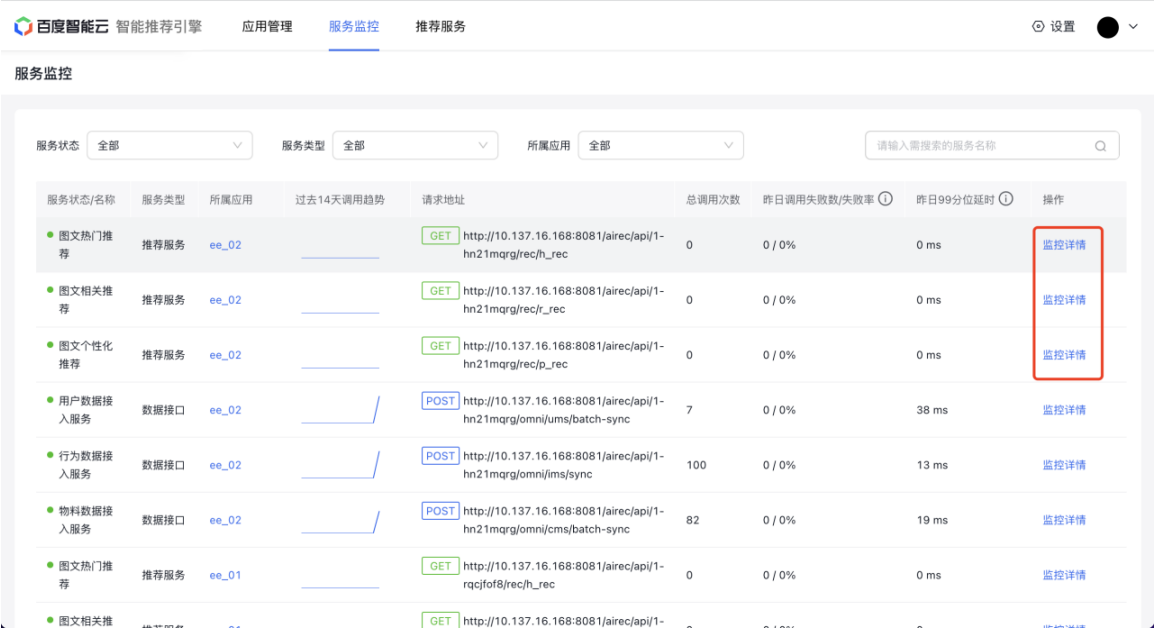
5 模拟体验

基于应用API生成的 PC端演示Demo，包括给定用户查看用户画像，生成物料数据，用户和点击数据，实时生成个性化推荐和相关推荐，支持给定内容查看内容理解中间标签结果，单次推荐结果的可解释性；手机扫码体验推荐效果，并可以在PC端演示的 Demo看到统计数据的变化。



6 监控报表

开发人员或者运维人员通过监控报表了解推荐服务健康度。全方位的服务监控包含了接口调用、接口异常、服务健康度、用户交互等统计数据，保障核心指标数据监控无死角。



以上就是使用智能推荐引擎的主要流程，详情请翻阅下文具体

操作指南

数据仓库

数据管理

概述

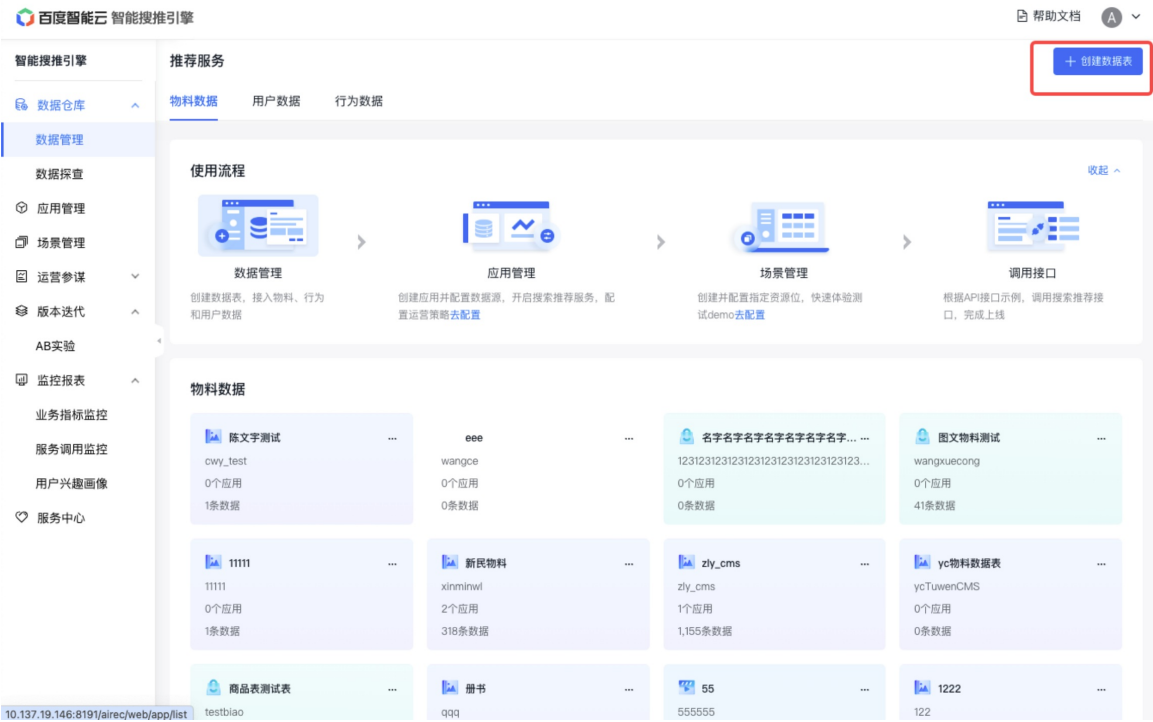
开发者创建数据表接入物料数据、行为数据和用户数据。解决用户使用过程中面临升级应用需要重复上传多次数据的情况，同时在银行客户不允许起多个服务，客户存在使用成本高、升级切换应用难、服务维护成本高、数据重复存储占用资源的问题，因此统一数据接入。

前提条件

有权限登录进入百度智能搜推引擎。

操作步骤

1登录进入智能推荐引擎，点击【+创建数据表】，弹出新建数据表弹框。



2在弹框内填写表中英文名称，选择接入的数据类型【物料数据】或【用户数据】或【行为数据】，和对应的数据表模版。

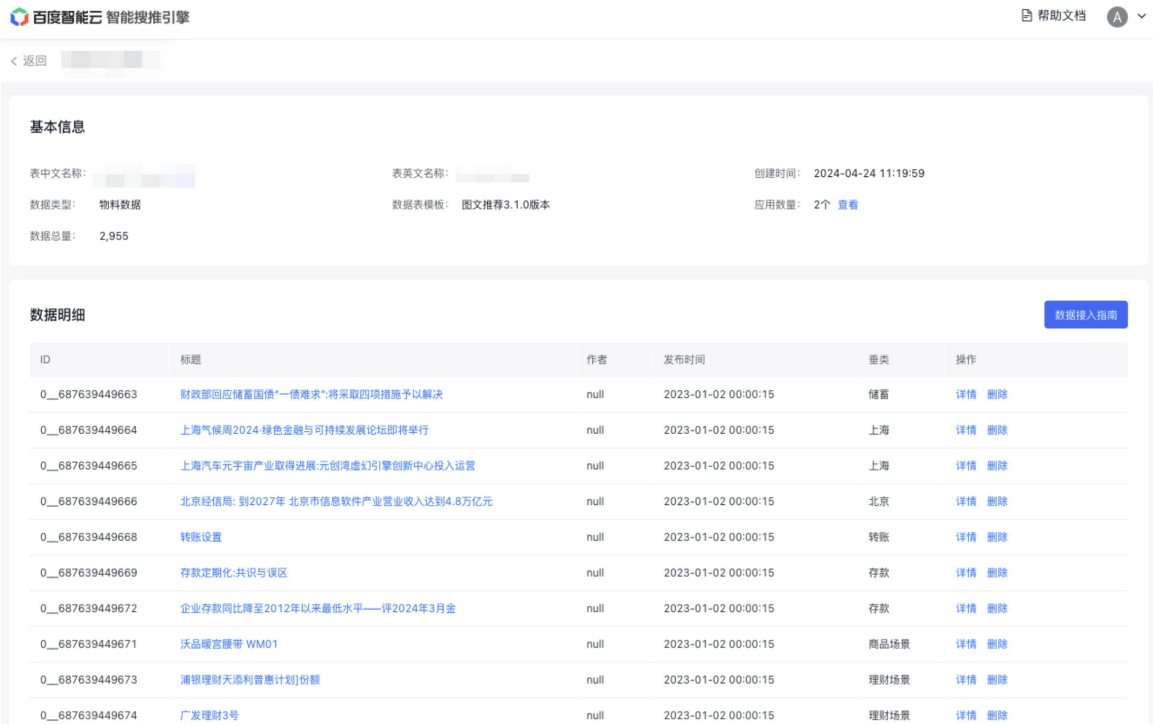


3可在卡片列表中查看用户数据、行为数据和物料数据卡片，进行查看详情和删除操作。

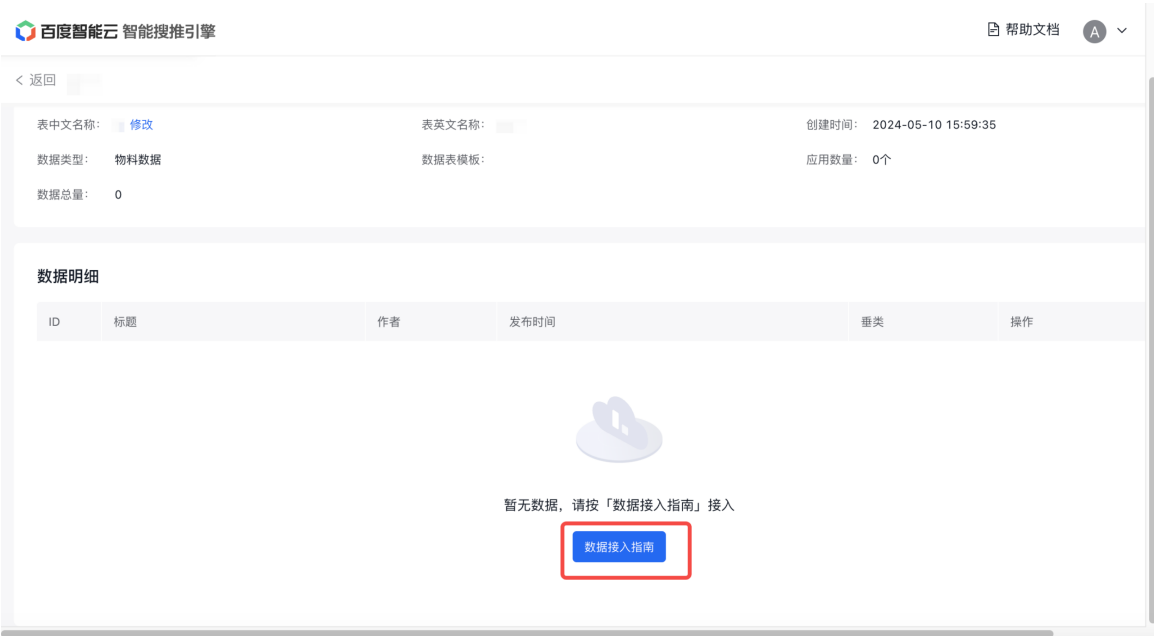


说明：
如果该数据表没有被引用，则该数据表可以删除；若已引用，则该数据表不可删除。

4 点击卡片后进入详情页，可查看对应的数据基本信息与数据明细，可对每条数据进行查看详情和删除的操作。



5 点击【数据接入指南】按钮，查看数据接入指南，按操作指引接入相关数据。



数据接入

概述

数据接入是使用推荐服务的前提，开发者需要在数据接入环节接入三种数据：物料数据、行为数据和用户数据。每种数据都规定了需要传入的字段，开发者根据系统定义的字段名称和要求上传数据，否则将无法正常使用推荐服务。

前提条件

- 1已创建好三种物料数据表。
- 2已经准备好要接入的物料、用户和行为数据。
- 3网络畅通：如果您是百度公有云的用户，需要保证您的数据可以发送到互联网；如果您是私有化部署的用户，需要保证您的数据和本系统网络是连接畅通的。

操作步骤

1登录进入智能推荐引擎,按[数据管理](#)操作，已建立好【数据表】，并进入【数据接入指南】页面，以下操作以【物料数据】为例：

百度智能云 智能搜推引擎

帮助文档

数据接入指南

数据API

物料数据接入

服务中 已接入317条数据

CMS 数据入库服务,接收外部请求传入物料数据

数据接入方式:

请求参数

CURL

Python

Java

CSV/Excel

请求地址: http://10.137.19.146:8059/airec/api/omni/cms/batch-sync

鉴权信息ak: 8873f6e0663593a09cc5fbad96d0d86c

表英文名: h6310v-l-27

数据接入规范:

字段名称	中文名	类型	是否必传	数据样例
nid	物料唯一标识	string	是	NID_1001
title	物料标题	string	是	物料标题, 主要用来进行语义提
content	文章正文	string	是	示例文章正文
publish_time	发布时间	long	是	1648561924764
ts	上传时间	long	是	1648561924764
status	状态	int	是	1

展开非必填字段

数据接入服务调用指南

更多接口使用说明, 请查阅《API接口文档》

第1步: 编写/获取代码

选择适合的数据接入方式: 可以使用CURL快速接入验证服务, 也可以选择相应语言复制代码示例或者编写代码

第2步: 替换模板中的参数

可复制「请求参数」中的URL、鉴权信息、aliasName, 并参考代码示例进行替换

第3步: 执行接入数据

返回'success':true即表明接入成功, 刷新数据状态, 接入必传数据即可

测试数据样例

您可以参考如下数据样例, 测试构建推荐应用。

A_CMSdemo.xlsx

A_IMSdemo.xlsx

A_UMSdemo.xlsx

2/准备数据：根据数据结构规范准备数据，如果您的数据字段名称和取值范围不符合规范，需要您做数据预处理。

百度智能云 智能搜推引擎

帮助文档

数据接入指南

数据API

请求地址: http://10.137.19.146:8059/airec/api/omni/ms/batch-sync

鉴权信息ak: ba09402842b7b64eb938159f1bee00c8

表英文名: rzp7egb4-21

数据接入规范:

字段名称	中文名	类型	是否必传	数据样例
nid	商品唯一标识	string	是	NID_1001
title	商品标题	string	是	商品标题, 主要用来进行
publish_time	商品上架时间	long	是	1648561924764
ts	上传时间	long	是	1648561924764
status	商品状态	int	是	1

展开非必填字段

第1步: 编写/获取代码

选择适合的数据接入方式: 可以使用CURL快速验证服务, 也可以选择相应语言复制代码示例或者编写代码

第2步: 替换模板中的参数

可复制「请求参数」中的URL、鉴权信息、aliasName, 并参考代码示例进行替换

第3步: 执行接入数据

返回'success':true即表明接入成功, 刷新数据, 接入必传数据即可

测试数据样例

您可以参考如下数据样例, 测试构建推荐应用。

g_cms.xlsx

g_ims.xlsx

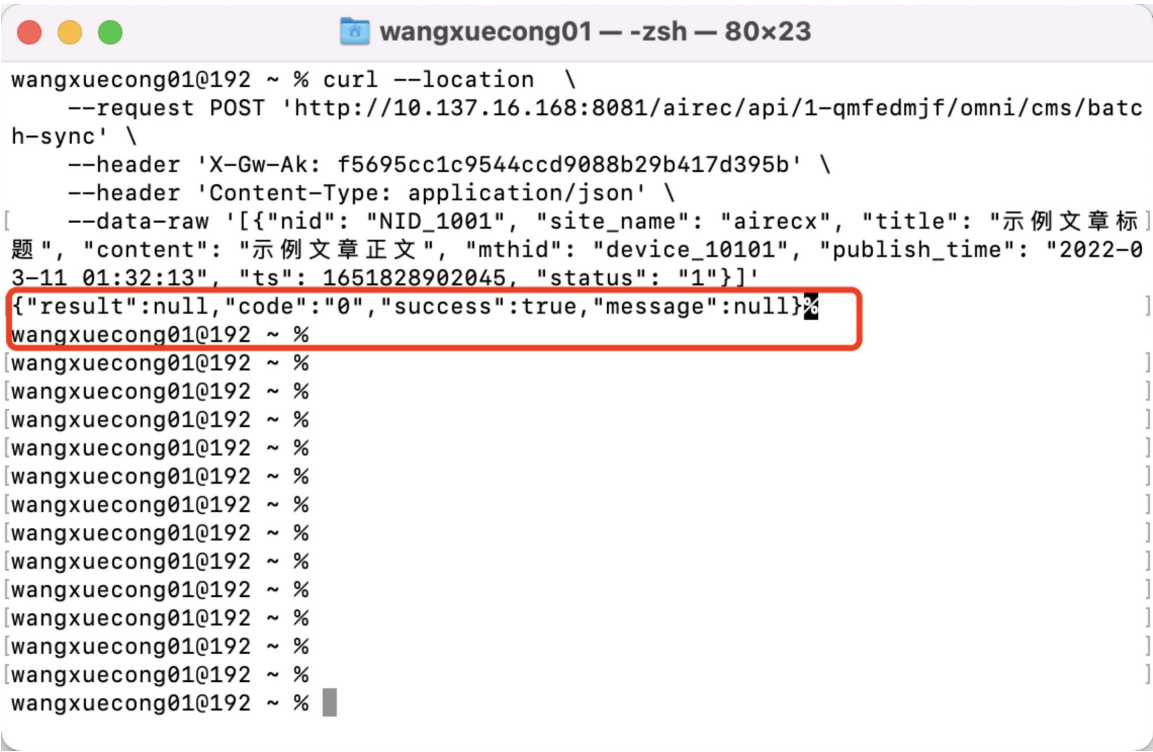
g_ums.xlsx

3 接入数据

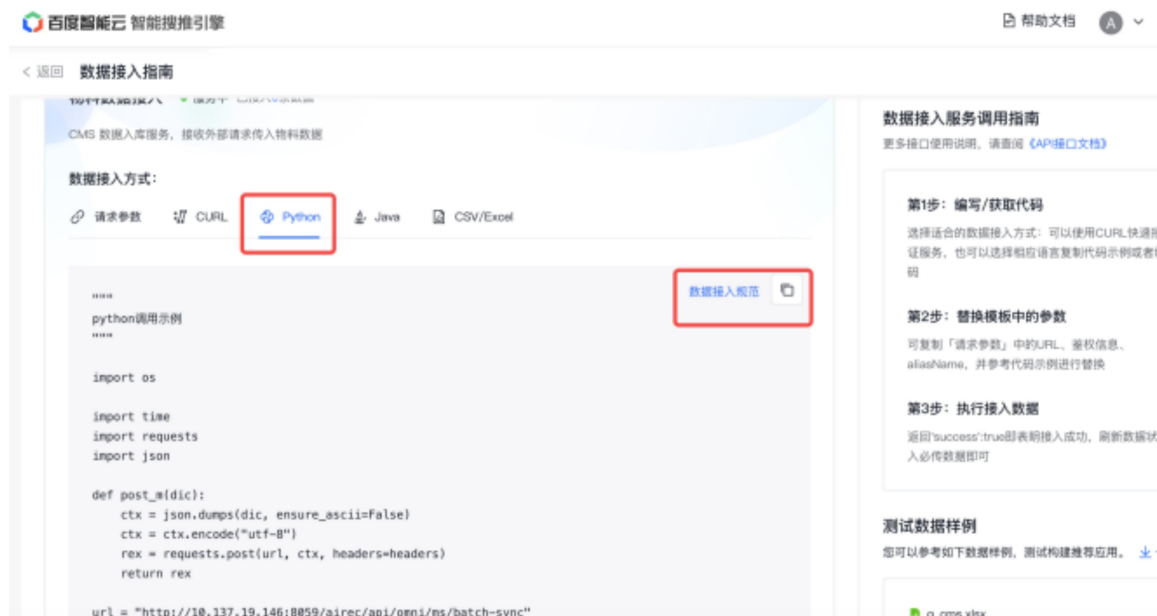
(1) 方法1:使用curl命令快速接入验证，copy示例代码。



在终端中执行，返回如下状态时，表示数据接入成功。



(2) 方法2:使用对应语言的脚本

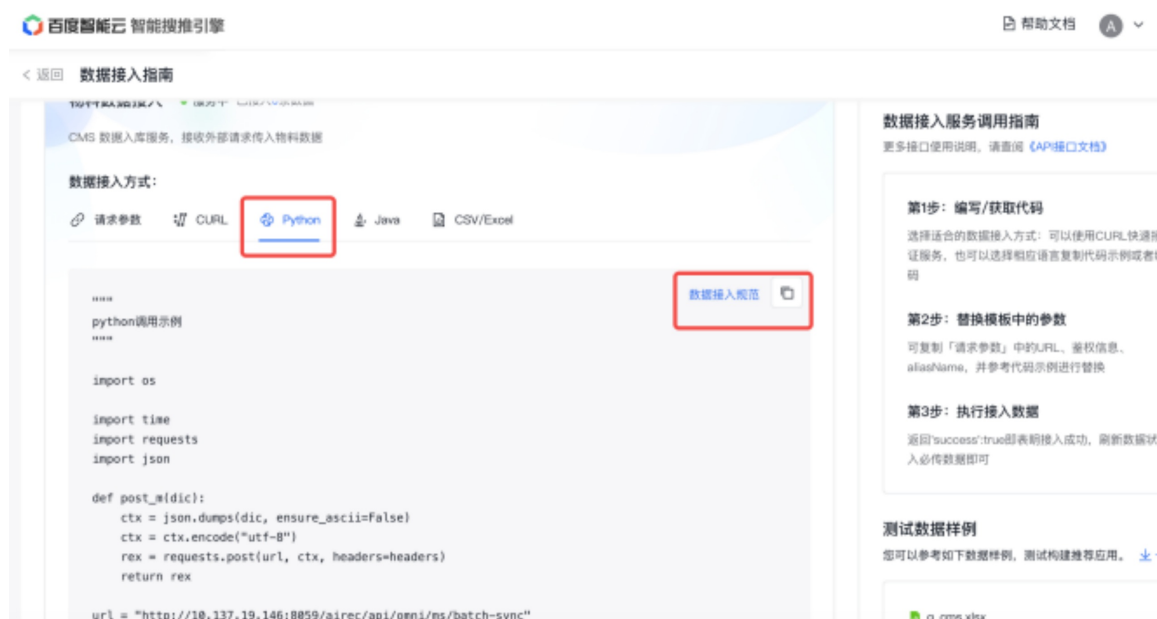


第1步：编写/获取代码模版，选择相应语言复制代码示例或者编写代码。

第2步：替换模板中的参数。复制基本信息中鉴权信息和数据API中的URL，并参考代码示例进行替换。

第3步：接入数据，返回'success':true即表明接入成功，刷新数据状态，接入必传数据即可。

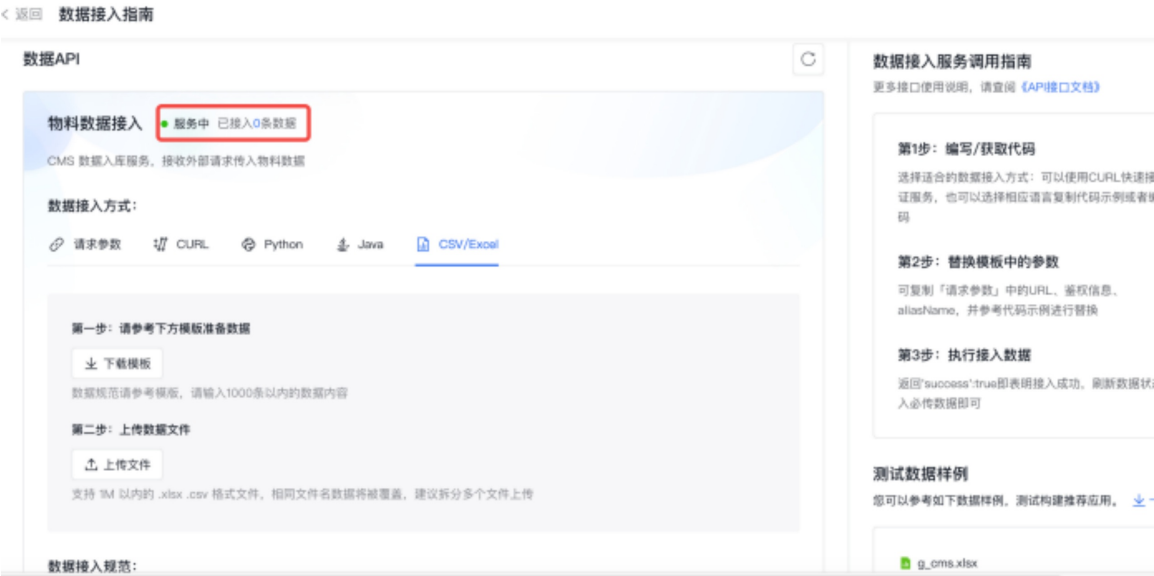
(3) 方法3:上传excel接入数据



第1步：点击【下载导入模板】按钮，使用下载的模板填充符合格式的数据。

第2步：点击【上传文件】按钮，选择文件目录，将包含数据的模板文件上传。支持多次上传，同一id的物料，最后上传的覆盖更新。支持文件类型：excel、csv。

4验证是否成功接入数据，在配置应用页面基础信息卡片查看数据接入条数是否符合预期。



数据探查

概述

数据探查是数据质量保障非常重要的一步，它是数据开发和模型训练的基础。数据探查通过自动化的手段了解数据结构，检查数据成分、数据关系及数据格式等问题，旨在了解来源数据的数据形态，帮助分析、判断、找出潜在的数据问题和风险。

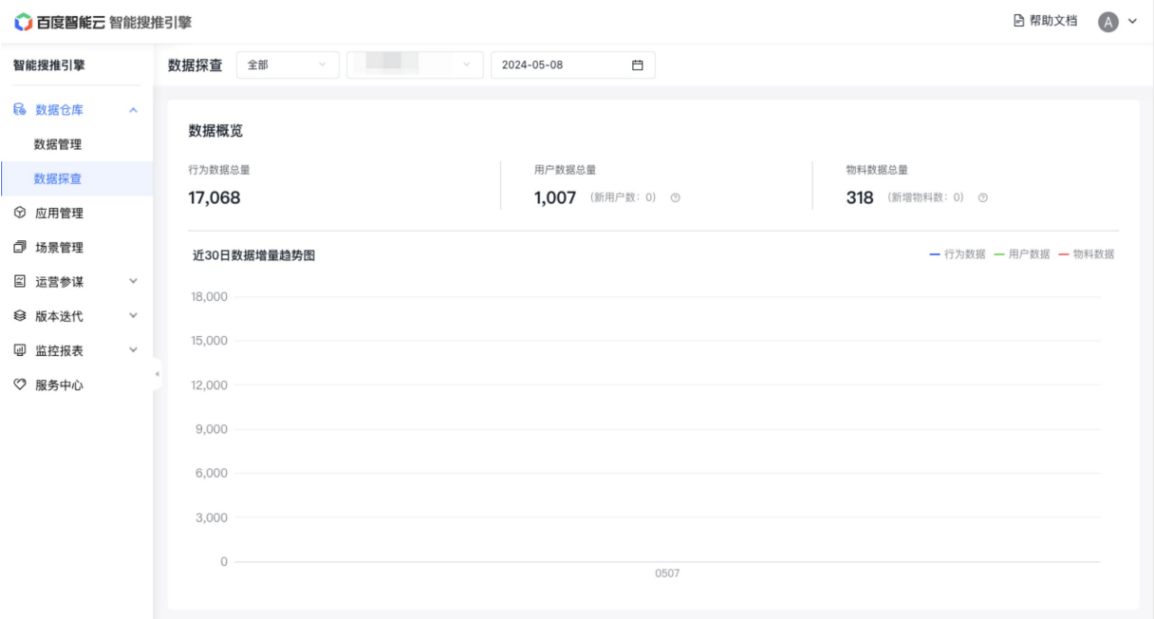
前提条件

- 1应用已经创建完成。
- 2数据接入服务和推荐服务已经开启。

操作步骤

1登录进入智能推荐引擎，查看数据探查的方式有3种：

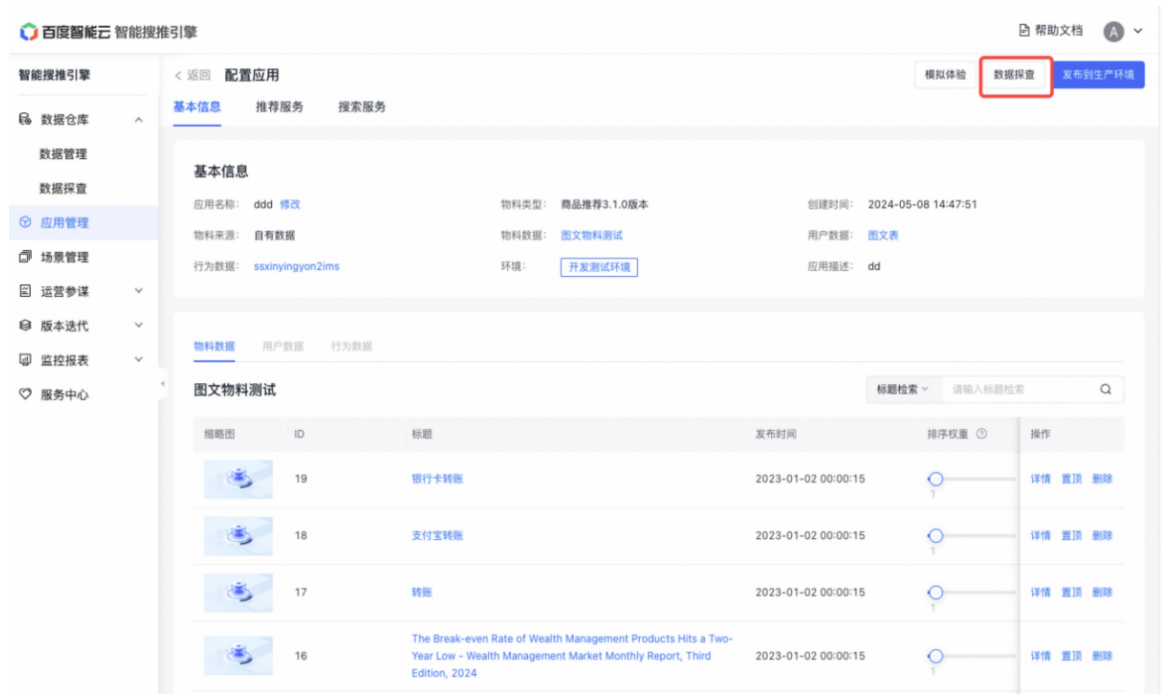
- (1) 【数据探查】页面中查看：在【数据仓库】下的二级页面【数据探查】中查看，进入数据探查页面。



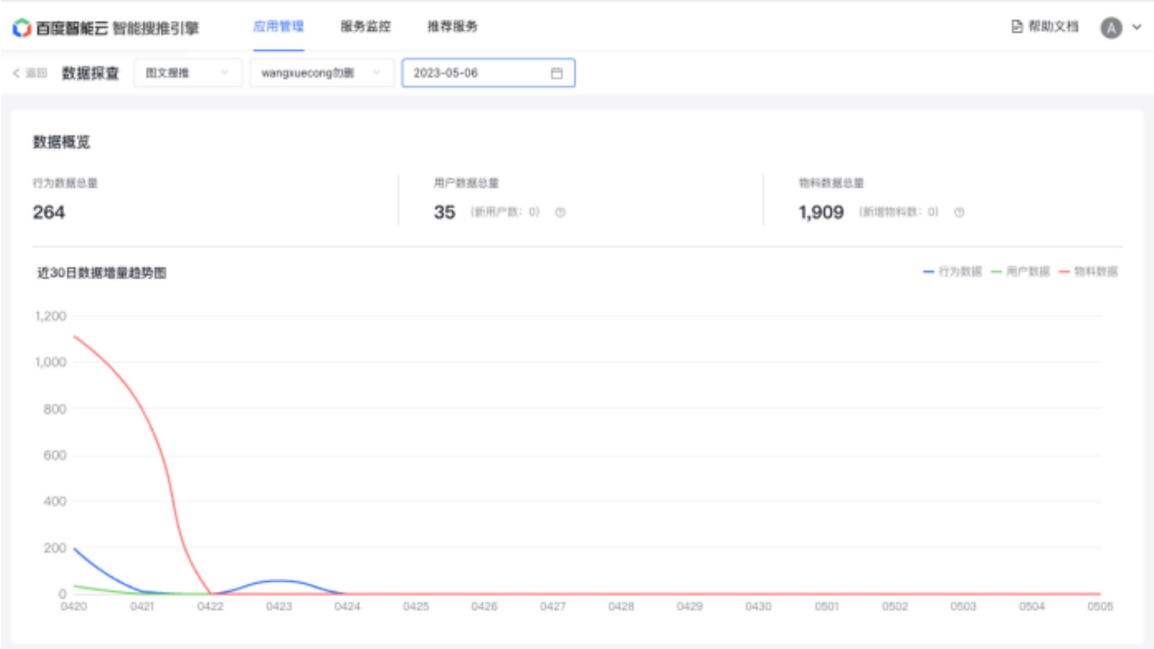
- (2) 【服务中心】精选服务中查看：在精选服务列表页面点击“数据探查”卡片，进入数据探查页面。



(3) 【应用管理】中查看：在应用列表页面点击【配置】按钮进入应用页，点击“数据探查”按钮，进入数据探查页面。



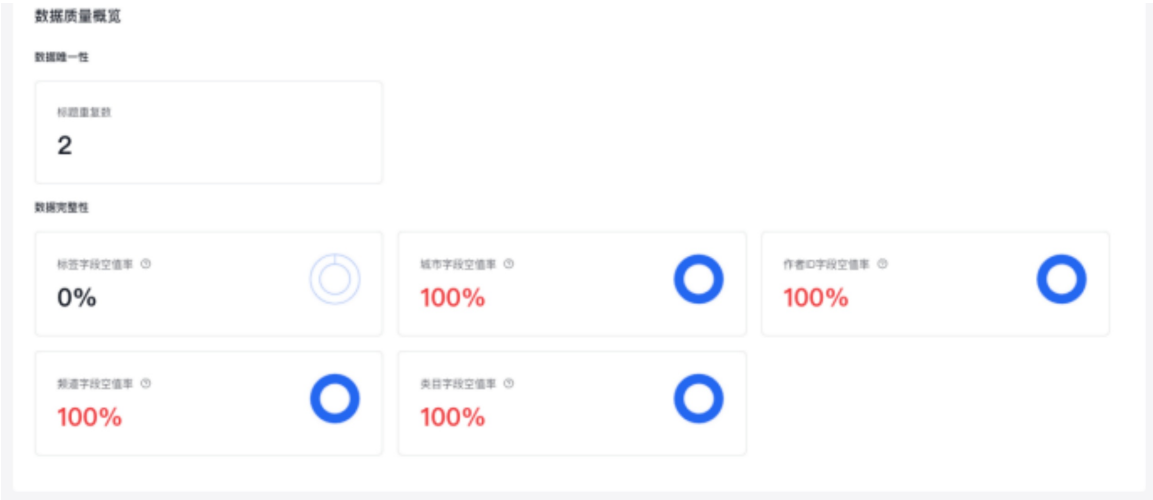
2查看用户数据、行为数据和物料数据概况。



参数	参数说明
三类数据总量统计	分别统计三类数据从接入到当前的数据总量。
三类数据每日增量统计	近30日数据增量趋势图，统计颗粒度为天，含今日数据。
查询条件	根据应用和截止时间（天粒度）查询，暂只支持图文物料。
统计时效性	离线t-1，天粒度
新用户	2个月内无点击行为的用户。
新物料	新物料：1天内上传的物料。

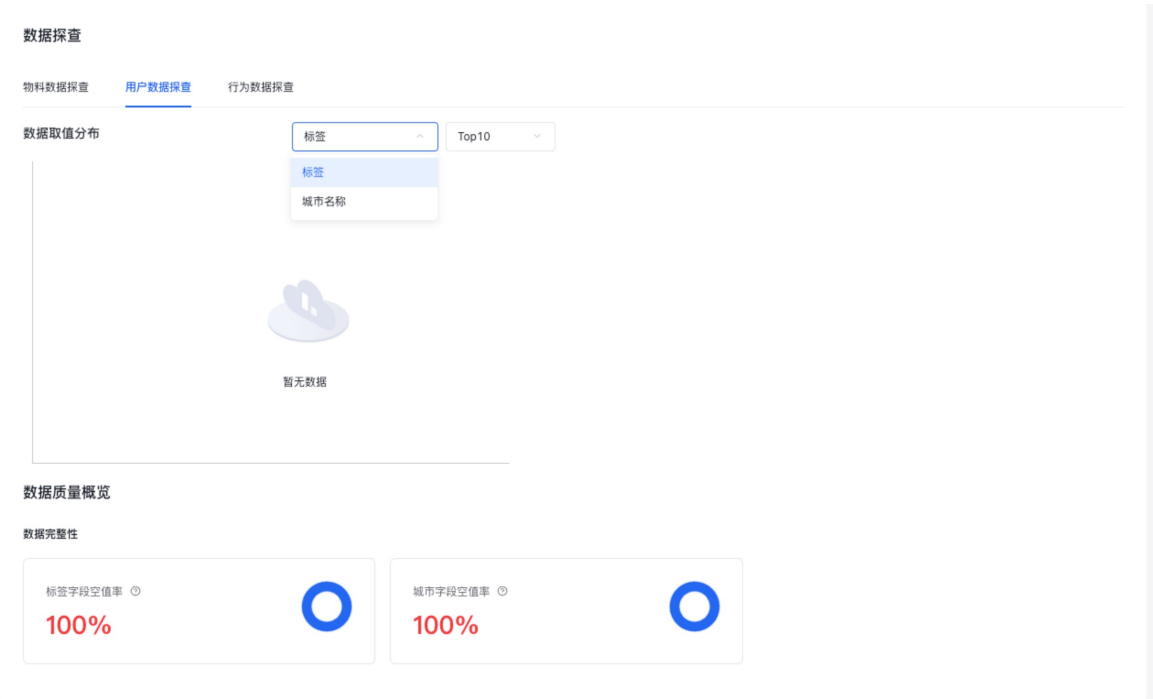
3物料数据探查：分析物料数据数值分布和数据质量指标。





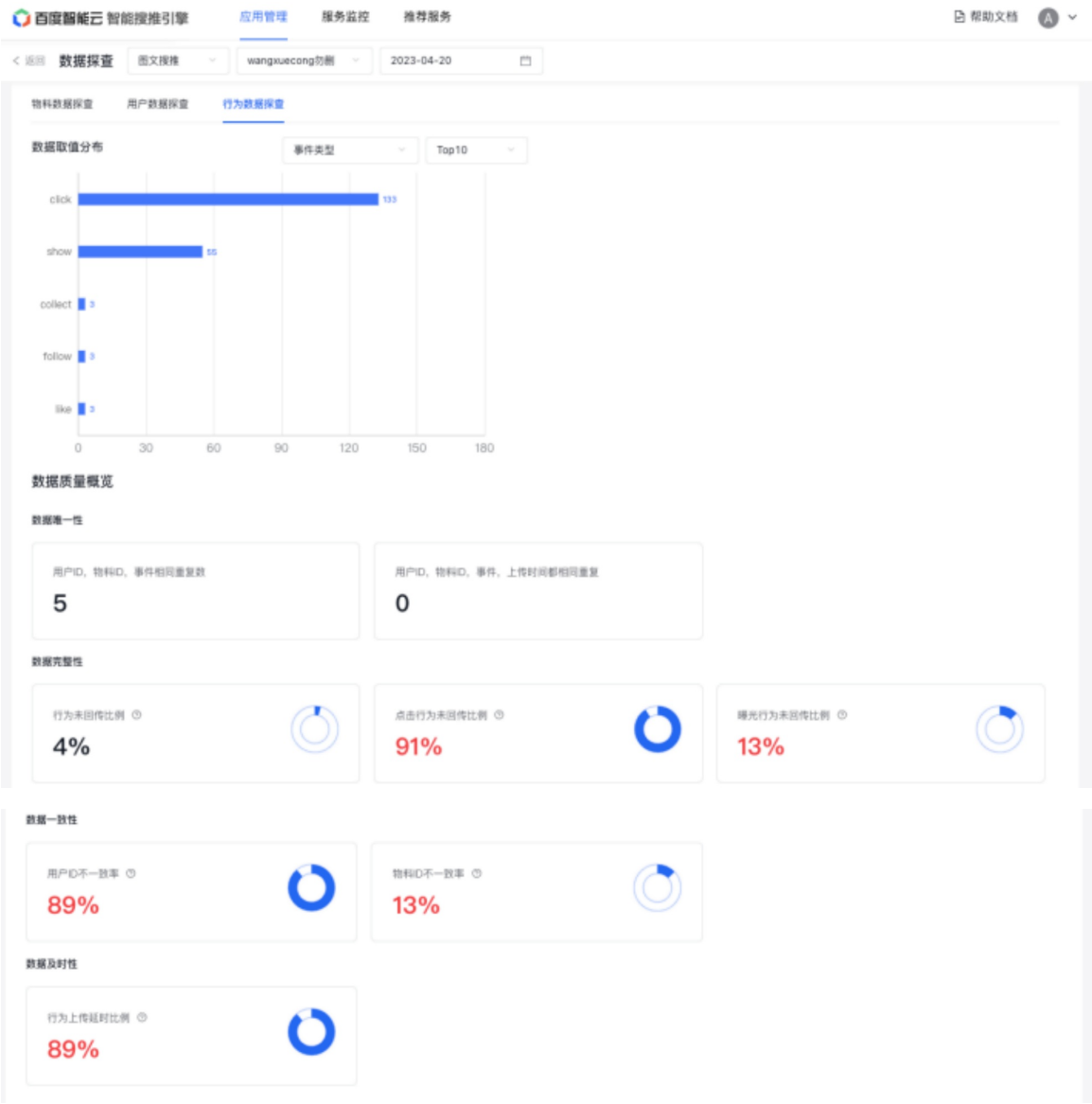
参数	参数说明
空值数	统计非必填字段空值数，图文主要统计author_id、channels、tags、category的空值数。
重复数	图文主要统计nid重复数、标题重复数、nid和标题重复数。
取值分布统计维度	图文主要统计物料上架状态、作者、一级类别、标签、频道的取值分布，可选择top10、20、50，统计每个取值的数据量和占比，并按照倒序进行排序。
标签字段空值率	存在x%的标签字段(tags) 缺失，标签特征发挥作用削弱，缺失该值的物料无法根据标签被召回，将会影响该物料的曝光，建议不要为空。
城市字段空值率	存在x%的城市字段(city) 缺失，城市特征发挥作用削弱，缺失该值的物料无法根据城市被召回，将会影响该物料的曝光和本地服务，建议不要为空。
作者id字段空值率	存在x%的作者id字段(author) 缺失，作者特征发挥作用削弱，缺失该值的物料无法根据相同作者的物料被召回，将会影响该物料的曝光，建议不要为空。
频道字段空值率	存在x%的频道字段(channels) 缺失，频道特征发挥作用削弱，缺失该值的物料无法根据频道筛选过滤，将会影响该物料的筛选，建议不要为空。
类目字段空值率	存在x%的类目字段(category) 缺失，类目特征发挥作用削弱，缺失该值的物料无法根据同类目的物料被召回，将会影响该物料的曝光，建议不要为空。

4用户数据探查：分析用户数据数值分布和数据质量指标。



参数	参数说明
空值数	统计非必填字段空值数，图文主要统计tags、城市的空值数。
重复数	图文主要统计uid重复数。
取值分布统计维度	图文主要统计标签和城市的取值分布；可选择top10、20、50，统计每个取值的数据量和占比，并按照倒序进行排序。
标签字段空值率	存在x%的标签字段(tags) 缺失，标签特征发挥作用削弱，缺失该值的用户无法根据标签进行兴趣召回，将会影响个性化服务效果，建议不要为空。
位置字段空值率	存在x%的位置字段(city) 缺失，城市特征发挥作用削弱，缺失该值的用户无法根据城市进行基于地理位置的兴趣召回，将会影响个性化服务效果和本地服务，建议不要为空。

5行为数据探查：分析数据数值分布和数据质量指标。



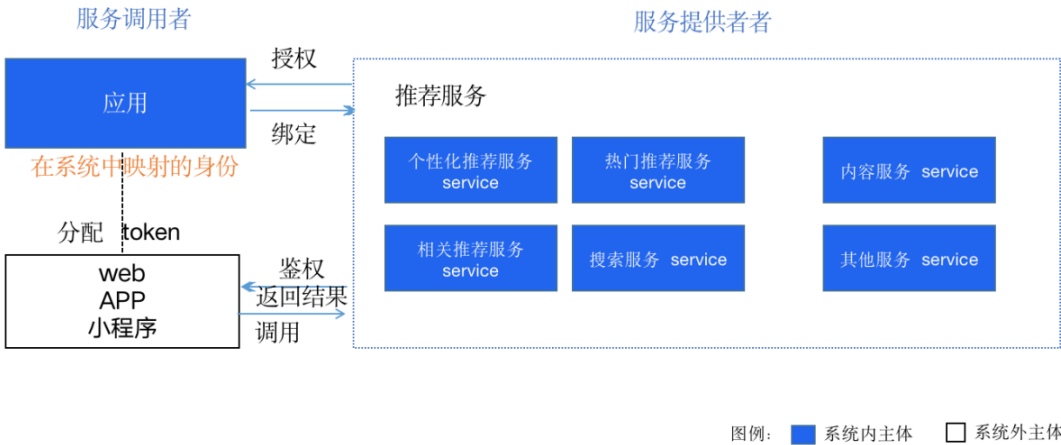
参数	参数说明
重复数	uid, nid, event相同重复数、id, nid, event, ts相同重复数
取值分布统计维度	图文主要统计事件类型和场景的取值分布。可选择top10、20、50，统计每个取值的数据量和占比，并按照倒序进行排序。
行为未回传比例	用户行为反馈合理性判断指标：存在0%的用户请求了推荐结果，但无后续的反饋行为（回传行为量较少，影响模型学习效果；当比例偏差较大时，请检查埋点是否正常;数据链路是否正常，是否存在数据漏传;请检查是否存在较多的缓存)
曝光行为未回传比例	曝光行为反馈合理性：存在0%的用户请求了推荐结果或者存在点击行为，但无对应的曝光行为回传。（曝光行为回传行为量较少，影响计算ctr预估值，影响模型学习效果；当比例偏差较大时，需要检查埋点是否正常;数据链路是否正常，是否存在数据漏传;请检查是否存在较多的缓存;)
点击行为未回传比例	点击行为反馈合理性：存在0%的用户请求了推荐结果，但无对应的点击行为回传。（曝光行为回传行为量较少，影响计算ctr预估值，影响个性化效果，影响模型学习效果；当比例偏差较大时，需要检查埋点是否正常;数据链路是否正常，是否存在数据漏传;请检查是否存在较多的缓存;)
物料id不一致率	行为表物料id在物料表中不存在率
用户id不一致率	行为表用户id在用户表中不存在率
行为上传时间延时比例	行为上传时间与行为发生时间的差值大于30min的数据所占的比率

应用管理

创建应用

概述

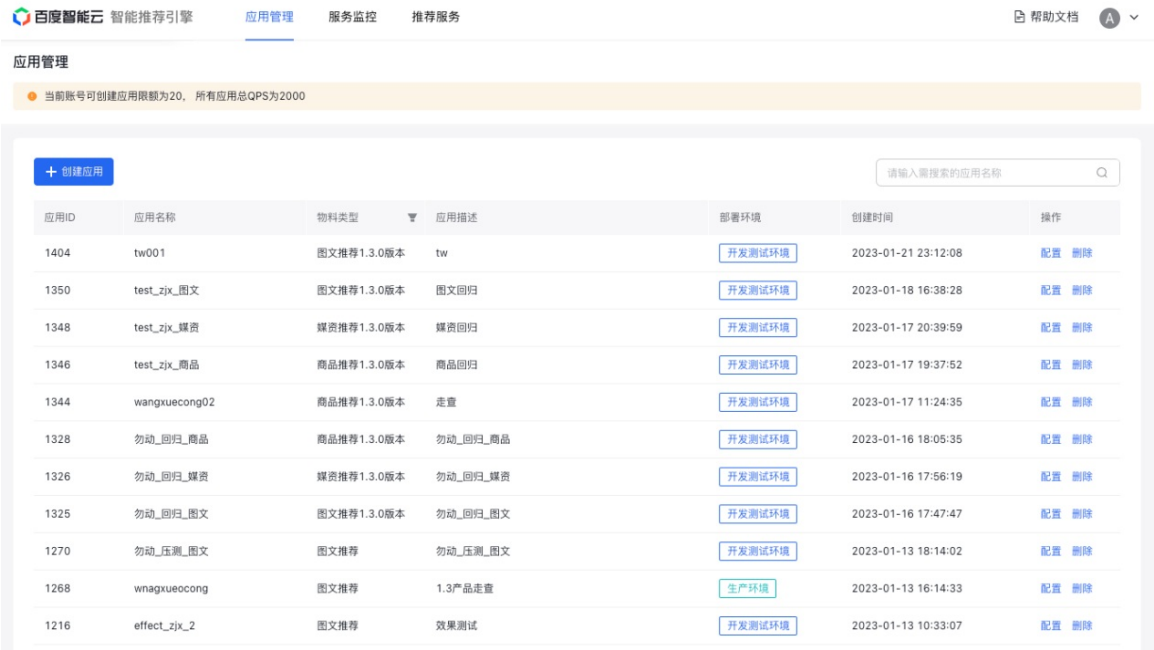
新建应用时定义服务消费者在推荐系统中映射的身份，同时定义使用相同物料数据和用户数据结构的服务集合。



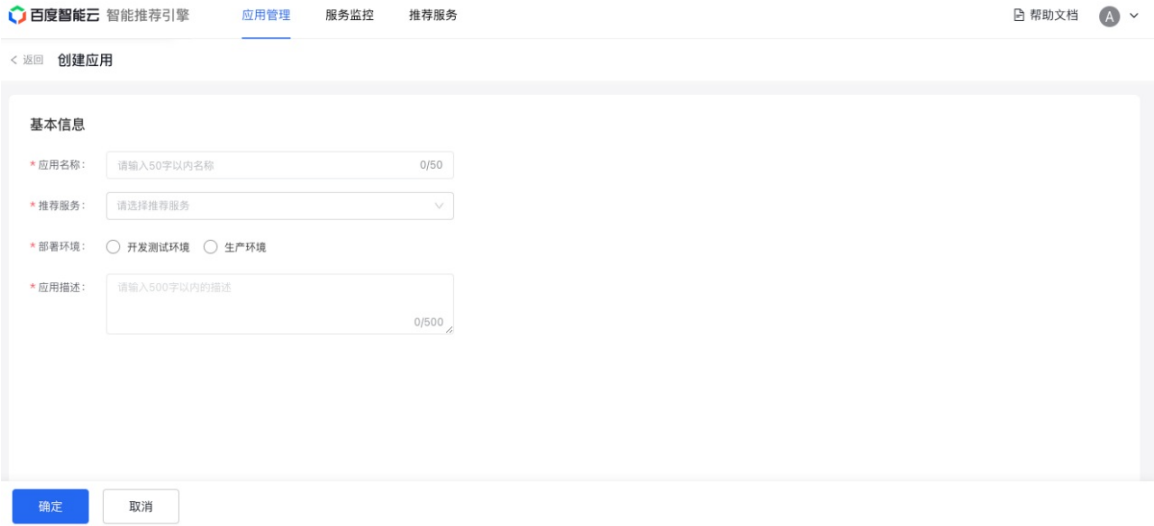
前提条件

有权限登录进入百度智能搜推引擎，同时具有应用创建权限。

操作步骤 1进入应用管理页面。



2 如果用户权益余额充足，单击顶部【创建应用】按钮，进入创建应用页面，依次填写应用各项参数，详见参数说明。点击【创建】按钮，进入创建应用提示页面。



应用名称：必填，允许输入中文、英文字母、数字和下划线，限制50个字符以内。名称类格式规范下文不在描述。应用描述：非必填，不支持特殊字符，限制500个字符以内。描述类格式规范下文不在描述。

4在应用创建提示页面，您可以按照下图操作指引进入应用配置页面或者返回应用列表选择下次配置使用应用。



调用服务

概述 数据接入后，开发者启动推荐服务，通过配置流量控制参数确定所需资源，一键启动推荐服务，启动服务后开发者可根据系统使用指南调用接口，并根据接口文档说明对API进行调试。同时，开发者可以在此节点查看监控服务运行状态。

前提条件 1应用已经创建完成，并且已经接入物料、用户和行为数据。 2具有服务调用token权限。

操作步骤 1登录进入智能推荐引擎。

2在应用列表页面选择相应的应用点击【配置】按钮。



3启用服务：进入引用配置页面，在服务调用选项卡下点击推荐API卡片【启用】按钮开启需要调用的服务。



服务的状态显示为“服务中”时，即可根据生成的请求地址调用接口。



4 调用服务，详见《API使用指南》。

- 第1步：编写/获取代码。选择适合您的调用方式：您可以选择相应语言复制代码示例或者编写代码。
- 第2步：替换模板中的参数。复制上方基本信息中鉴权信息和服务API中的URL，并参考代码示例进行替换。 第3步：执行代

码调用服务。返回'code':0即表明调用成功。

兴趣构建（LLM增强版功能）

概述

大模型在用户理解方面具有巨大的潜力，能够提供更深入的理解。我们利用聚类算法将整个物料库聚类为若干集群。使用LLM技术个性化提取兴趣旅程，并结合少样本提示、提示调整和微调等技术来优化模型的性能，使其能够更好地理解和总结用户的兴趣旅程。

前提条件

- 1应用已经创建完成，并且已经接入物料、用户和行为数据。
- 2具有已开启大模型权限。

操作步骤

1登录进入智能推荐引擎，进入【应用配置】接入数据--【兴趣构建】页面。系统会每天例行更新聚类，初次聚类需要物料数量大于30条，如果已满足请点击【立即聚类】。建议初始化物料数据接入完成后执行此操作，这样生成的聚类更全面。



2聚类过程大约需要10分钟左右，请您耐心等待。聚类完成后显示兴趣簇矩阵，矩形大小代表了聚类数量多少。



3点击某个矩形，显示该聚类下的文章列表和细化的兴趣点。



概念名称	概念解释
兴趣簇	由文章物料聚类而成，由大模型根据文章标题生成兴趣簇的名称。矩形大小代表聚类物料数量。
兴趣点	由大模型来抽取兴趣簇内文章的特点作为兴趣点。

查看模拟用户画像

概述

根据模拟用户行为和历史推荐结果可以在PC端演示的Demo看到统计数据的变化，以便直观的体验推荐效果

前提条件

- 1租户完成应用创建。
- 2以下三种情况出现任何一个都无法使用：

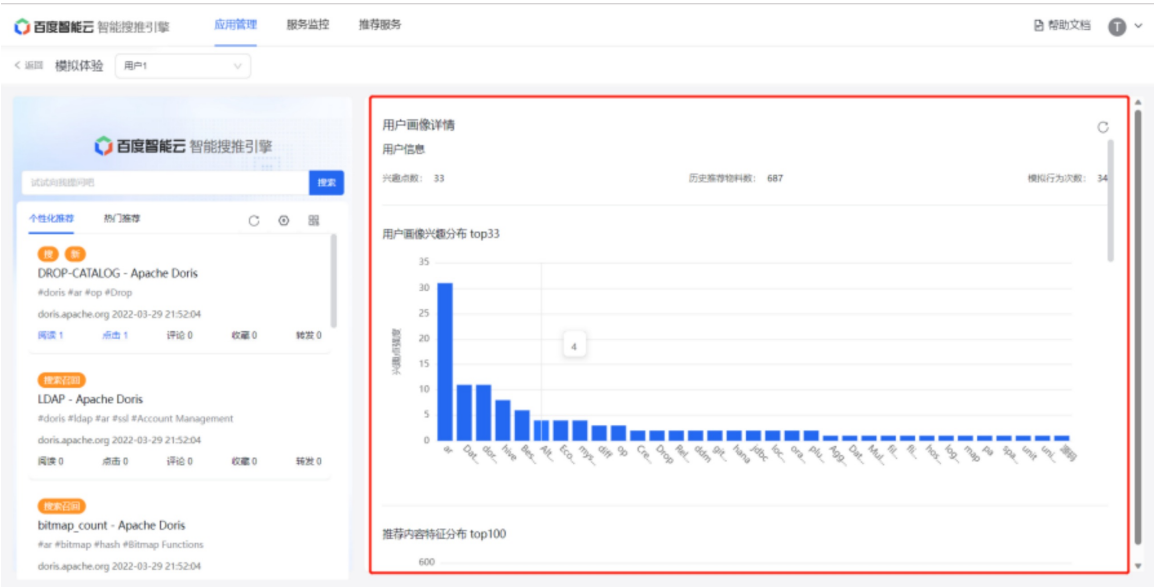
(1) 数据服务未部署

(2) 物料数据未接入

(3) 数据服务未启动完成
- 3个性化推荐服务、热门推荐服务至少开启1个。

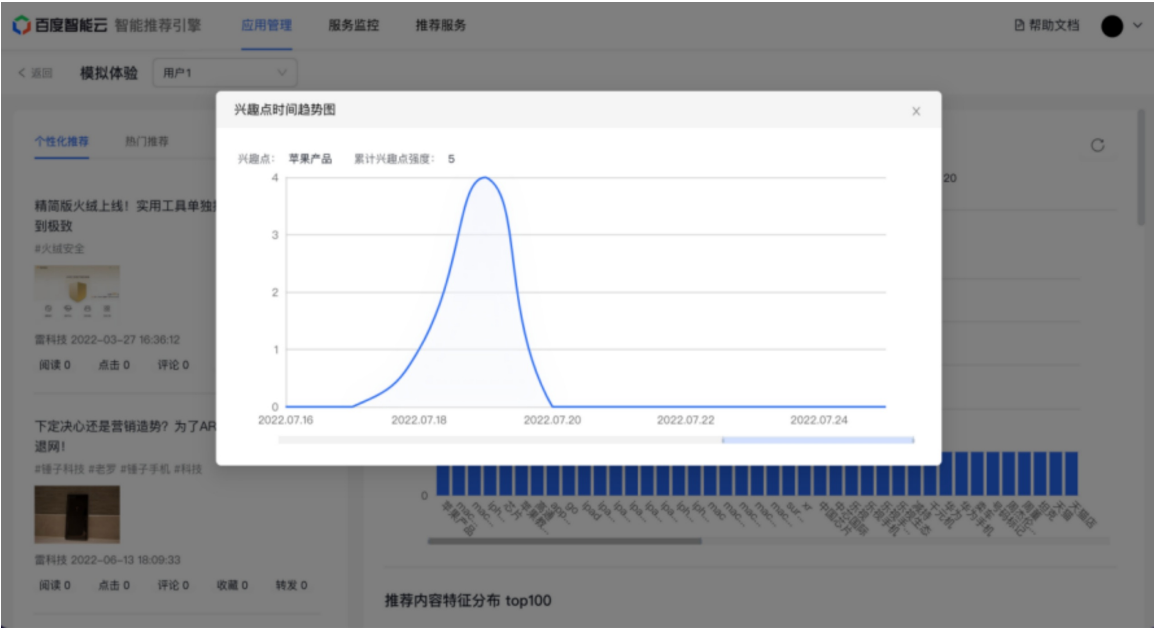
操作步骤

1登录进入智能推荐引擎，【应用列表】选择相应的应用--点击【配置】按钮，切换到服务调用页面，点击【模拟体验】页签进入用户画像。

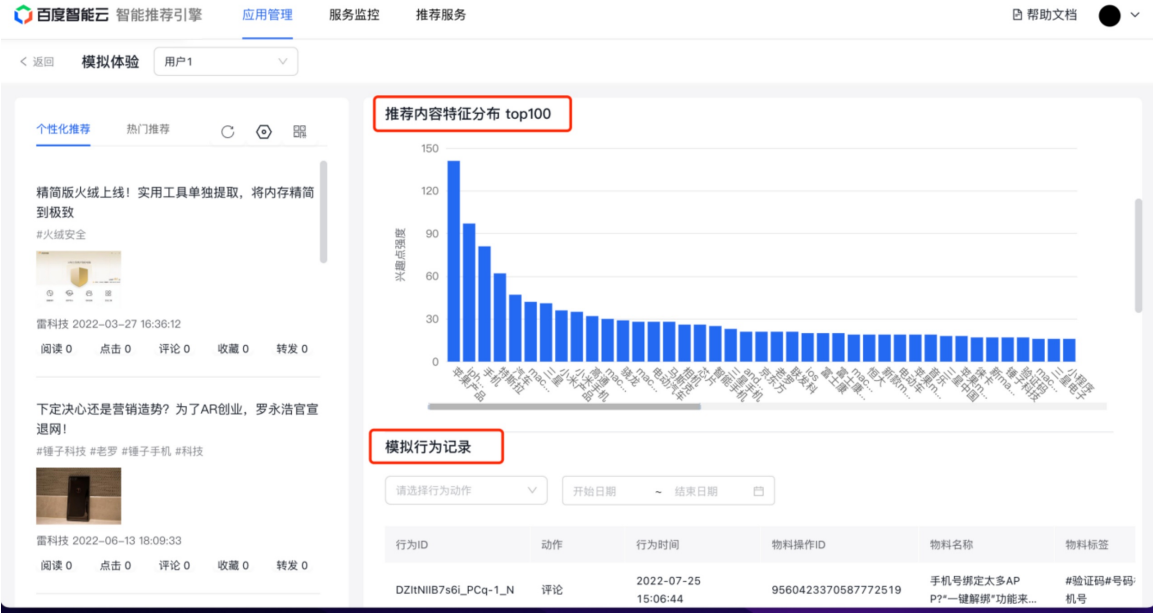


参数	参数说明
兴趣点数	统计用户下所有标签数。
兴趣点强度	感兴趣（有过行为）的物料按照标签粒度聚合的数量。
模拟行为次数	某个用户在pc端和手机端模拟行为的次数。
历史推荐物料数	某个用户在pc端和手机端推荐的物料数。

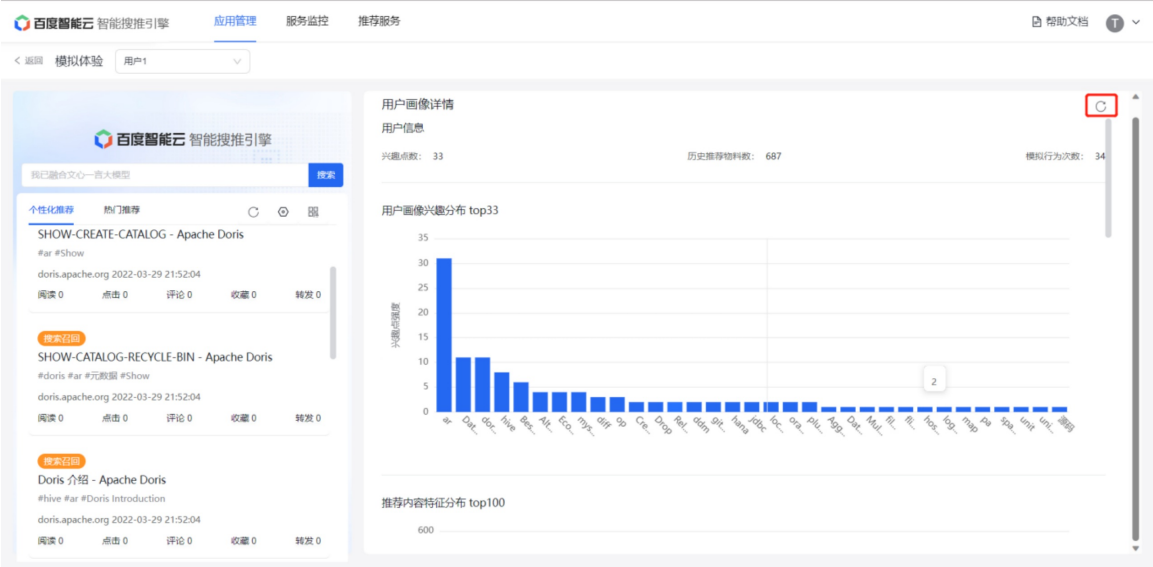
2查看用户兴趣点变化趋势：点击模拟用户的兴趣点可以查看该兴趣点（标签）在每天的变化趋势。



3查看推荐内容分布和用户兴趣分布是否吻合，或者查询模拟行为记录。



4. 点击【刷新】按钮，刷新用户画像。



说明：

问题：为什么有些文章我模拟了行为，却没有显示该兴趣点？

可能的原因：

这种情况主要是由于该内容过短或者其他原因导致没有成功的从文章中提取出标签，导致这篇文章没有统计到。

模拟体验PC端demo

概述

百度智能搜推引擎基于应用中自动生成的API生成的 PC端演示demo和手机端demo，这里以图文推荐demo为例，其他物料的推荐demo类似。基于应用中自动生成的API生成的 PC端演示Demo，支持给定模拟用户查看用户画像，生成点击等行为数据，即时体验个性化推荐和相关推荐效果，支持给定内容查看内容理解中间标签结果。

前提条件

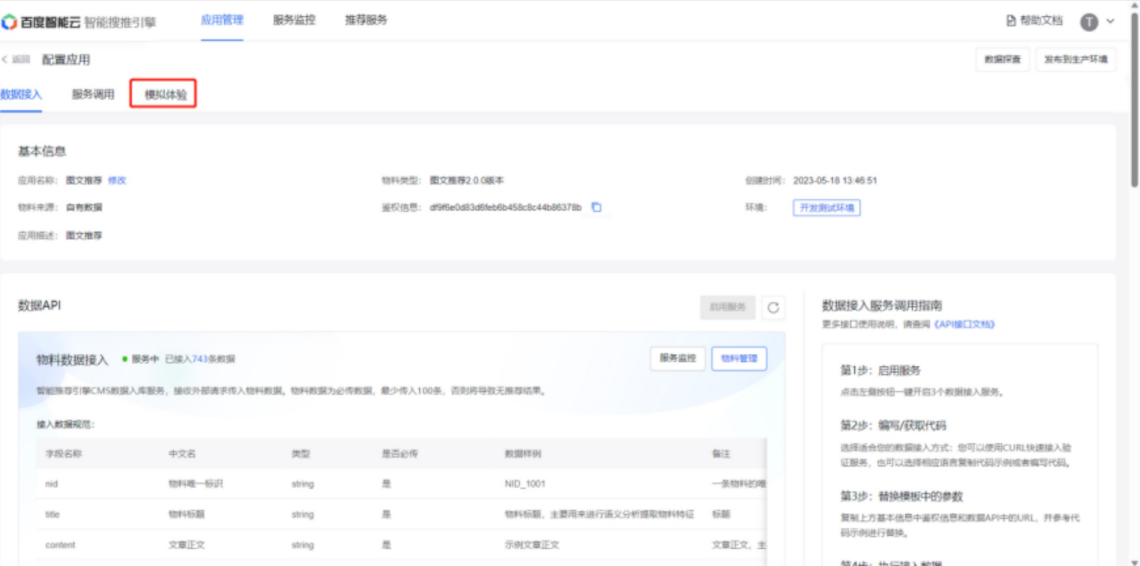
1. 租户完成应用创建。
2. 以下三种情况出现任何一个都无法使用：
 - (1) 数据服务未部署
 - (2) 物料数据未接入

(3) 数据服务未启动完成

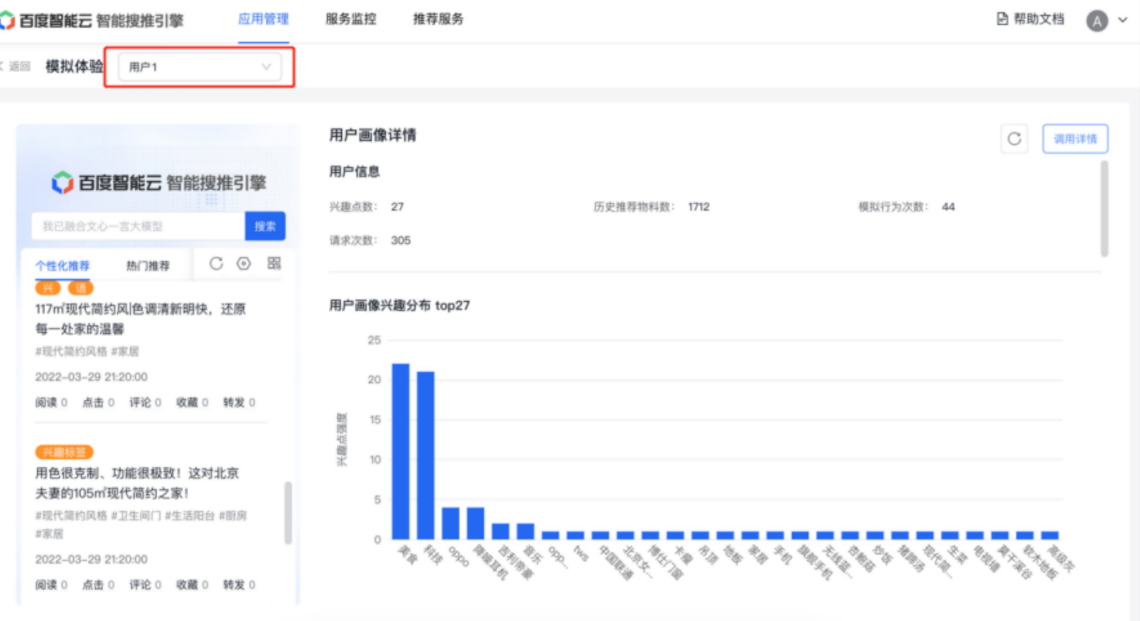
3个个性化推荐服务、热门推荐服务至少开启1个。

操作步骤

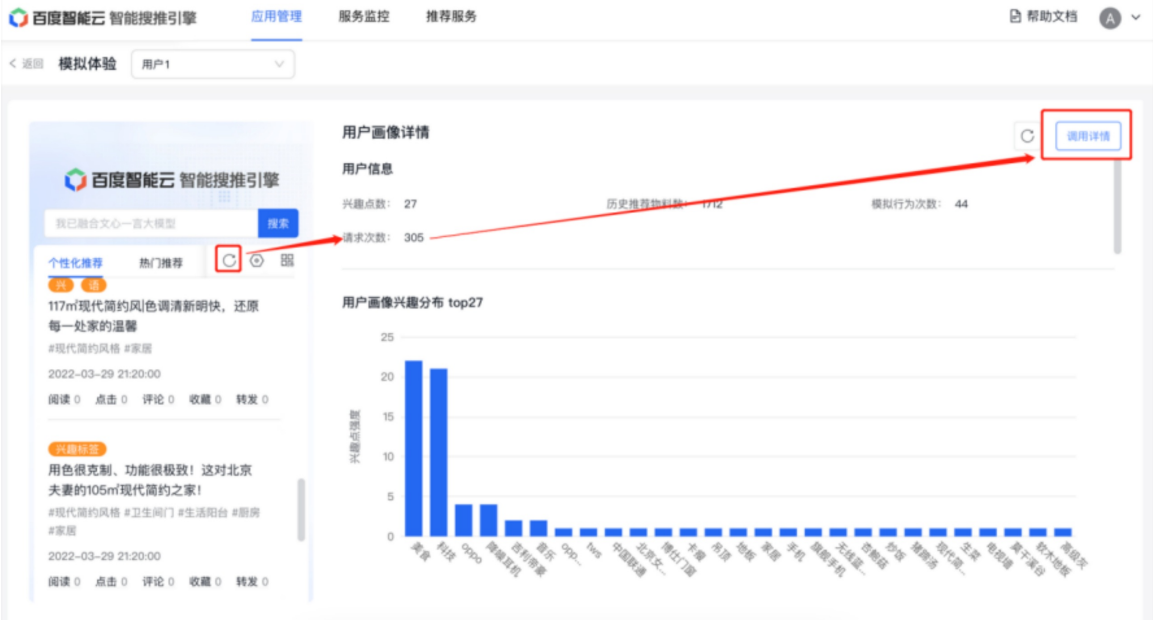
1登录进入智能推荐引擎，【应用列表】选择相应的应用--点击【配置】按钮，切换到服务调用页面，点击【模拟体验】页签。



2切换模拟用户身份，系统提供了10个模拟的用户用来区分不同的用户推荐效果，模拟体验的用户不需要事前上传。



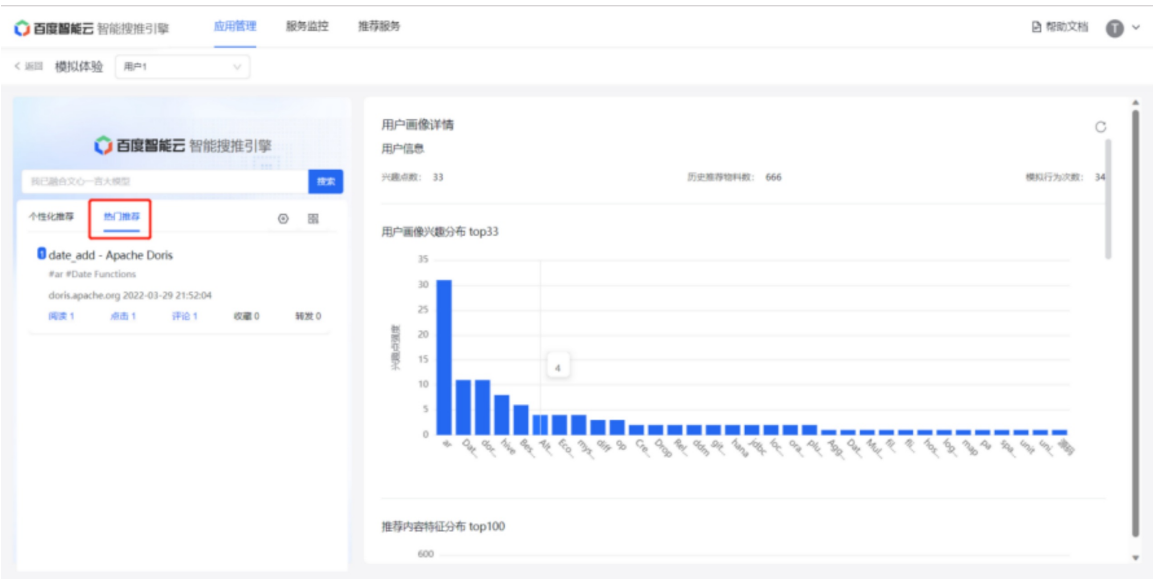
3体验个性化推荐：点击【刷新】按钮和手机端下拉刷新触发debug模式，触发后会更新当前用户请求累计次数并生成请求记录明细，请求记录明细可点击【请求记录】按钮进入服务监控的调用详情页内查看。您还可以点击【阅读（曝光）】、【点击】、【评论】、【收藏】、【转发】按钮模拟用户的点击行为，每次点击都会生成一条行为数据用于给当前用户推荐，暂不支持取消行为。



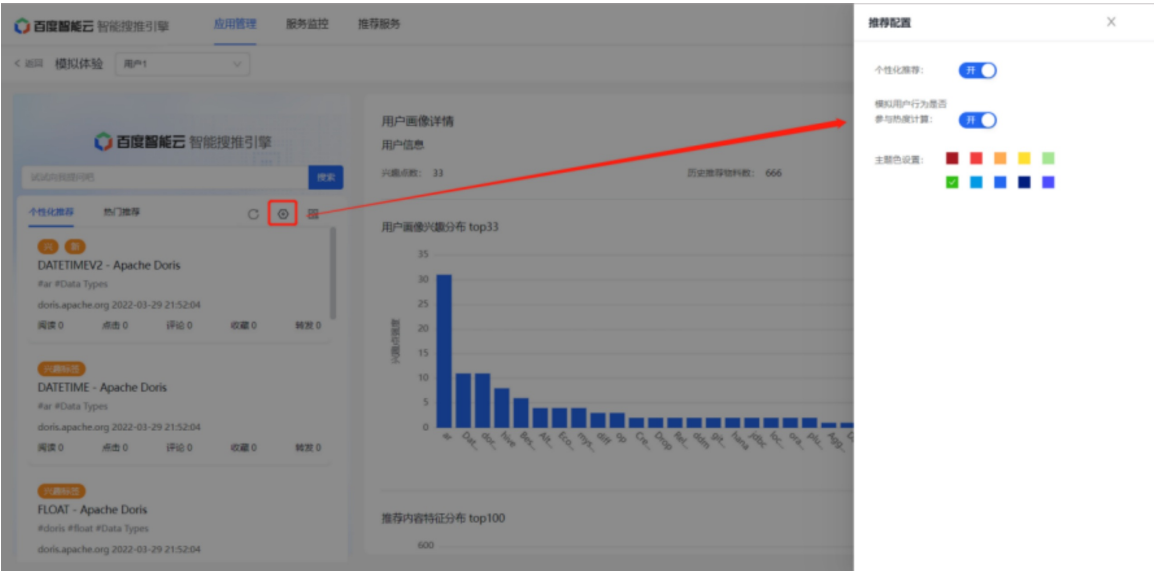
4体验相关推荐：点击您感兴趣的文章标题，弹出内容详情页，滑到底部“相关推荐”卡片下，便可浏览相关推荐结果。



5体验热门推荐：切换到热门推荐卡片下，滑动推荐列表可查看昨日文章的点击数排行榜。



6推荐设置：支持设置关闭个性化推荐效果和设置app端主主题。



说明：

问题：热门推荐和相关已经开启了，显示当前服务未生效。

可能的原因：

- (1) 未在服务调用页面开启本服务；
- (2) 服务已开启，但由于系统存在T-1的批量计算机制，需要您上传数据后1天后再来体验。

场景管理

场景管理

概述

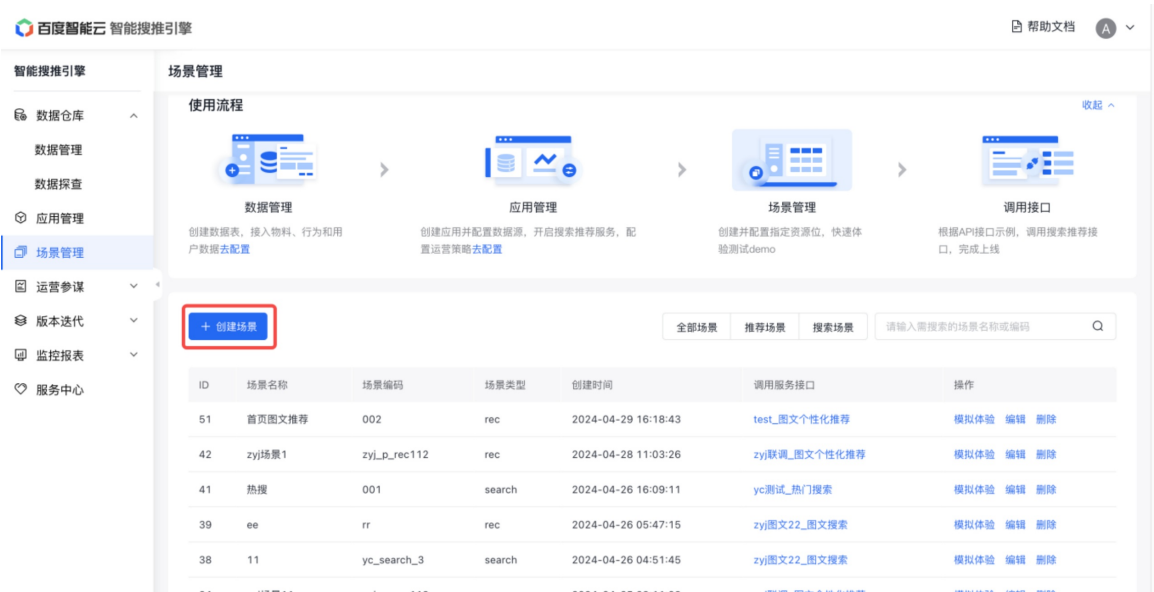
便于管理用户推荐栏位，更细粒度的分析业务指标效果。

前提条件

- 1应用已经创建完成。
- 2已经接入物料数据。

操作步骤

- 1登录进入智能推荐引擎，【场景管理】点击【创建场景】按钮。

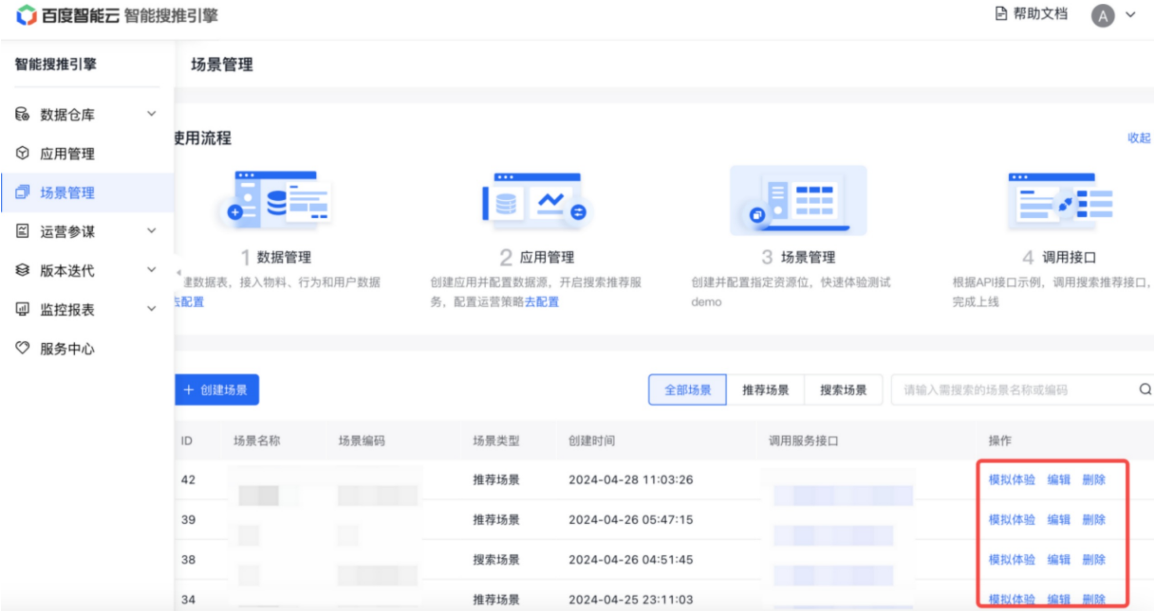


- 2弹出创建场景框，输入场景名称，场景编码，选择相应的场景类型（推荐/搜索场景），并选择调用的服务。

说明：
创建场景时，需要注意场景编码需要和行为数据表中event scene场景编码一致，且一一对应。



3可以使用场景名称或编码，找到寻找的场景，对指定场景进行编辑、删除和模拟体验等操作。



推荐策略管理

强插策略

概述

强制插入最近一个正反馈相关的几个相关推荐内容相似资源，推荐效果提升。

前提条件

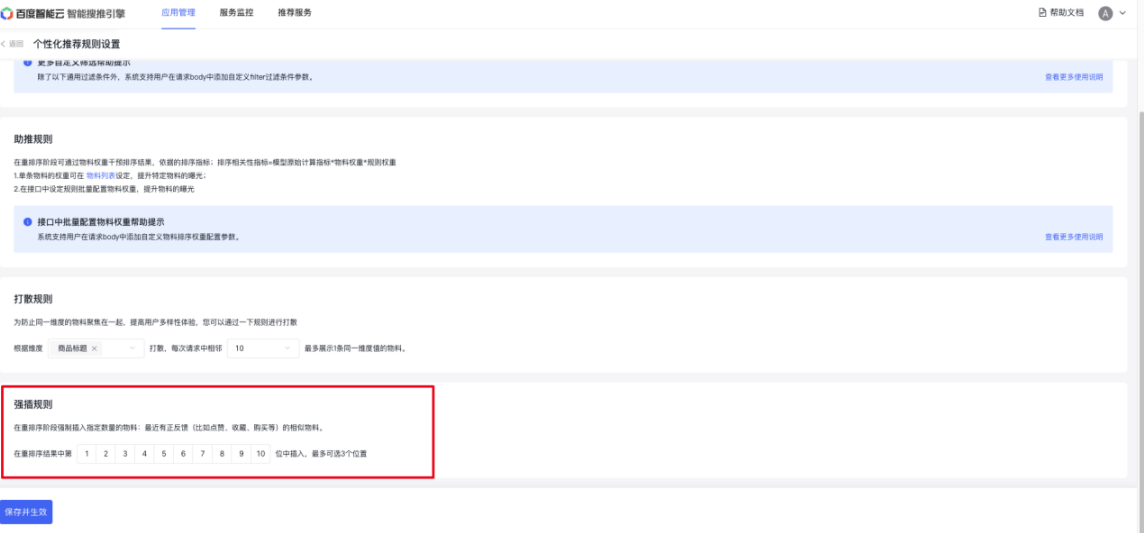
- 1应用已经创建完成。
- 2应用已经创建完成，并且已经接入物料数据。
- 3 服务已经启用。

操作步骤

1登录进入智能推荐引擎。

2配置重排序阶段强制插入指定数量的物料。在重排序结果中第x位，插入最近有正反馈 (比如点赞、收藏、购买等) 的相似物

料。



参数	说明
位置	第1~10均可选

打散策略

概述 原始推荐结果按照相似度打散为多个请求中。这种策略可以有效地避免用户对同一类内容多次浏览，从而提高用户体验和推荐算法的效率。

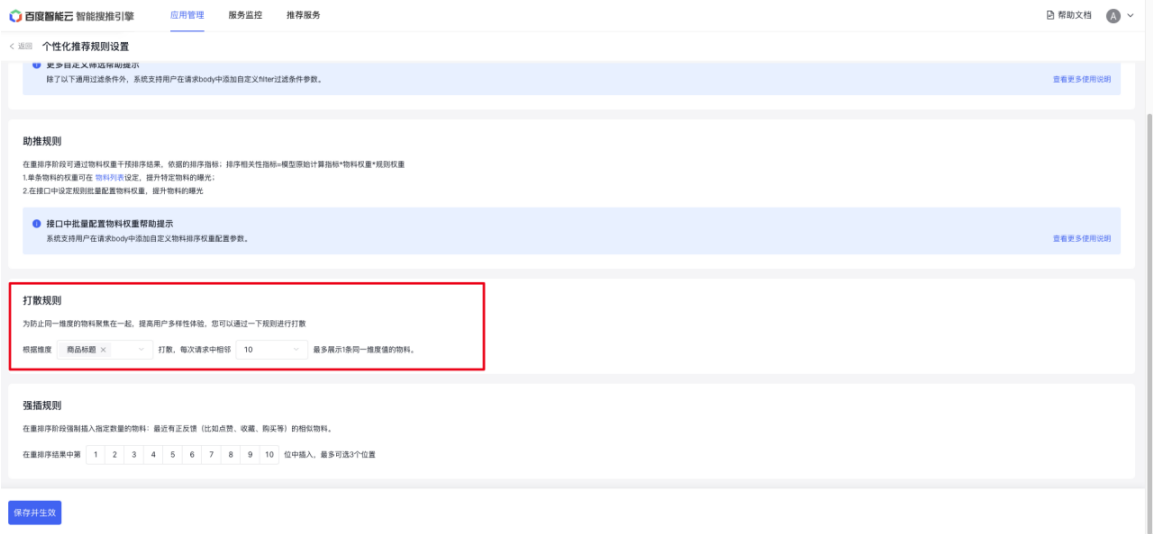
前提条件 1应用已经创建完成。

2应用已经创建完成，并且已经接入物料数据。

3 服务已经启用。

操作步骤 1登录进入智能推荐引擎。

2配置打散规则：为防止同一维度的物料聚集在一起，提高用户多样性体验，您可以通过以下规则进行打散。可设置【打散维度】，每次请求中最多展示 1条同一维度值的物料。



参数	说明
打散维度	图文：标题、作者、分类、标签、城市名称或编码、频道;商品：商品标题、店铺ID、商品品牌、商品自有标签、类别、商品来源类型、商品价格、城市名称或编码列表;媒资：媒资标题、国家地区、上映年份、演员、创作者、物料来源、媒资类型、内容类型、内容类型、运营商名称、是否需要付费、频道、影视语言、清晰度、城市名称或编码

排序调权策略

概述

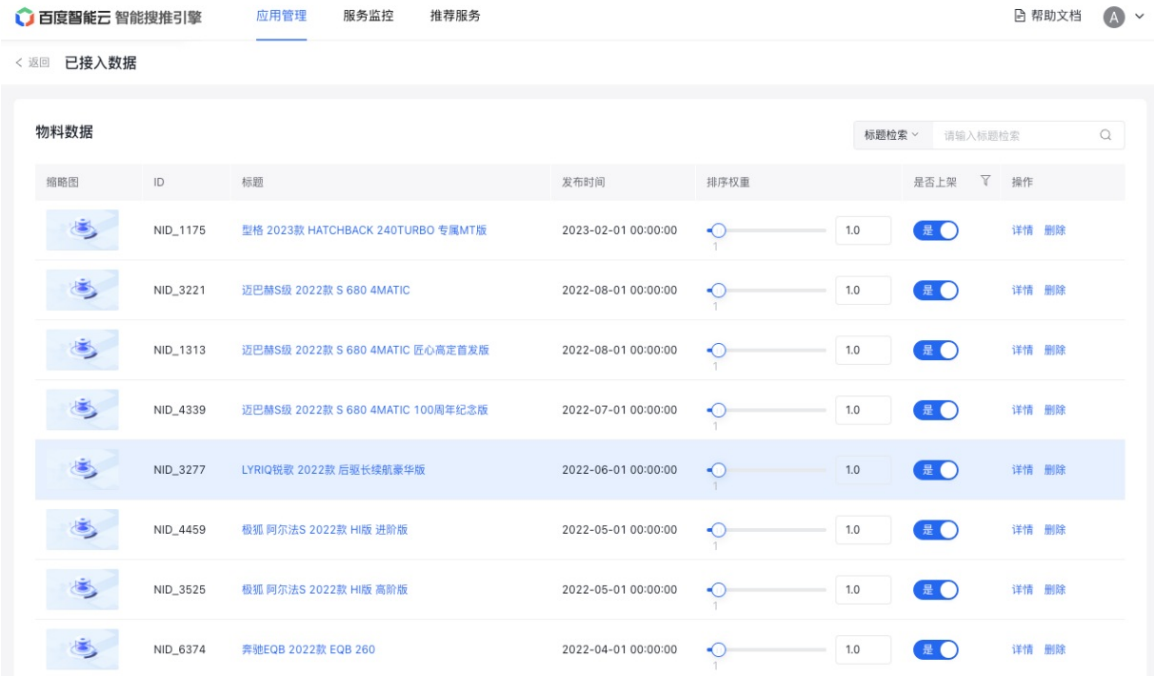
完全基于用户行为和模型的推荐往往可以有效满足用户的个性化需求，然而，对于大部分盈利性组织不得不考虑平台收益目标，例如提升付费内容的曝光，或者提升某类政治性稿件的排序等需求。加上在精细化运营的背景下，推荐系统需要支持在重排序阶段通过调整物料权重干预从而影响最终的排序结果。

前提条件

- 1应用已经创建完成。
- 2应用已经创建完成，并且已经接入物料数据。
- 3 服务已经启用。

操作步骤

- 1登录进入智能推荐引擎。
- 2在应用列表进入应用数据接入页面，点击【物料管理】按钮，进入已接入数据列表页，查询需要配置权重的物料，权重取值范围[0.1, 10]，可通过调节滑杆增大减小权重，步幅为0.1。



参数	参数说明
物料排序权重	1、单条物料的权重可在物料列表设定，提升特定物料的曝光。2、传入为空时默认值为1。

推荐物料选品

概述

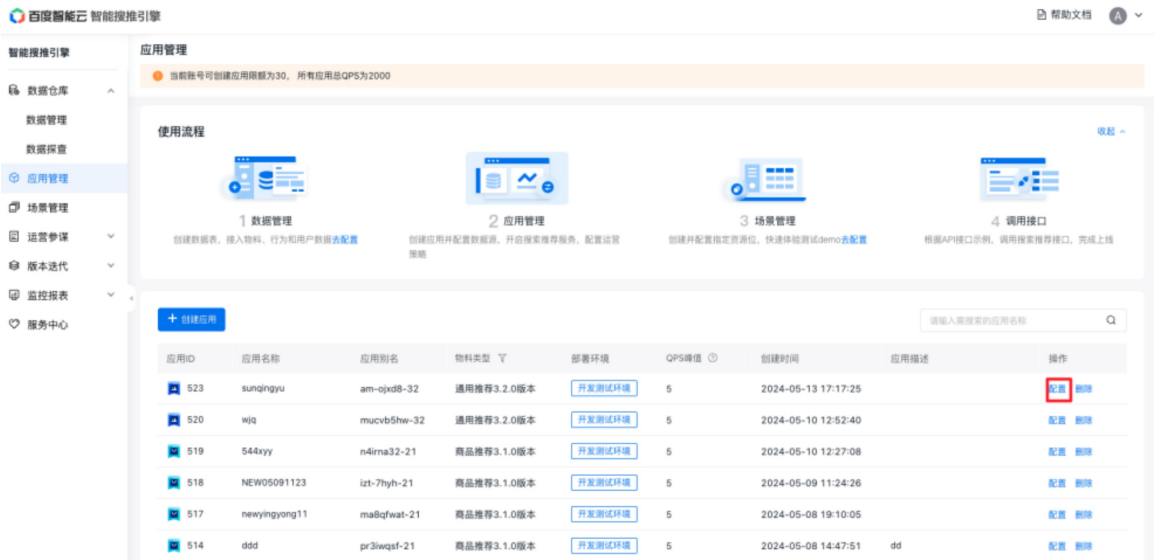
支持查看物料数据，并且可以查询物料状态。包含查询应用已上架的资源总量(上架总量、下架总量)以及查询上架、下架任务的id列表。

前提条件

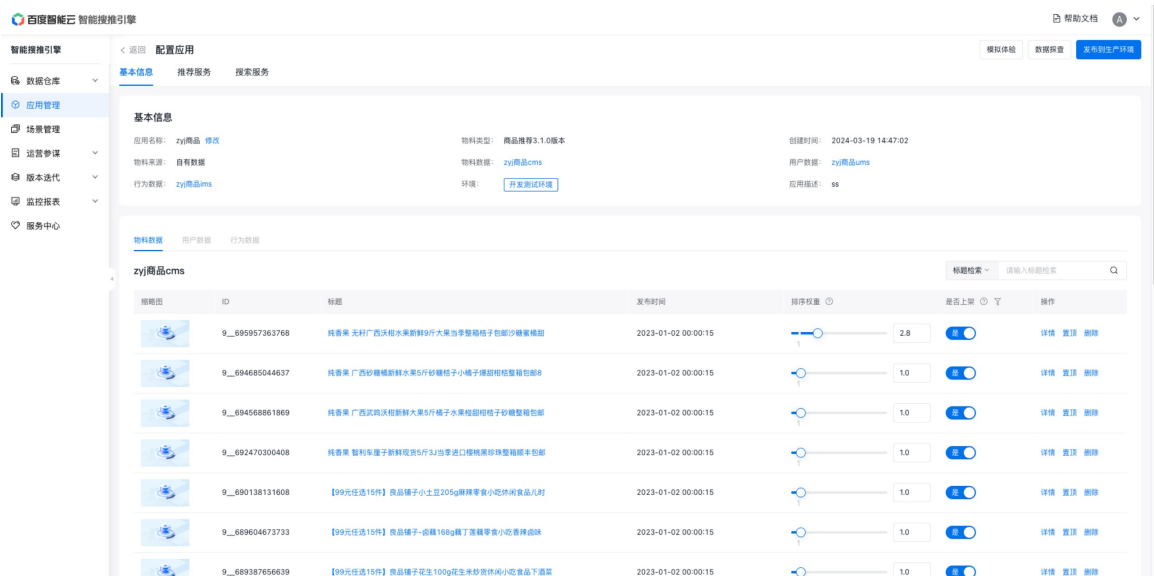
- 1应用已经创建完成。
- 2已经接入物料数据。

操作步骤

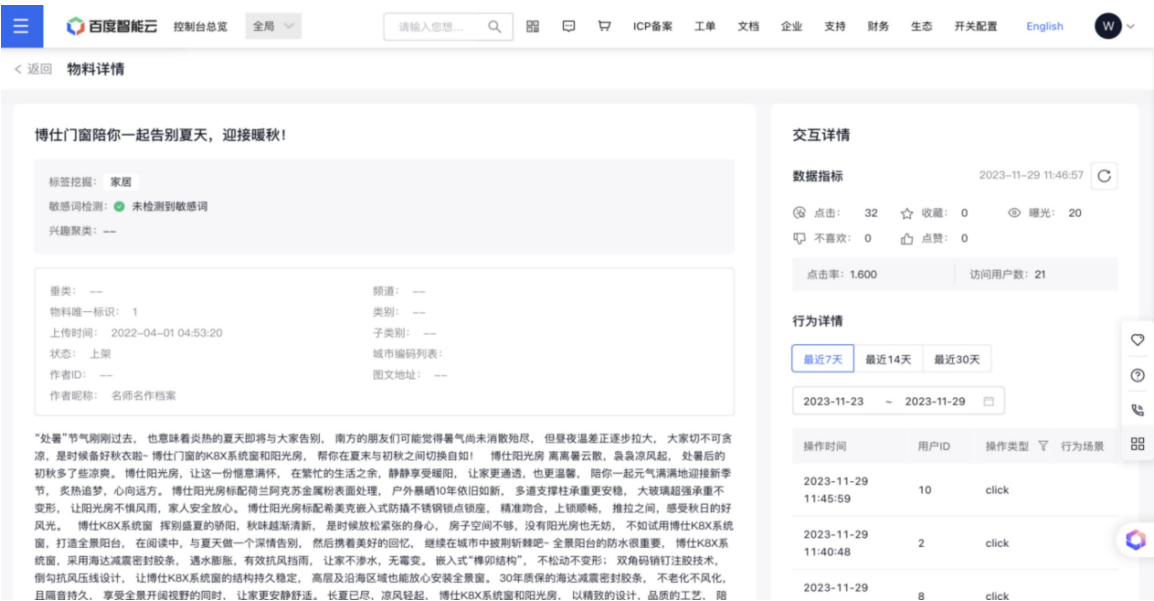
- 1登录进入智能推荐引擎，【应用管理】选择相应的应用--点击【配置】按钮。



2 进入配置应用页面，可以对某条物料进行查询、上下架、查看详情和删除操作。



3 支持单个物料粒度的点击数、收藏数、曝光数、点击率、点赞数和访问用户数等指标的统计和行为数据查询。



物料过滤策略

概述

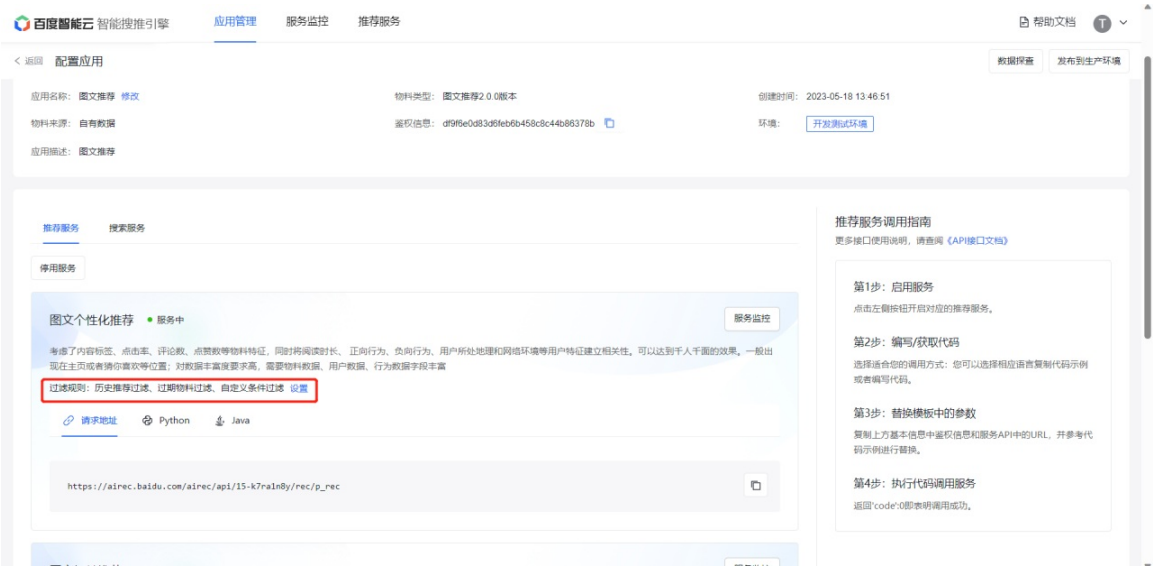
支持物料设置过滤规则满足客户精细化运营需求，提高运营干预力，提高用户操作自主性。

前提条件

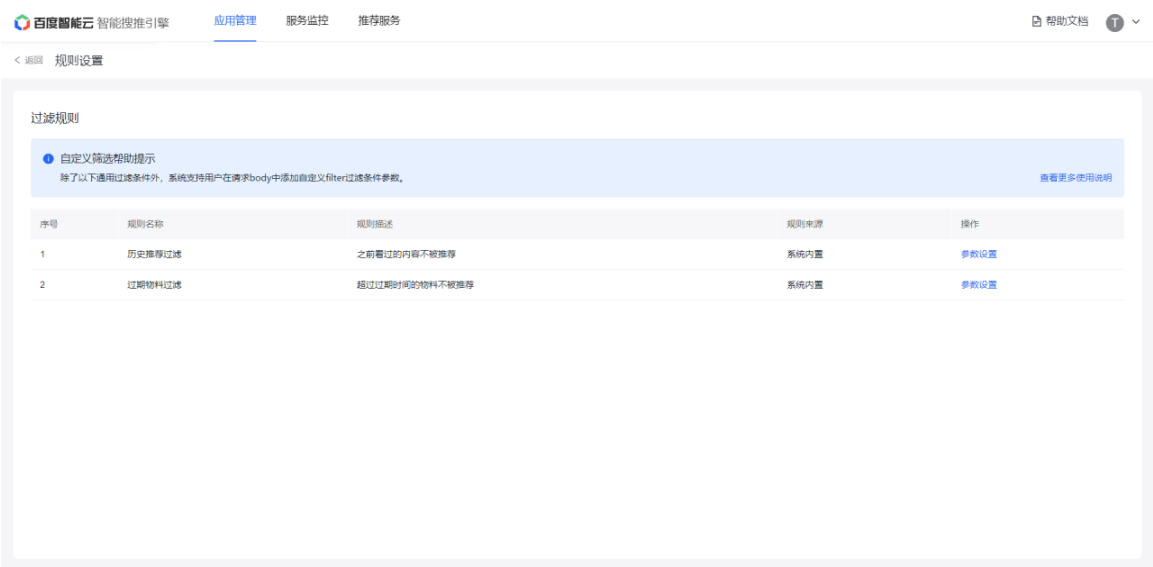
- 1应用已经创建完成。
- 2应用已经创建完成，并且已经接入物料数据。
- 3 服务已经启用。

操作步骤

- 1登录进入智能推荐引擎。
- 2在应用列表页面选择相应的应用，进入服务调用页面，点击过滤规则【设置】按钮，进入规则设置页。



- 3 系统预置历史推荐过滤、时效性过滤两种规则设定，点击【参数设置】按钮进行调整。



- 4 除了以上通过滤条件外，系统支持用户在请求body中添加自定义filter过滤条件参数。 推荐过滤功能支持对物品的单个属性字段的单值过滤，也支持对多个属性字段的组合过滤。 1、单值过滤：

支持的过滤条件有： 包含/不包含（contain/not_contain） ，仅支持多值字段 相等/不相等（equal/not_equal，仅支持单值字段 空值/非空值（null/not_null） 大于/小于/区间（greater_than/less than/including） ,仅支持数字或日期型字段

单值过滤JSON格式的过滤规则的具体语法：

```
{
  "cond": "contain" | "not_contain" | "equal" | "not_equal", // 要进行的比较操作
  "field": "<field_name>", // 要筛选的字段名, 如"tags"、"channel"等
  "value": "<compare_value>" // 要筛选的目标值, 如"美食"。
}
```

2、组合过滤：对单值过滤或其他组合过滤的结果进行“且/或”的逻辑组合。过滤规则通过一个JSON字符串来表示。组合过滤JSON格式的过滤规则的具体语法：

```
{
  "join": "and" | "or",
  "filters": [
    { 组合过滤规则 | 单值过滤规则 },
    { 组合过滤规则 | 单值过滤规则 },
    ...
  ]
}
```

基于地理位置召回

概述

本地生活服务的重要工具，配合客户完成本地服务触达，例如本地探店类文章推荐。

前提条件

- 1应用已经创建完成。
- 2应用已经创建完成，并且已经接入物料数据。
- 3 服务已经启用。

操作步骤

- 1登录进入智能推荐引擎。
- 2数据接入时物料传入位置数据。
- 3用户请求中增加位置参数。

百度智能云 智能推荐引擎

应用管理 服务监控 推荐服务

帮助文档 A

返回 配置应用

status	状态	int	是	1
author_id	作者ID	string	否	author_10101
author	作者昵称	string	否	张三
channels	频道	strings	否	["体育", "科技"]
tags	自有标签	strings	否	["iphone", "集度汽车"]
category	类别	string	否	娱乐
sub_category	子类别	string	否	电影
cities	城市名称或编码列表	strings	否	["beijing", "shanghai"]
url	图文地址	string	否	
summary	物料摘要	string	否	物料摘要
search_hint	搜索提示词	strings	否	["西红柿炒鸡蛋怎么做"]

收起非必填字段

数据探查 发布到生产环境

您可以参考如下数据样例，测试构建推荐应用。

A_CMSdemo.xlsx

A_IMSdemo.xlsx

A_UMSdemo.xlsx

点击下载

说明：1城市精确到区县级，建议按照位置名称文字匹配或者城市code，只要保持物料数据取值和用户请求中的位置数据取值一致即可。2如果有需要，建议组合使用筛选规则过滤非本地物料。

负反馈与黑名单

概述

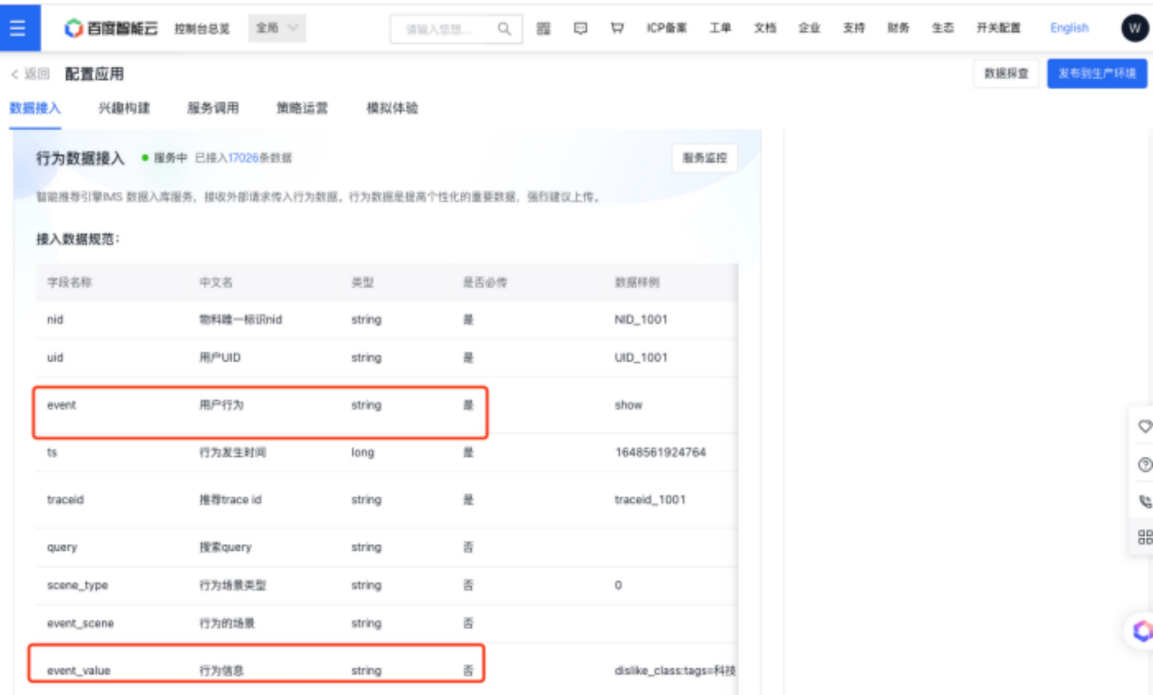
用户可以在推荐页面点击x按钮来添加黑名单。一旦物料被添加到黑名单中，系统将不会向用户推荐这些物料，有助于提高用户推荐的准确性和用户体验。

前提条件

- 1应用已经创建完成，并且已经接入物料、用户和行为数据。
- 2具有服务调用token权限。

操作步骤

1登录进入智能推荐引擎,按照行为数据规范，接入负反馈行为。行为数据增加新类型：dislike。用户可标记不喜欢或者选择：过滤标签、屏蔽作者、屏蔽类别。



2进入用户画像页面，点击黑名单【详情】按钮。

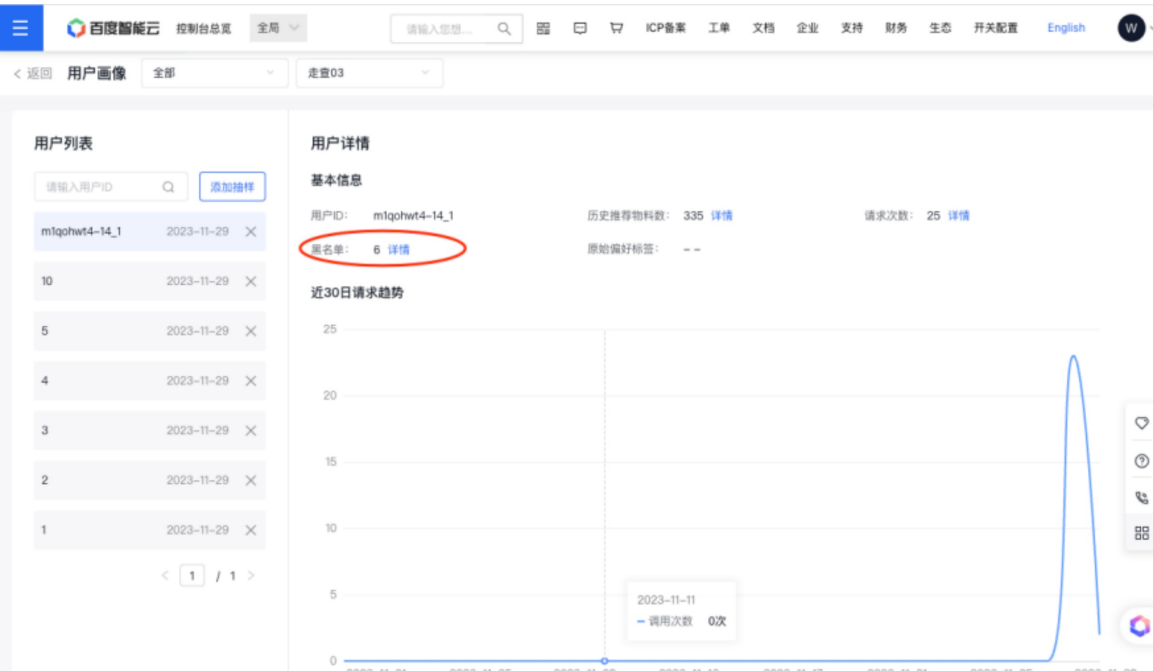
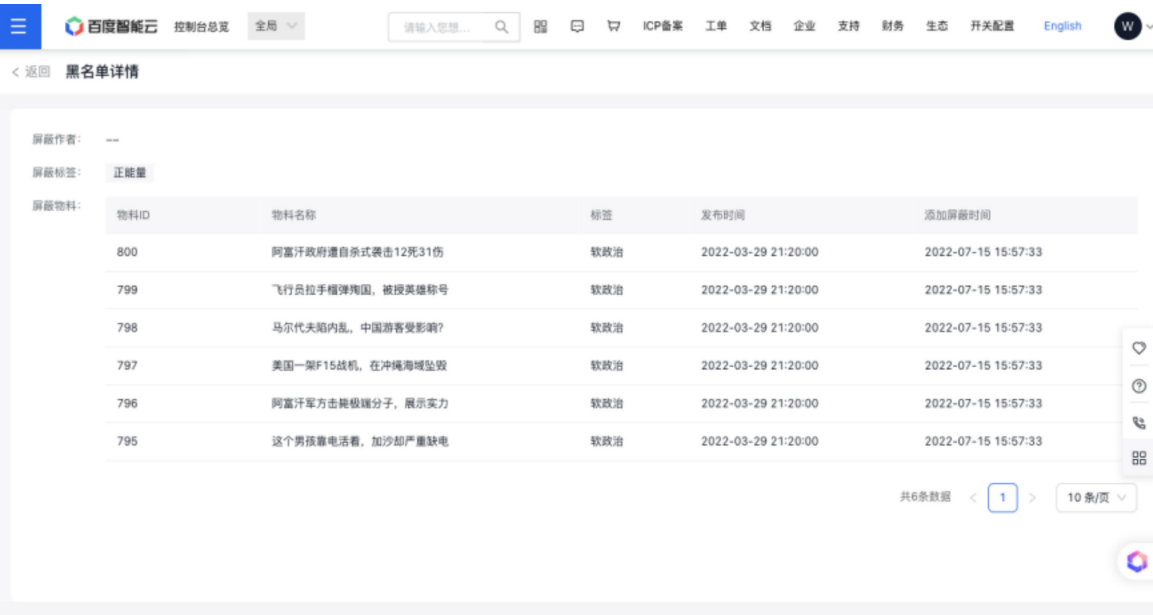


image.png

3进入黑名单详情页，查看屏蔽作者、标签和物料列表。



流量调控策略

概述

通过机器学习和数据分析技术，对新物料进行初始流量保量的功能。基于用户反馈行为的后验指标，减少无效流量推荐，同时放大优质内容，以提高点击和转化率。目前支持图文和媒资物料。

前提条件

- 1 应用已经创建完成。
- 2 已经接入物料数据。
- 3 服务已经启用。

操作步骤

1登录进入智能推荐引擎，【应用列表】选择相应的应用--点击【配置】按钮，进入【配置应用】--【推荐服务】--【个性化推荐】--【过滤规则】页面，进行流量调控策略配置。



2配置流量调控策略，设置【冷启动基础流量曝光】来保证新物料得到适当曝光，选项范围有无限、100、1k、1w、10w、

100w。

3二次推荐阶段，设置【内容优选指标基线】，优选指标包含CTR点击率、CVR购买转化率。



置顶策略

概述

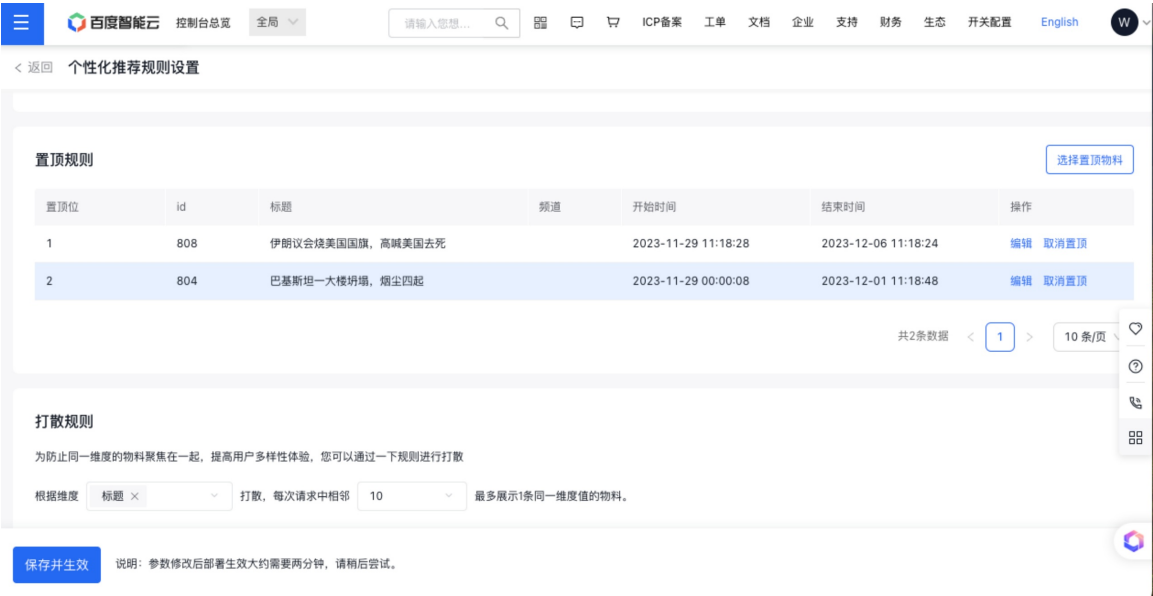
支持置顶一个或多个物料，可对物料列表中的物料进行编辑和取消置顶的操作。能够在内容分发过程中将选定内容放置在显眼位置，以吸引更多用户关注。

前提条件

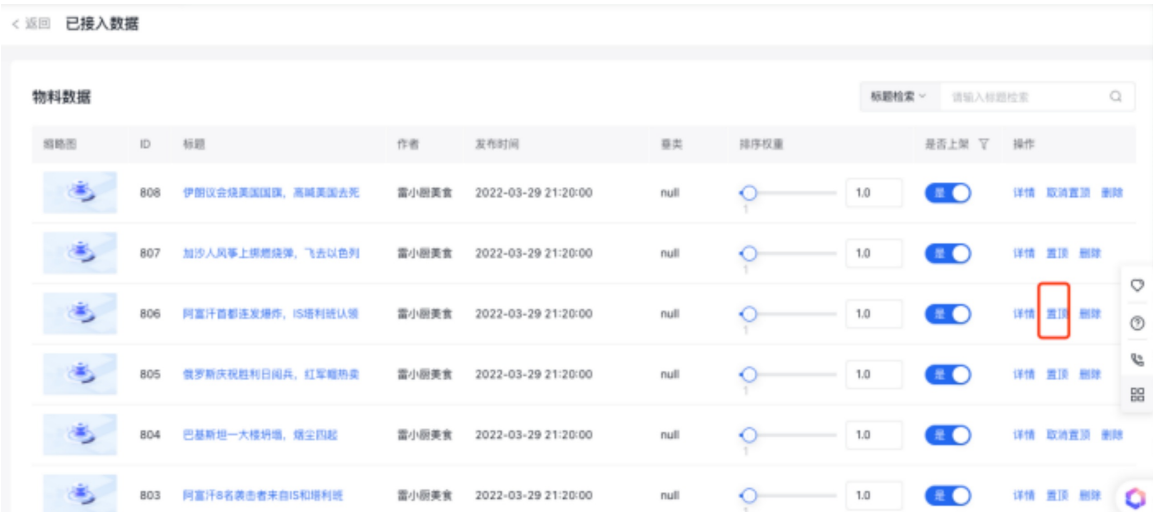
- 1 应用已经创建完成。
- 2 已经接入物料数据。
- 3 服务已经启用。

操作步骤

- 1登录进入智能推荐引擎。【应用列表】选择相应的应用-【策略运营】页面，点击【策略配置】。



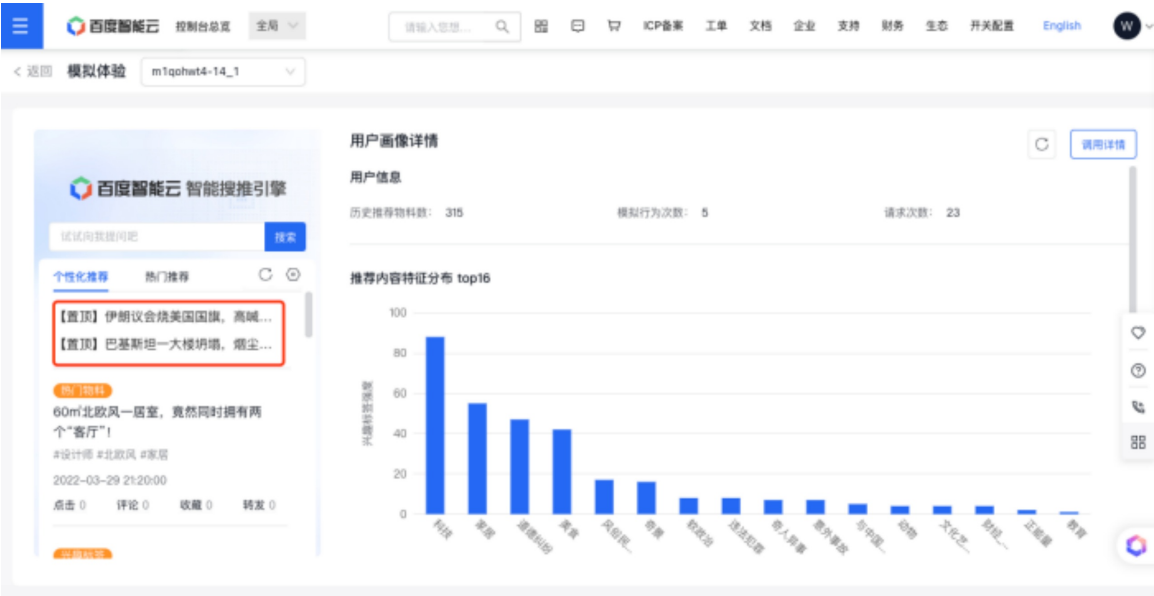
2 点击【选择置顶物料】进入物料列表页，筛选物料。



3 配置置顶参数，指定位置和时间，点击确定完成置顶操作。



4 进入模拟体验页面，查看是否配置成功。



搜索策略管理

搜索query词召回

概述

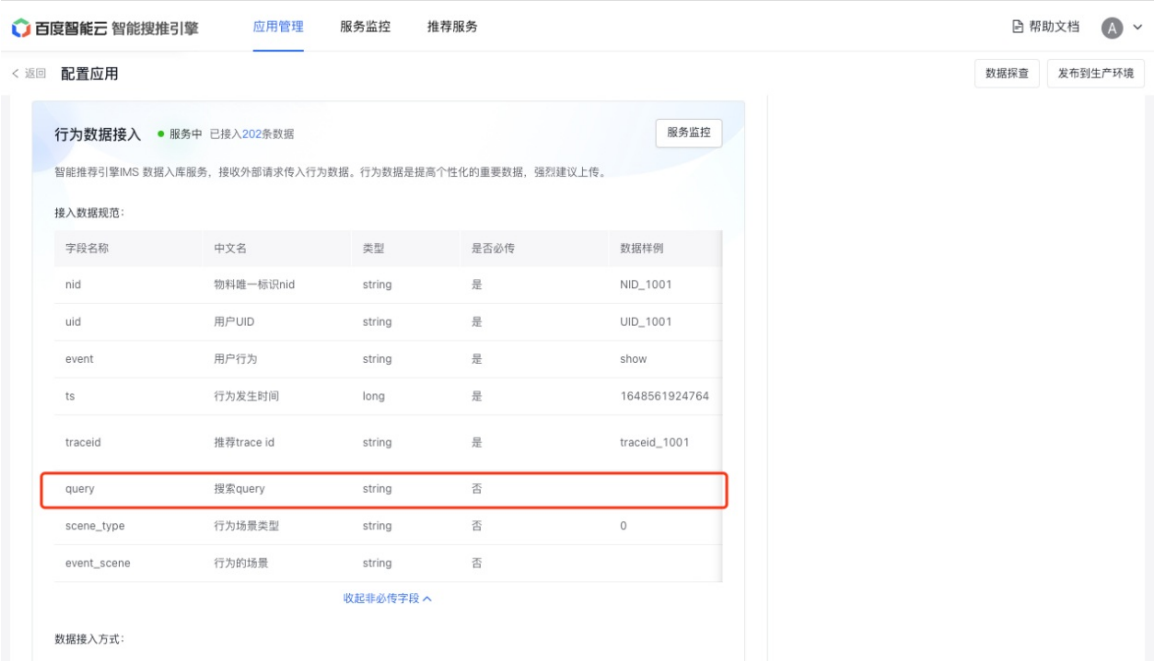
搜推一体化在业务层面最直接体现，搜索行为可即时反馈到推荐效果上。搜索和推荐行为相互增益，有助于更精准理解用户意图。

前提条件

- 1应用已经创建完成。
- 2应用已经创建完成，并且已经接入物料数据。
- 3 服务已经启用。

操作步骤

- 1登录进入智能推荐引擎。
- 2行为数据传入行为场景字段区分，行为发生位置，建议格式为：app-页面-区域，例如baidu_shouye_feeds
- 3行为数据事件字段增加搜索行为，支持上传query。



说明： 1搜索query取行为数据中的搜索行为，取最近的3个query，不足3个取全部。 2两次搜索query召回的结果会做去重。

搜索精确度配置

概述

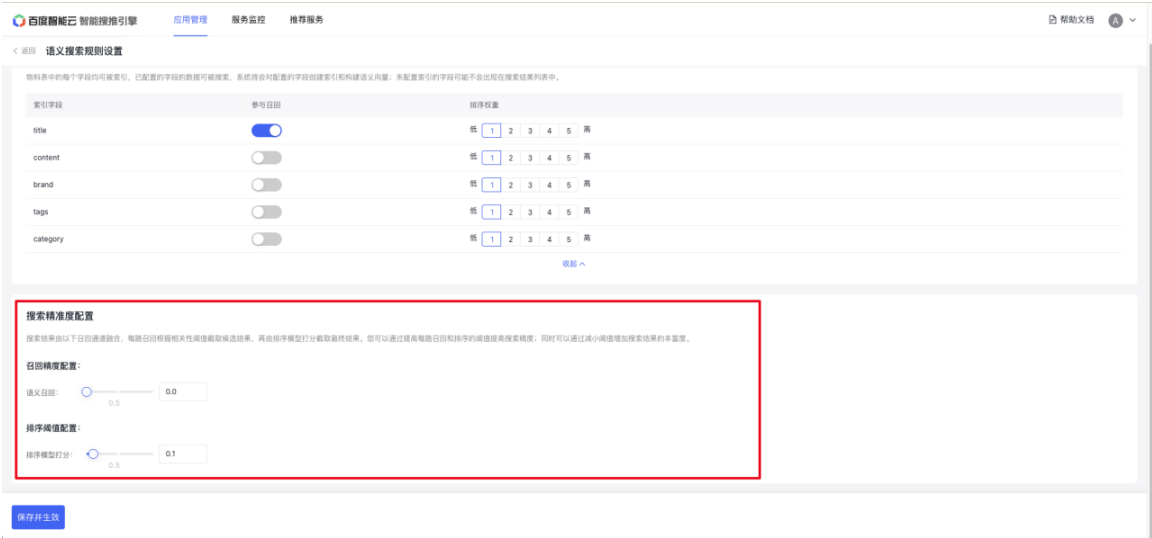
当前主搜索采取的是截断策略，且不支持翻页。支持翻页是用户心智中必有功能；调节阈值可以自由控制搜索精度。

前提条件

- 1应用已经创建完成。
- 2应用已经创建完成，并且已经接入物料数据。
- 3 服务已经启用。

操作步骤

- 1登录进入智能推荐引擎。
- 2搜索精确度配置：搜索结果由多种召回通道融合，每路召回根据相关性阈值截取候选结果，再由排序模型打分截取最终结果。



可以通过提高每路召回和排序的阈值提高搜索精度;同时可以通过减小阈值增加搜索结果的丰富度。语义召回和排序模型打分都可以使用滑块在0到1的范围内滑动，同时支持输入阈值。

索引配置

概述

已配置的字段的数据可被搜索，系统将会对配置的字段创建索引和构建语义向量；未配置索引的字段可能不会出现在搜索结果列表中。

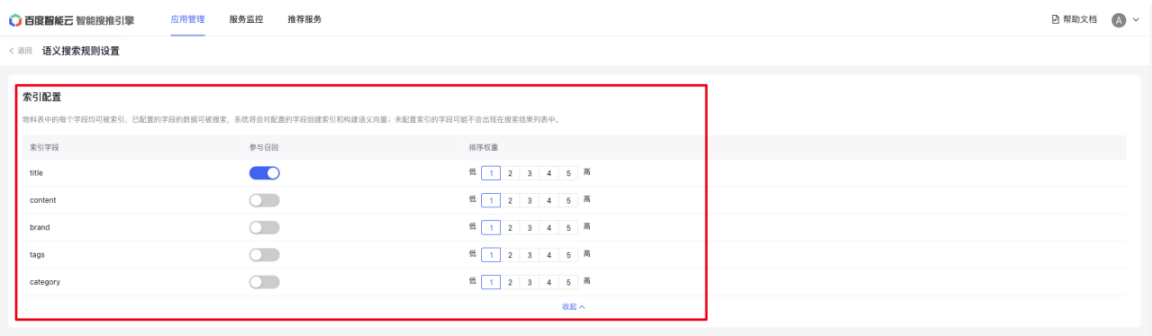
前提条件

- 1应用已经创建完成。
- 2应用已经创建完成，并且已经接入物料数据。
- 3 服务已经启用。

操作步骤

- 1登录进入智能推荐引擎。

2索引配置：物料表中的每个字段均可被索引，已配置的字段的数据可被搜索，系统将会对配置的字段创建索引和构建语义向量；未配置索引的字段可能不会出现在搜索结果列表中。点击【添加】按钮，可添加一条新的索引选择框，在索引选择框可选择索引字段。点击【删除】按钮，会删除选择的索引（默认索引不允许删除）。



词典管理

概述

将用户输入的query进行意图理解，拆分为专有名词、敏感词等，通过对专有词、同义词、纠错词以及敏感词等行业知识库管理，增强对用户搜索意图的识别能力。快速优化专有名词漏召回、同义词意图识别召回效果差、错误词意图无法识别、召回风险内容等问题。

前提条件

- 1应用已经创建完成。
- 2已经接入物料数据。
- 3服务已经启用。

操作步骤

1登录进入智能推荐引擎，进入【运营参谋】--【策略运营】选择对应【应用】--【搜索策略运营】，进行策略配置操作。

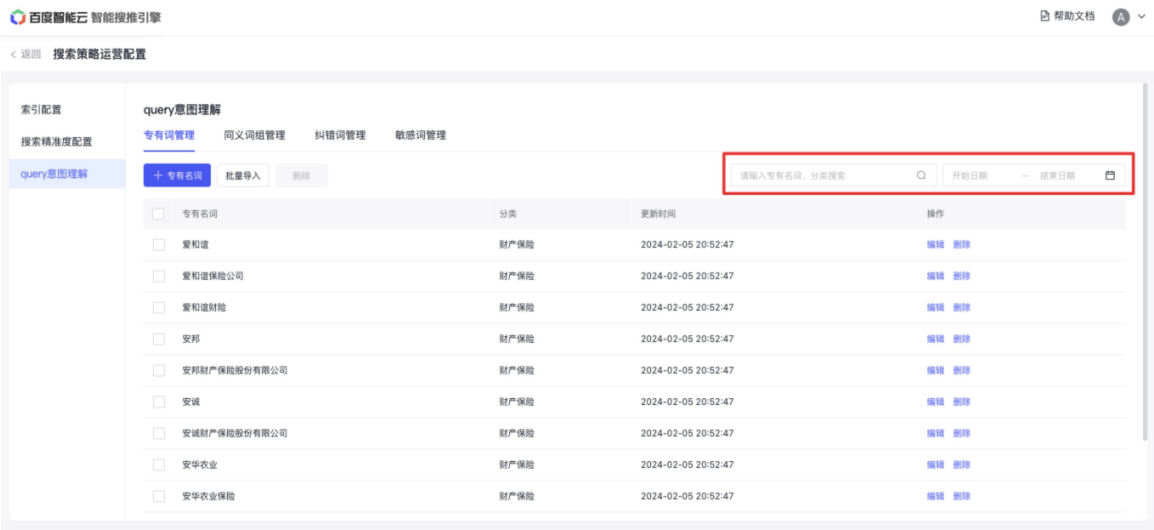


2词库管理：进入【词典管理】界面，共配置四类词库分别为专有词库、同义词库、纠错词库和敏感词库。

（1）词库导入：词库添加方式支持单独添加和批量导入，批量导入需下载对应的批量导入模版，根据模版格式生产完成词库后上传导入。词库配置后大约10分钟内生效。



(2) 词库筛选编辑：您可以使用搜索和日期筛选功能，定位到想要寻找的词典，对指定词典行编辑和删除操作。



热搜榜配置

概述

根据用户搜索行为配置计算主榜和分类榜单，帮助用户发现平台搜索需求和兴趣热点。热搜榜一方面能反映目前平台的流量趋势，关注度高的搜索词具有普适性并引发用户跟随，另一方面，热搜帮助平台运营者推广和运营活动。支持人工干预搜索词内容、排序、统计规则以及更新周期。

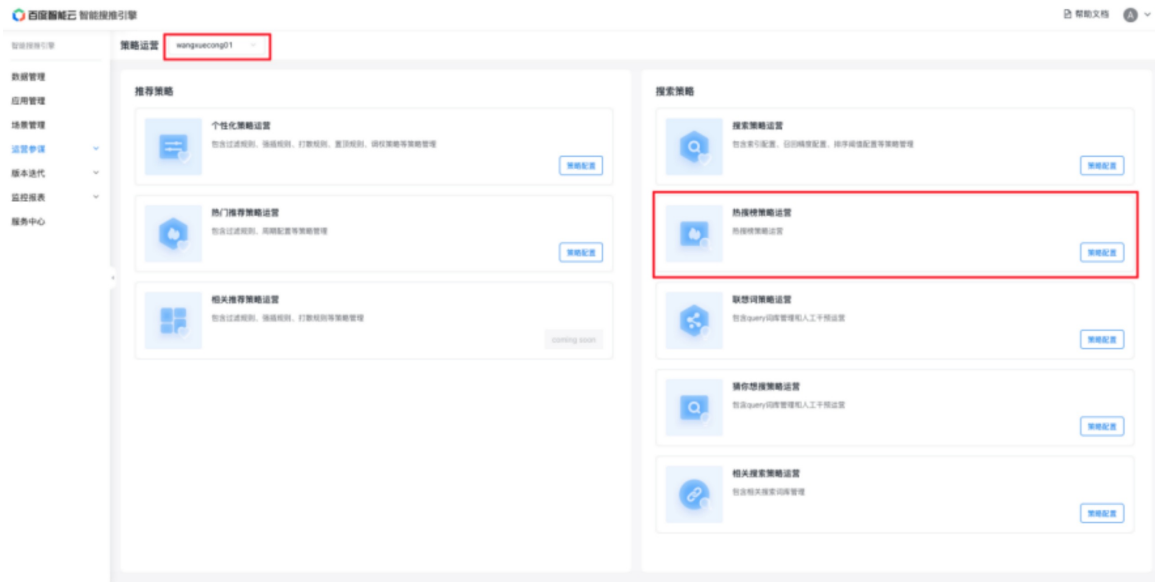
前提条件

- 1应用已经创建完成。
- 2已经接入物料数据。
- 3服务已经启用。

操作步骤

1登录进入智能推荐引擎，进行热搜榜单策略运营配置：

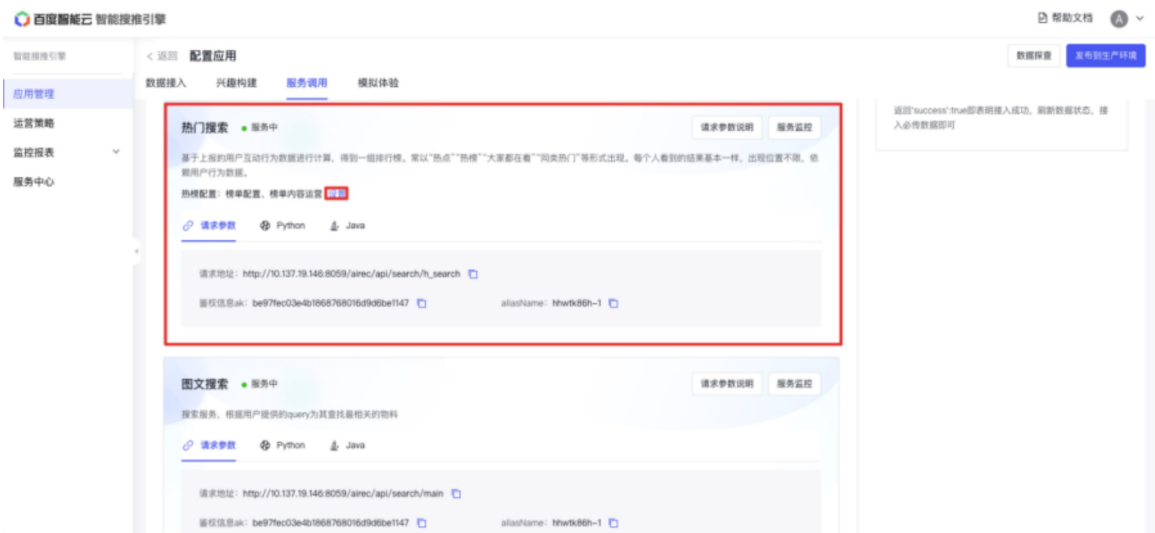
- (1) 方法一：进入【运营参谋】--【策略运营】选择对应【应用】--【热搜榜策略运营】，进行策略配置操作。



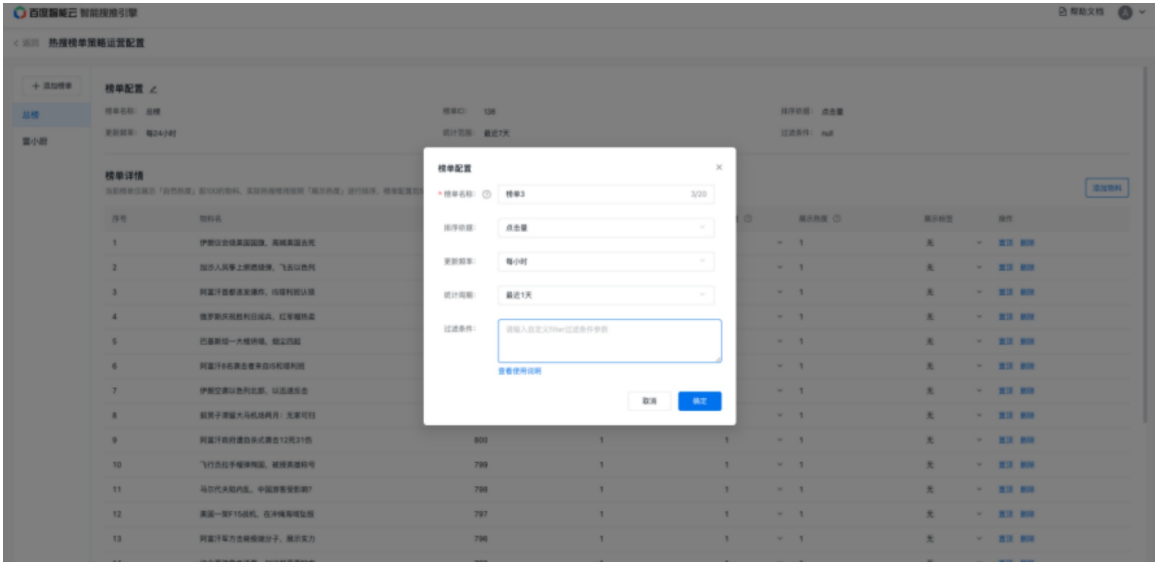
(2) 方法二：

第一步：进入【应用管理】，选择对应的应用，点击【配置】按钮进入配置页面。

第二步：在【配置应用】界面点击【搜索服务】--选择【热门搜索】--点击【设置】按钮，进行热榜配置。



2添加榜单：进入【榜单配置】界面，点击【添加榜单】按钮，填写榜单名称，配置排序依据、更新频率、统计周期、过滤条件等参数。



参数	参数说明
榜单名称	支持输入中英文、数字，建议4字以内，最多8字。
排序依据	query或物料的先后顺序的统计方式，包括：点击量、曝光量、关注量、收藏量等。
更新频率	重新统计用户行为数据的间隔，默认每小时，可选择每3、6、12、24小时。
统计周期	统计用户行为数据的范围，默认最近1天，可选择最近7、15、30天。
展示标签	展示标签默认无，可选择实时、热、推荐三款标签，增加热榜内容的吸引力。
过滤条件	输入自定义filter过滤条件参数。

3配置榜单：添加完榜单后可以直接修改榜单名称、排序依据、更新频率、统计范围参数。支持对单独对榜单内容进行操作。包括热度权重调整、标签选择、置顶操作、删除。

百度智能云 智能搜推引擎

帮助文档

返回

热搜榜单策略运营配置

+ 添加榜单

总榜

榜单配置

榜单名称: 总榜 榜单ID: 58 排序依据: 点击量 更新频率: 每24小时 统计范围: 最近7天

榜单详情

当前榜单仅展示「自然热度」前100的物料。实际热搜榜将按照「展示热度」进行排序

添加物料

序号	物料名	物料ID	自然热度	权重	展示热度	展示标签	操作
1	阿富汗8名袭击者来自IS和塔利班	803	1	1	1	无	取消置顶 删除
2	叙男子滞留大马机场两月：无家可归	801	1	1	1	无	取消置顶 删除
3	阿富汗首都连发爆炸，IS塔利班认领	806	1	10	10	无	置顶 删除
4	伊朗议会烧美国国旗，高喊美国去死	808	1	10	10	无	置顶 删除
5	80后大叔，高颜值+颜值风，太有质感了	2318	1	1	1	无	置顶 删除
6	一张门票45万！达沃斯论坛是干啥的	1172	1	1	1	无	置顶 删除
7	意大利北突遇冰雹，葡萄酒产地遭殃	606	1	1	1	无	置顶 删除

说明：
展示热度是用户端最终显示的热度值，具体计算数值=自然热度*权重，通过权重调整可以使得某个物料的热度放大。

4置顶操作，点击【置顶】按钮，可以将某个物料在设定时间内排序到指定位置。点击【取消置顶】可以将某个已经置顶的物料还原到原来的位置。

百度智能云 智能搜推引擎

帮助文档

返回

热搜榜单策略运营配置

+ 添加榜单

榜单1

榜单2

序号	物料名	物料ID	自然热度	权重	展示热度	展示标签	操作
1	小户型也可以拥有炫酷的全黑色厨房	2551	1	1	1	实时	取消置顶 删除
2	伊朗议会烧美国国旗，高喊美国去死				1	实时	置顶 删除
3	阿富汗首都连发爆炸，IS塔利班认领				1	热	置顶 删除
4	阿富汗8名袭击者来自IS和塔利班				1	无	置顶 删除
5	伊朗空袭以色列北部，以迅速反击				1	无	置顶 删除
6	加沙人风筝上绑燃弹，飞去以色列				1	无	置顶 删除
7	俄罗斯庆祝胜利日阅兵，红军唱热曲				1	无	置顶 删除
8	巴基斯坦一大楼坍塌，烟尘四起				1	无	置顶 删除
9	叙男子滞留大马机场两月：无家可归				1	无	置顶 删除
10	阿富汗政府遭自杀式袭击12死31伤	800	1	1	1	无	置顶 删除
11	飞行员徒手榴弹殉国，被授英雄称号	799	1	1	1	无	置顶 删除
12	马尔代夫内乱，中国游客受影响?	798	1	1	1	无	置顶 删除
13	美国一架F15战机，在冲绳海域坠毁	797	1	1	1	无	置顶 删除
14	国家卫健委发布春节假期提示：避免乏力	796	1	1	1	平	置顶 删除

配置置顶

物料ID: 799

物料标题: 飞行员徒手榴弹殉国，被授英雄称号

* 指定位置: 指定位置

* 置顶时间: 开始日期 结束日期

取消 确定

相关搜索

概述

在搜索结果页和详情页推荐与该query和内容相关的其他query。用户可以在一个搜索结果页面上获取更多与该主题相关的搜索

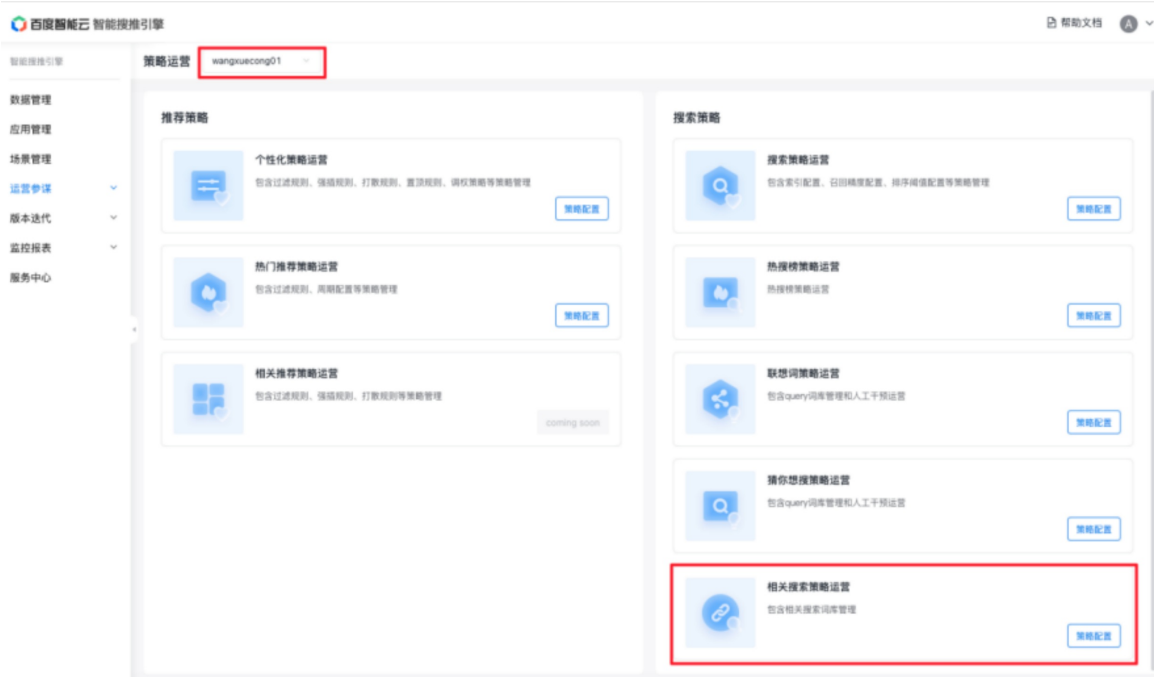
词，而不需要多次输入不同的关键词进行搜索。

前提条件

- 1应用已经创建完成。
- 2并且已经接入物料数据。
- 3服务已经启用。

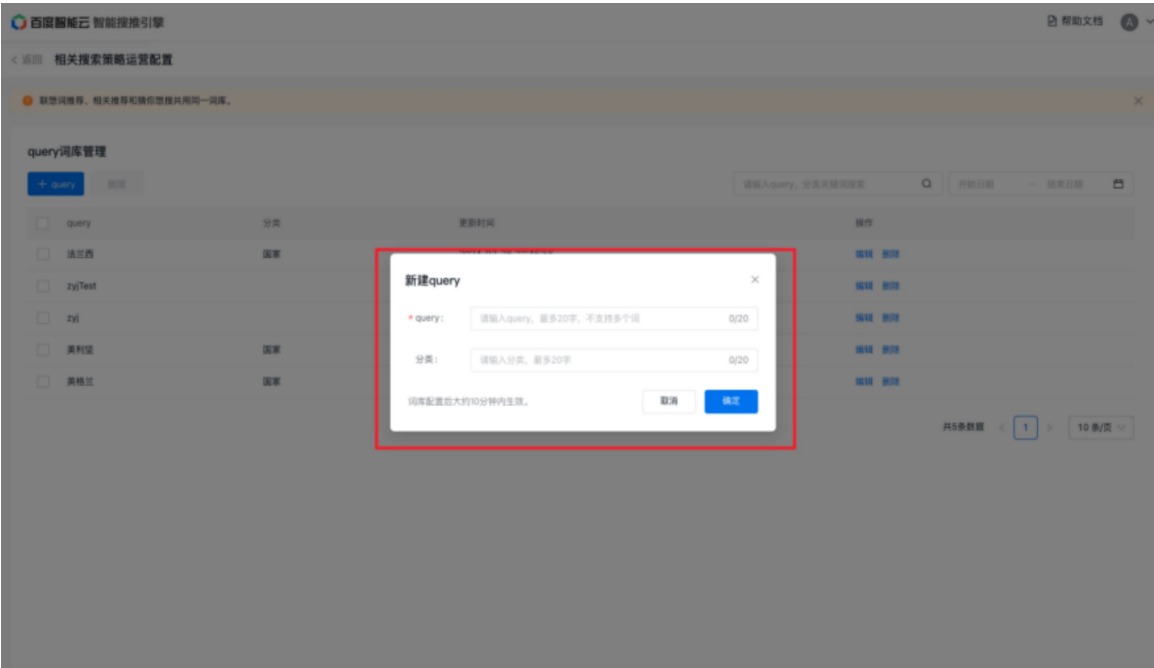
操作步骤

1登录进入智能推荐引擎，进入【运营参谋】--【策略运营】选择对应【应用】--【相关搜索策略运营】，进行策略配置操作。

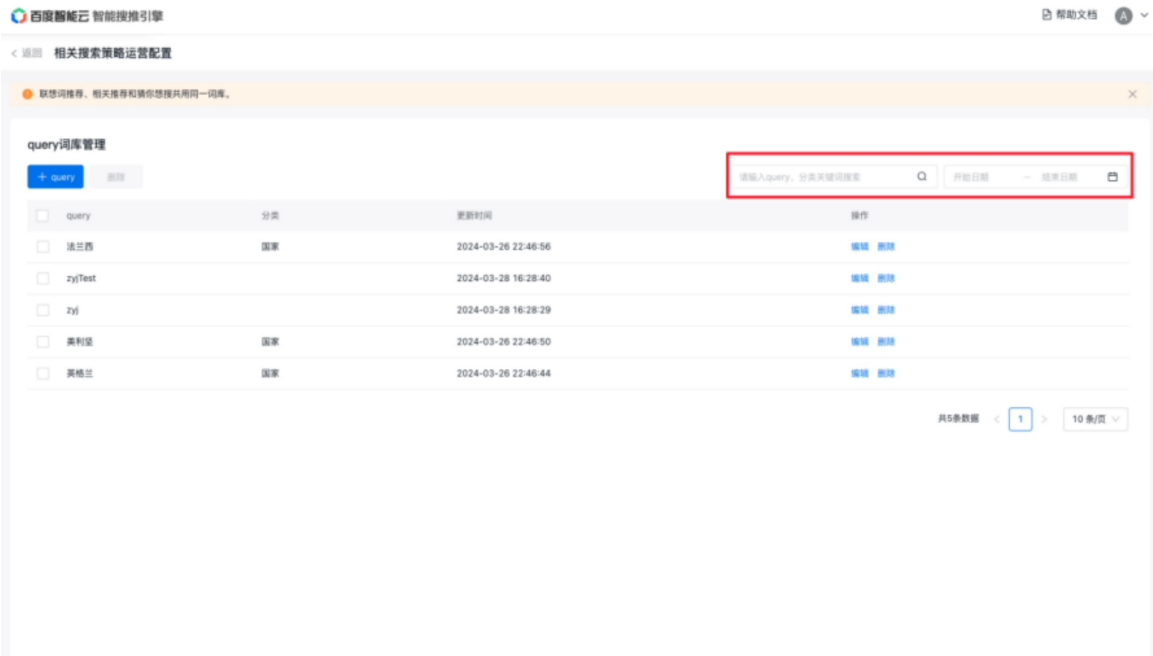


2query词库管理：

- (1) 新建query：进入【query词库】页面，点击【+query】新建query并输入分类，词库配置后大约10分钟内生效。



- (2) 词库筛选编辑：您可以使用搜索和日期筛选功能，定位到想要寻找的词典，对指定词典进行编辑和删除操作。



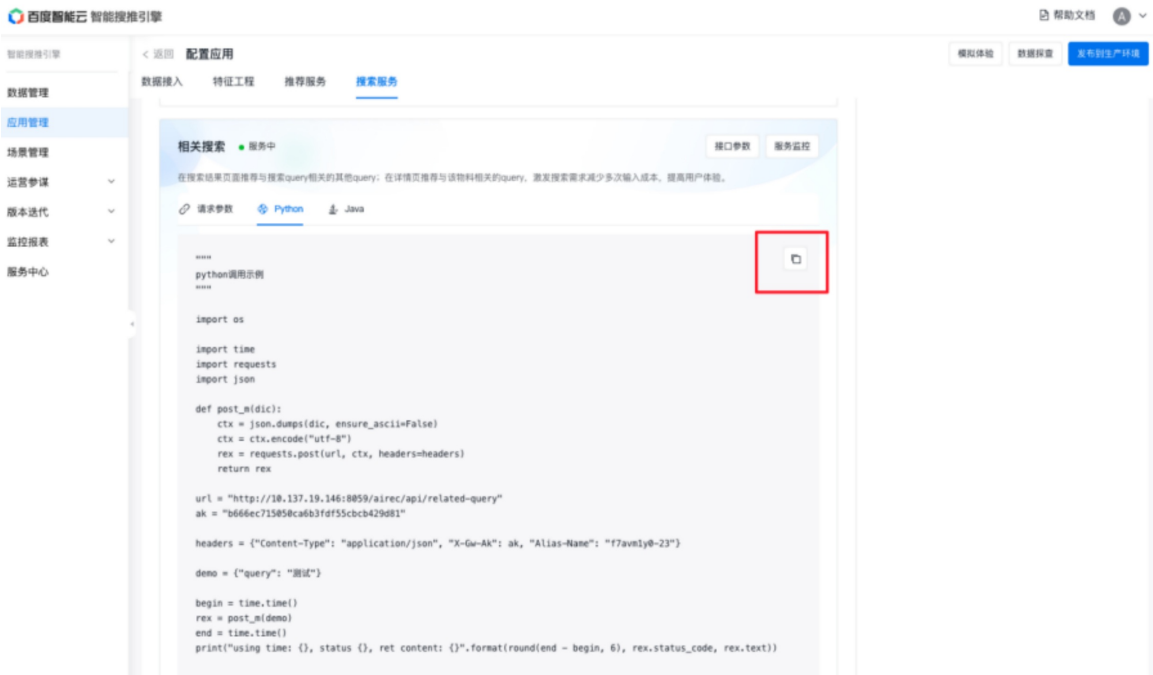
3参数配置步骤：

(1) 登录进入智能推荐引擎，进行相关搜索参数配置：

第一步：进入【应用管理】，选择对应的应用，点击【配置】按钮进入配置页面。

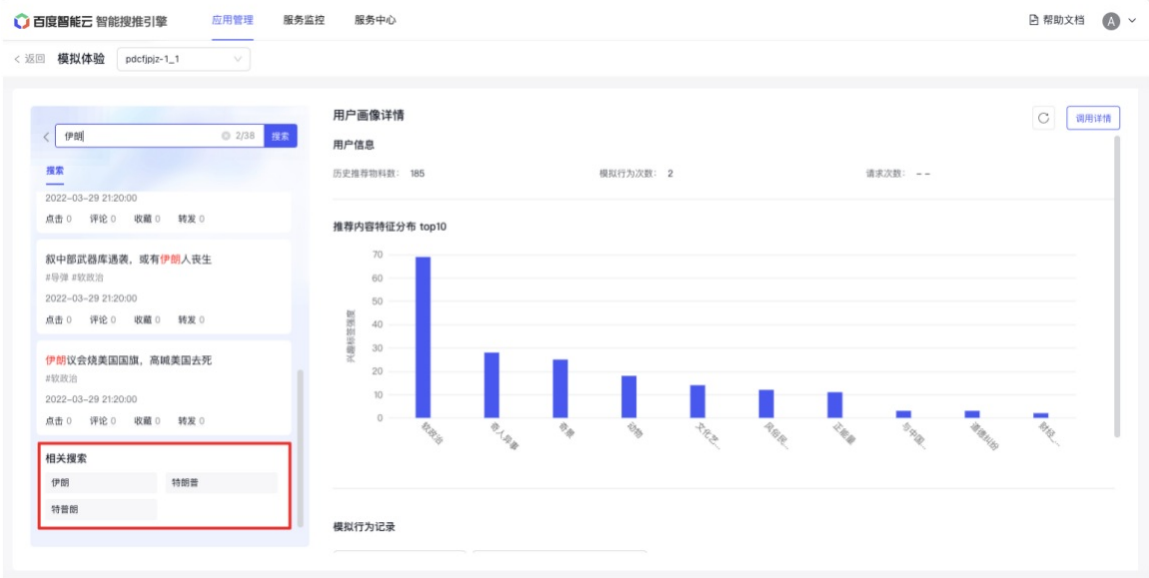
第二步：在【配置应用】界面点击【搜索服务】--选择【相关搜索】，进行相关搜索参数配置。

(2) 接口调用：复制代码脚本，根据API文档对服务进行调试。



4参数场景说明：

(1) 场景1:query-query：搜索结果页中推荐相关搜索词：



(2) 场景2:item-query：物料页面中推荐相关搜索词：



拼音召回策略

概述

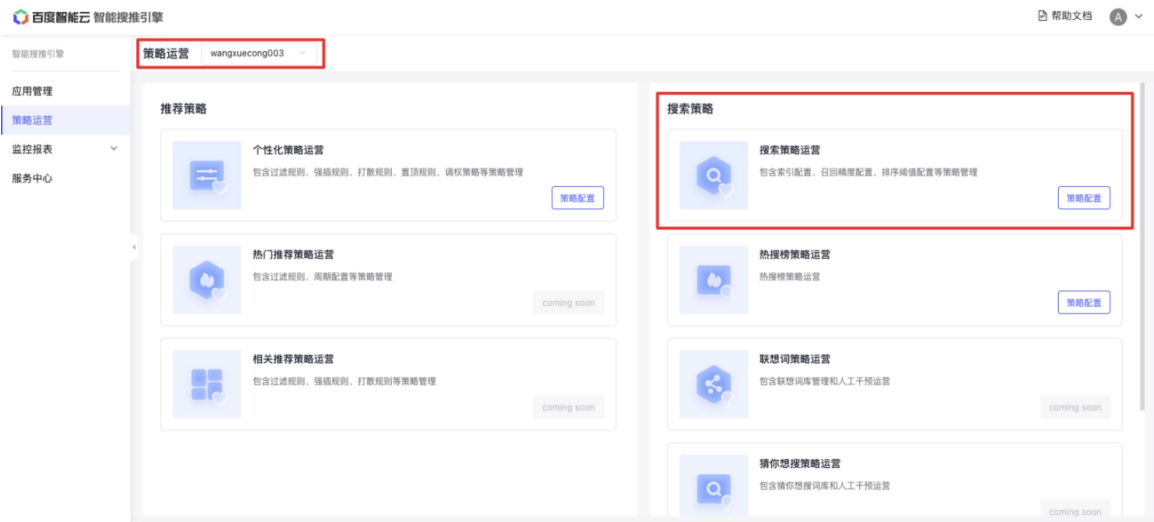
对用户输入的全拼、首字母缩写、拼音+汉字能自动进行搜索。

前提条件

- 1应用已经创建完成。
- 2已经接入物料数据。
- 3 服务已经启用。

操作步骤

1登录进入智能推荐引擎，进入【运营参谋】--【策略运营】选择对应【应用】--【搜索策略运营】，进行策略配置操作。



2拼音召回策略：进入【召回配置】界面，选择【拼音召回】，打开【开/关】按钮，进行【策略配置】。



3进入【策略配置】界面，可选择是否开【中文query转拼音】，并选择【支持query类型】和【覆盖转化字段】，点击【保存并生效】。



联想词策略运营

概述

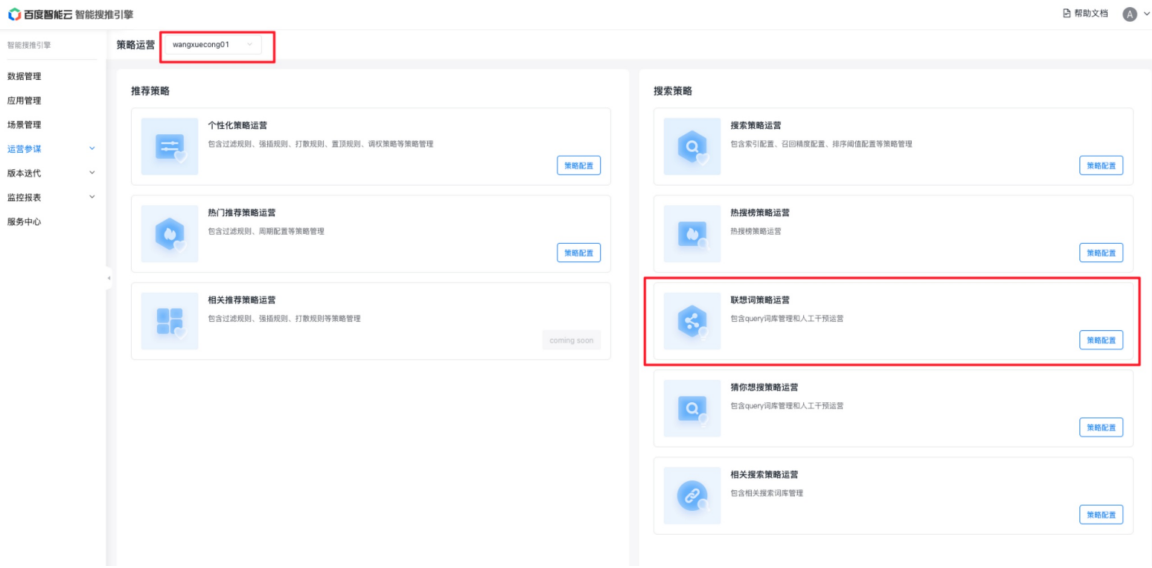
用户在输入框输入字符时，相关词会展示在列表中，帮助用户快速输入，将query补全。通过对query词库的维护，来优化联想词效果差的问题。

前提条件

- 1应用已经创建完成。
- 2已经接入物料数据。
- 3服务已经启用。

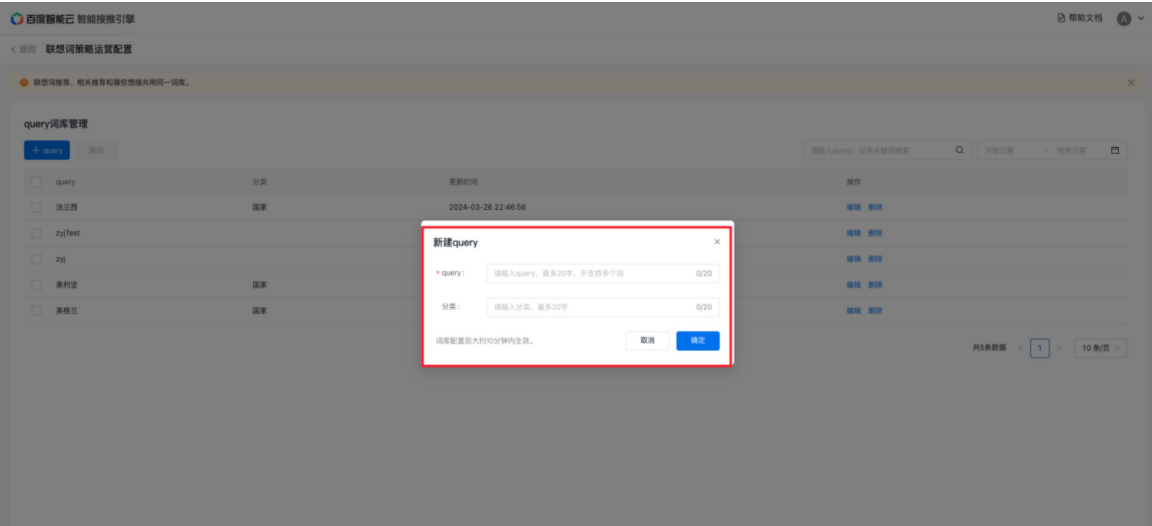
操作步骤

1登录进入智能推荐引擎，进入【运营参谋】--【策略运营】选择对应【应用】--【联想词策略运营】，进行策略配置操作。

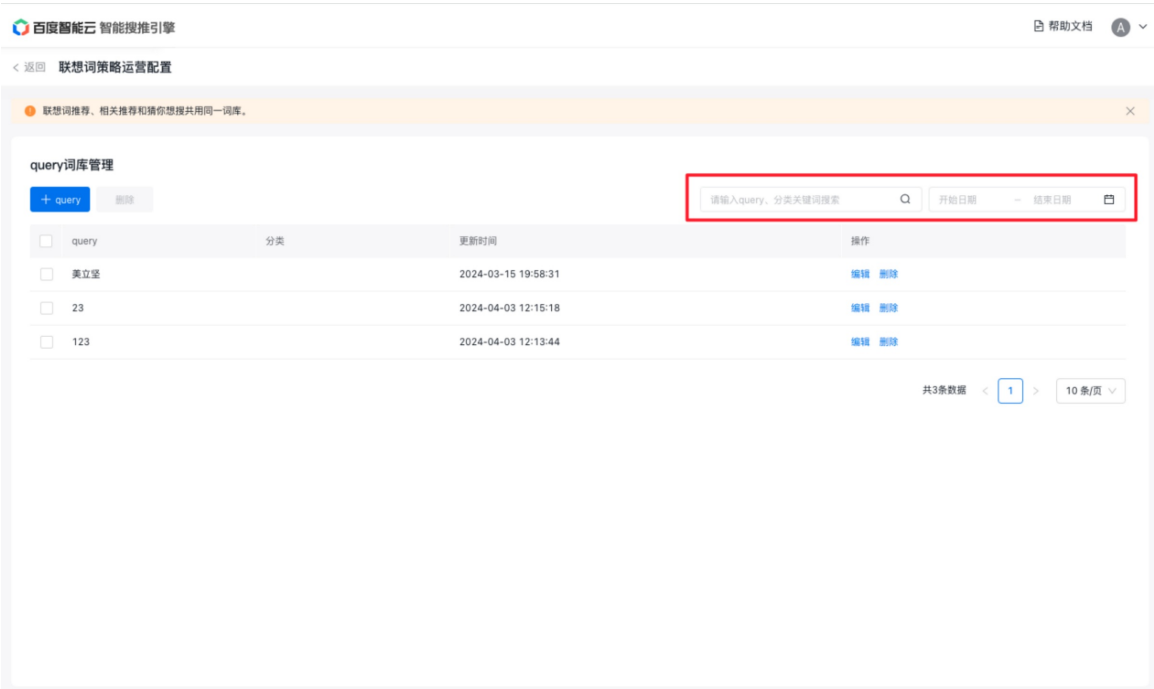


2query词库管理：

(1) 新建query：进入【query词库】页面，点击【+query】新建query并输入分类、搜索直达URL和标签，词库配置后大约10分钟内生效。



(2) 词库筛选编辑：可以使用搜索和日期筛选功能，定位到想要寻找的词典，对指定词典进行编辑、删除和置顶操作。



猜你想搜策略运营

概述

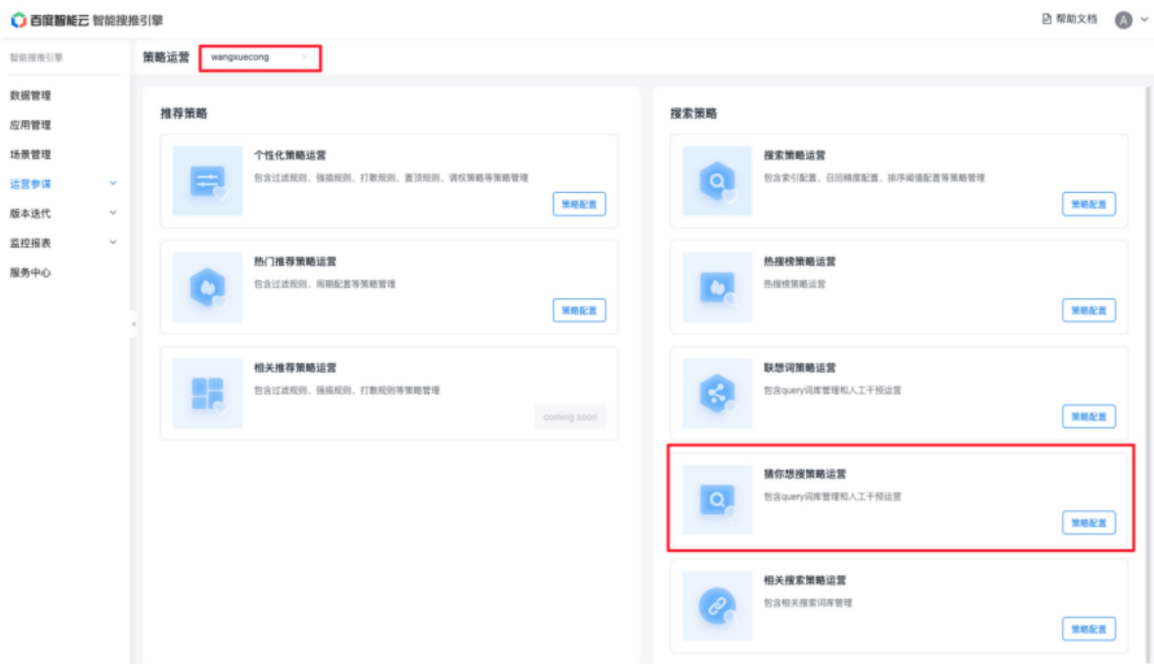
根据用户的搜索历史和偏好，推荐相关的搜索query，提供更加个性化的搜索体验。

前提条件

- 1应用已经创建完成。
- 2已经接入物料数据。
- 3服务已经启用。

操作步骤

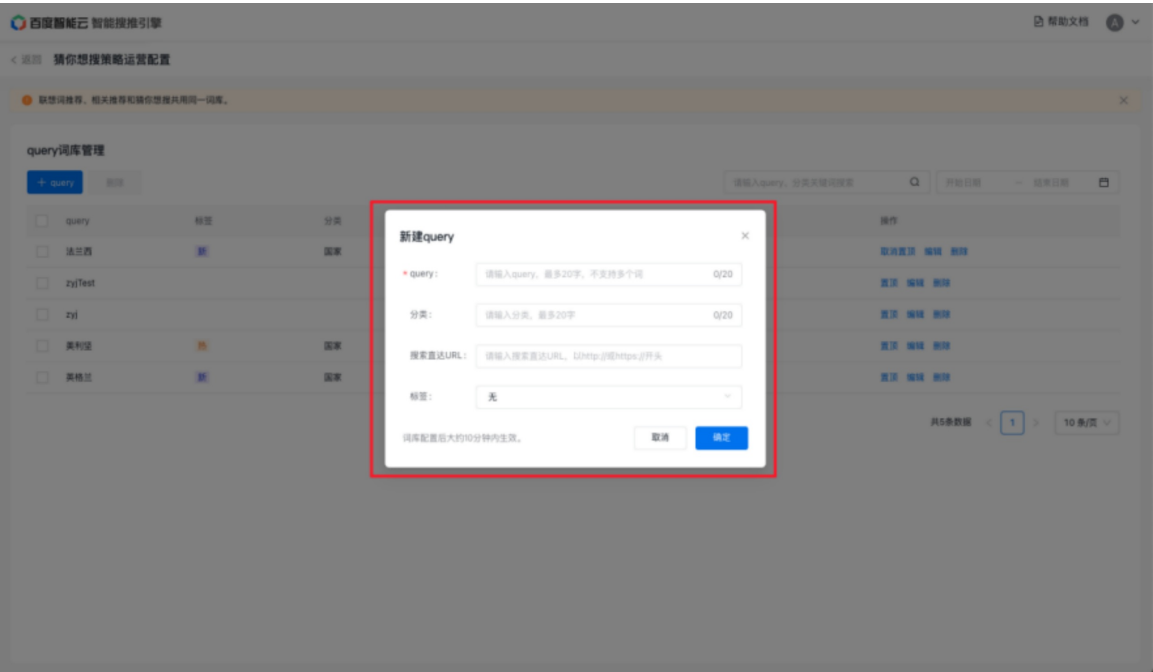
1登录进入智能推荐引擎，进入【运营参谋】--【策略运营】选择对应【应用】--【猜你想搜策略运营】，进行策略配置操作。



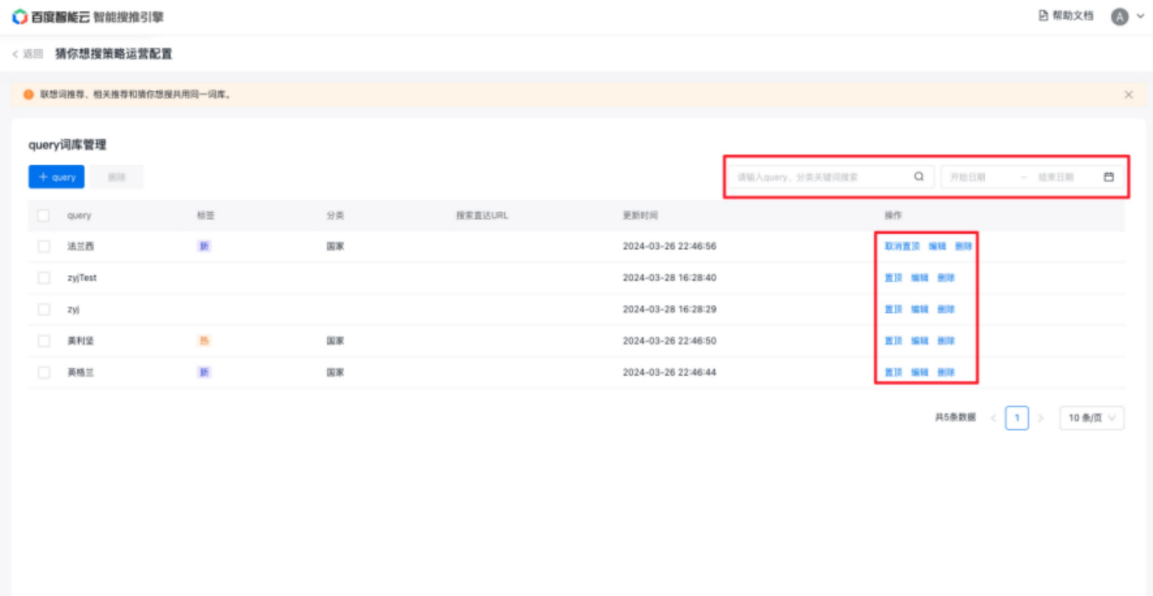
2query词库管理：

- (1) 新建query：进入【query词库】页面，点击【+query】新建query并输入分类、搜索直达URL和标签，词库配置后大约10

分钟内生效。



(2) 词库筛选编辑：可以使用搜索和日期筛选功能，定位到想要寻找的词典，对指定词典进行编辑、删除和置顶操作。



用户运营

概述

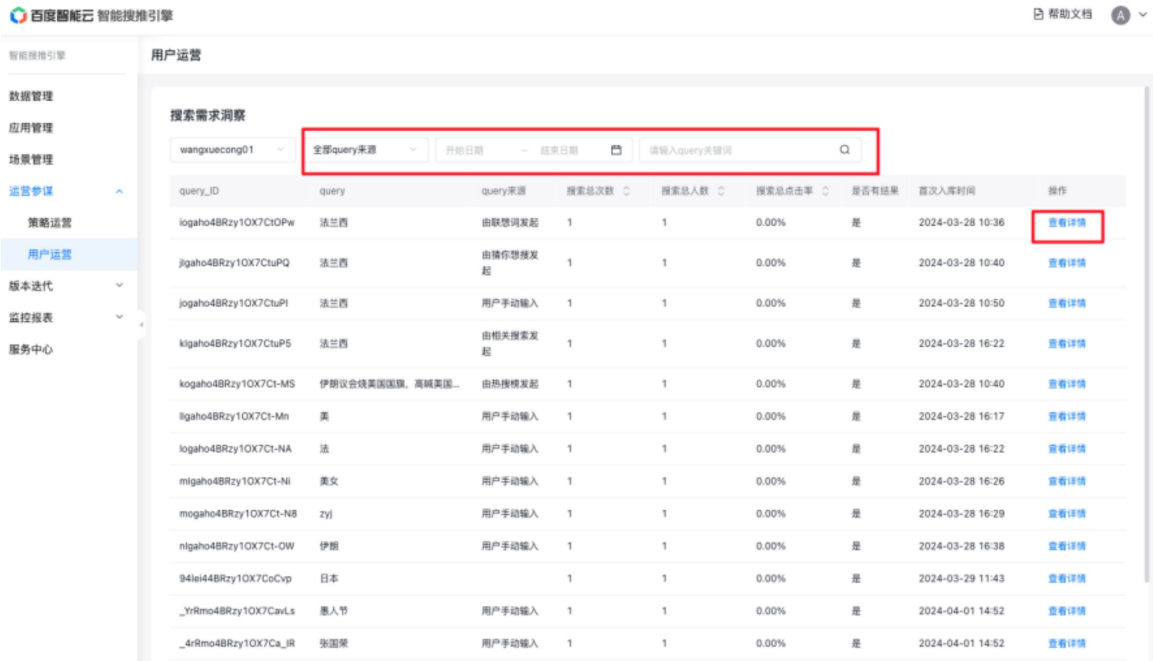
对用户的搜索需求进行调查，查看query来源，查看特定时间段的搜索指标数，用于指导搜索优化。

前提条件

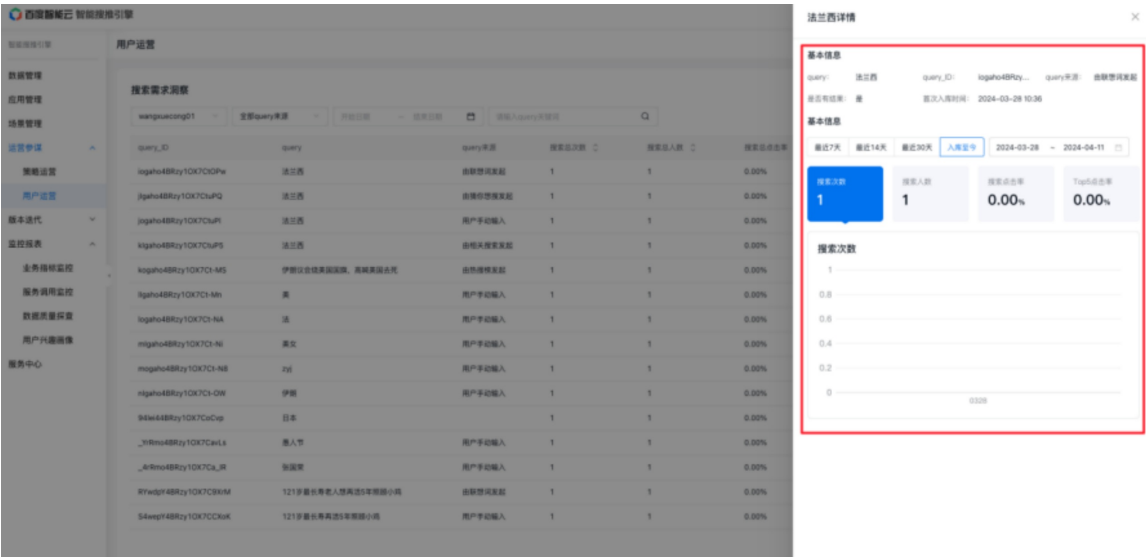
- 1应用已经创建完成。
- 2已经接入物料数据。
- 3服务已经启用。

操作步骤

1登录进入智能推荐引擎，进入【运营参谋】--【用户运营】--进入【搜索需求洞察】页面，可输入关键字搜索查询query。

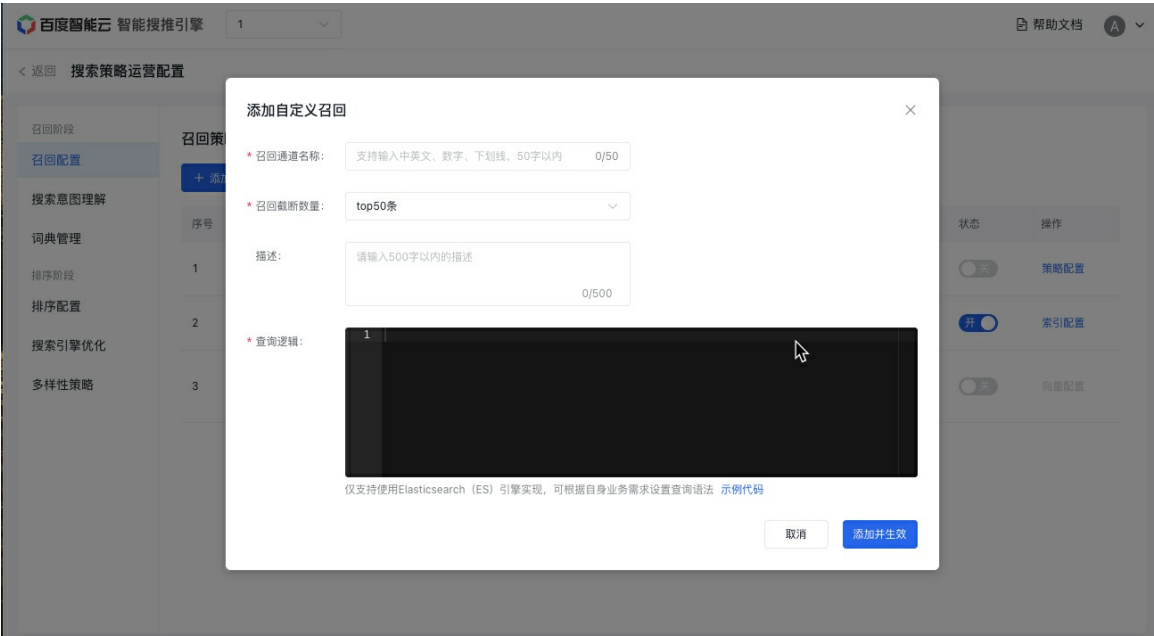


2查看详情：点击对应的query【查看详情】按钮，可查看首次入库时间、搜索次数、搜索人数、Top5点击率、是否有结果、query来源等指标。



搜索自定义召回

配置说明 自定义搜索召回可以配置过滤条件，检索条件，及排序条件三个字段，分别按照如下规则配置，搜索自定义召回按照如下格式配置，不属于配置关键字的部分将被忽略。



```
{
  "query": {
    "bool": {
      "filter": [{}],
      "should": []
    }
  },
  "sort": {}
}
```

过滤条件

配置在 query/bool/filter下，使用list类型，具体可参考[过滤搜索结果](#)

如希望只召回cities字段为上海的物料，可按如下方式配置。

```
{
  "query": {
    "bool": {
      "filter": [
        // 需要配置的自定义过滤条件
        {
          "bool": {
            "must": [
              {
                "term": {
                  "cities": "上海"
                }
              }
            ]
          }
        }
      ]
    }
  }
}
```

配置的过滤条件会和原本的过滤条件共同生效，因此一般不建议配置过滤条件，因为可以通过接口传入过滤条件对所有召回进行过滤，除非希望自定义召回有特定的过滤条件才需要配置。
基于es存储存储字段类型区别，字段名需要加 ".keyword" 后缀，对字段名配置有疑问的可以询问研发人员。

查询条件

配置在 query/bool 下，具体可参考[使用查询规则搜索](#)

如一个 match_phrase 召回，可以使用如下召回语句，其中需要用 "@query" 来标记原始query，该query是对原始的搜索query进行纠错(若启用)后的结果，但不会受到其他搜索配置如同义词，搜索意图理解和词权重的影响。

```
{
  "query": {
    "bool": {
      "should": {
        "bool": {
          "must": [
            {
              "match_phrase": {
                "title": {
                  "query": "@query"
                }
              }
            }
          ]
        }
      }
    }
  }
}
```

预置检索函数

仅供开发使用，直接使用关键词召回如 全覆盖检索 full_cover_query，精简检索 reduced_query，核心词检索 kernel_word_query 的搜索语句，只能配置其中一个，这样配置可以使用全部关键词召回相关的搜索配置，包括纠错，分词，同义词，词权重，紧密度等。

```
{
  "query": {
    "bool": {
      "should": "@full_cover_query"
    }
  }
}
```

排序条件

配置在 sort 下，无特殊要求，参考 [排序搜索结果](#)

```
{
  "sort": [
    {
      "publish_time": {
        "order": "desc"
      }
    }
  ]
}
```

如希望按照发布时间排序，可以按照如下配置。需要注意，这只会影响一路召回的排序结果，最终还会和其他召回进行合并排

序，因此最终结果并不会按照发布时间排序，但是可以一定程度上增加搜索结果中新物料的比例。

监控报表

服务监控

服务大盘展示服务性能指标，开发人员或者运维人员通过服务大盘了解推荐服务健康度。全方位的服务监控包含了接口调用、接口异常、服务健康度、用户交互等统计数据，保障核心指标数据监控无死角。

概述

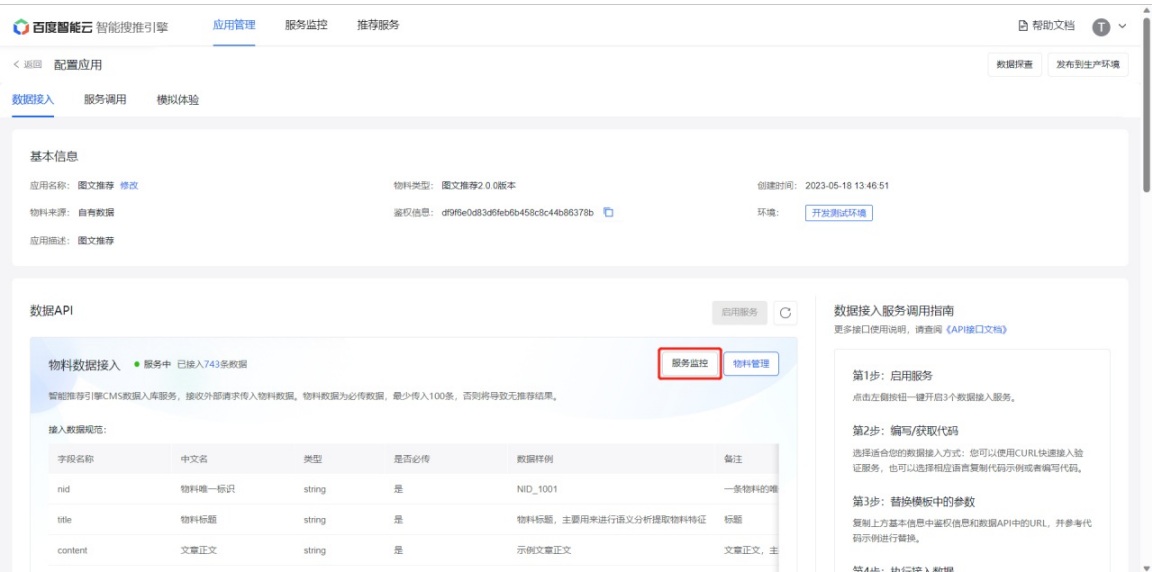
服务调用监控展示了服务具体的指标数据，从调用统计刻画服务，如调用次数、调用成功次数、调用失败次数、调用失败率和QPS峰值等核心指标，并可以查询指标趋势。

前提条件

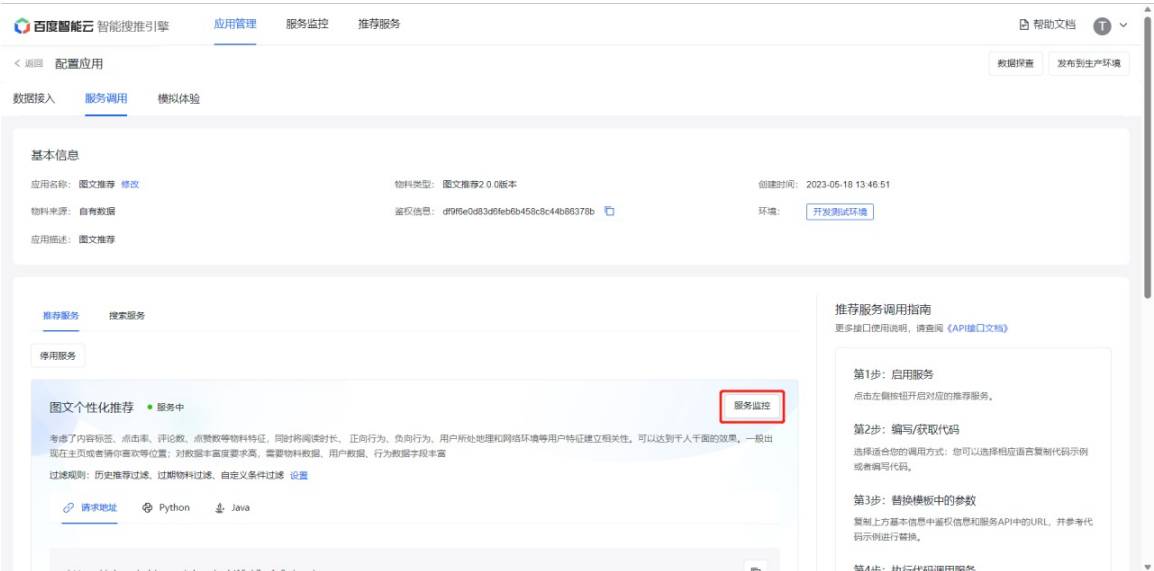
- 1应用已经创建完成。
- 2数据接入服务和推荐服务已经开启。

操作步骤

- 1登录进入智能推荐引擎。
- 2查看服务调用统计的方式有两种：应用中查看和服务监控中查看。
- 3应用中查看：在应用列表页面选择相应的应用点击【配置】按钮进入配置页面。
 - (1) 在数据接入页签下查看数据接入服务的调用信息；



- (2) 在配置页面，在服务调用页签下查看推荐服务的调用信息。



4服务监控中查看：



点击导航进入服务监控页面，在左侧的应用列表下选择生产环境或开发测试环境下的某一应用，右侧服务列表会显示该应用对应的服务。在应用列表中点击【运维监控】按钮进入应用的监控详情页。在服务列表中点击【调用详情】按钮进入调用详情页。

参数	说明
调用总次数	某个服务从开启到当前时间被应用调用的次数总和。
调用次数趋势	某个服务近14天每天被调用次数趋势图。
昨日调用失败率	该服务昨日调用失败的次数与该服务昨日总调用次数的比值
昨日99分位延时	统计昨天每次调用所消耗的时间，并将这些调用按时间从小到大的顺序进行排序，并取出排序为当天总次数 * 99%的时间值。

在应用的监控详情页中，在列表中点击【调用详情】按钮会进入到调用详情页。

百度智能云 智能搜推引擎

应用管理 服务监控 推荐服务

帮助文档 A

< 返回 调用详情

时间筛选: 不限 最近7天 最近14天 最近30天

开始日期 ~ 结束日期

请求用户: 请输入用户ID

traceid	调用时间	请求用户	调用耗时	返回数据量	调用状态	操作
192045040557021694792404845001255019251	2023-07-13 15:37:03	isfcrd1n-21_1	1.39	10	成功	推荐链路
031041002439809590187901040883466228778	2023-07-13 15:35:18	isfcrd1n-21_1	1.73	10	成功	推荐链路
321987555800384172006567254487902129502	2023-07-13 15:31:21	uid_7	1.15	10	成功	推荐链路
181574195268020720884834497658303926383	2023-07-13 15:26:43	isfcrd1n-21_1	1.49	10	成功	推荐链路
034051894859950825189777331443928323134	2023-07-13 15:23:09	isfcrd1n-21_1	1.37	10	成功	推荐链路
088077968242248919179497306464404549963	2023-07-13 15:11:02	uid_7	1.36	10	成功	推荐链路
015769636948257841152449936837307540396	2023-07-13 15:10:57	uid_7	1.27	10	成功	推荐链路
201457165881912127292455743833719030355	2023-07-13 14:57:31	isfcrd1n-21_1	1.3	10	成功	推荐链路
080405473662511132834800385950295628774	2023-07-13 14:46:03	isfcrd1n-21_1	1.88	10	成功	推荐链路
128853797890817889118606548671620498804	2023-07-13 14:45:52	isfcrd1n-21_1	1.41	10	成功	推荐链路

共38条数据 1 2 3 4 10 条/页

参数	说明
调用次数	某个服务某一天被应用调用的次数。
调用成功次数	某个服务某一天被应用调用成功的次数
异常调用失败次数	某个服务某一天被应用调用失败的次数
调用失败率	某个服务某一天被应用调用失败的次数占当天调用的次数的比率。
QPS峰值	某个服务某一天每秒调用的最大值。
99分位时延	T-1日调用时长99分位值，统计每天每次调用所消耗的时间，并将这些时间按从小到大的顺序进行排序, 并取出排序为总次数 * 99%的时间。

推荐链路

概述

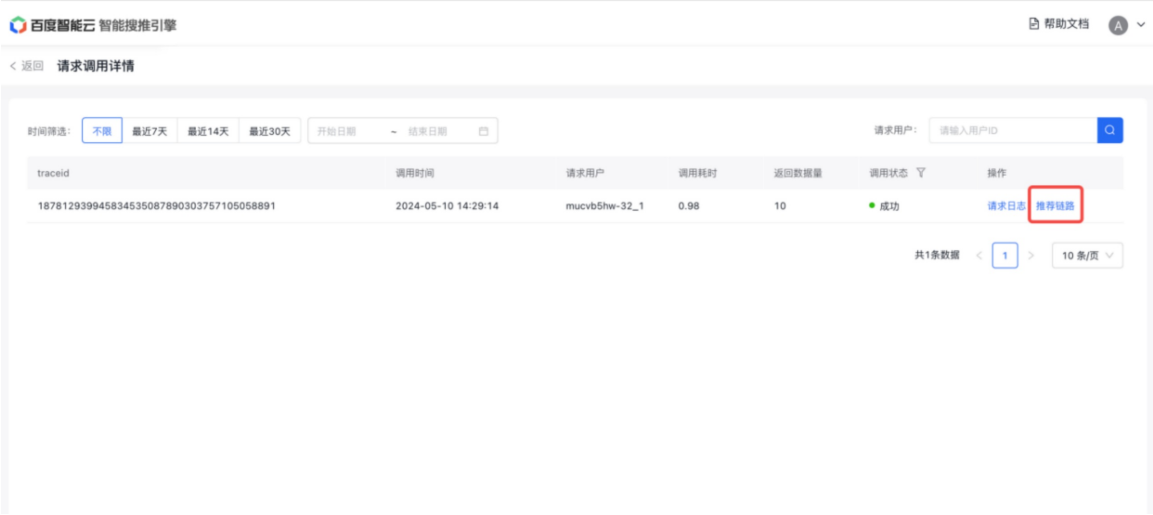
推荐过程观测是排查效果问题的利器，相当于医生的检测仪器，效果问题有效诊断才能对症治疗。

前提条件

- 1应用已经创建完成。
- 2数据接入服务和推荐服务已经开启。

操作步骤

1在服务的调用详情页中，用户在搜索框中输入用户id或选择其他查询条件，点击“查询”按钮系统根据查询条件匹配，在调用详情列表中显示查询结果，清空输入或选择框，列表返回初始状态。调用详情列表支持查看日志和推荐链路（暂仅支持个性化推荐服务）。点击【推荐链路】按钮，会进入到推荐链路可视化分析页面。



2在推荐链路可视化分析页面中，支持对桑基图画布进行缩放和下载操作。在页面的右上方点击【放大】或【缩小】按钮，可实现桑基图画布的放大缩小。点击【下载】按钮，可将画布下载为图片文件。



除了对画布整体进行操作外，支持用户点击桑基图画布的节点（深色色块），会弹出节点下物料id列表（例如下方的热门召回、排序Top10节点）。

百度智能云 智能搜索引擎

应用管理 服务监控 推荐服务

帮助文档

返回 推荐链路

推荐链路桑基图分析

物料候选池 召回策略 排序阶段 重排: 调权策略 重排: 打散策略 重排: 强插策略 曝光阶段

地理位置召回 搜索需求召回 最新发布召回 协同过滤召回 全部召回 正反馈召回 热度TOP召回 短期兴趣召回 历史推荐过滤

排序1-20 排序21-40 排序40+ 调权重排1-20 调权重排21-40 调权重排40+ 打散重排1-20 打散重排21-40 打散重排40+ 强插TOP10 强插10+ 最终展现 最终未展现

打散过滤

链路节点: 最新发布召回

物料ID	物料名称	标签	发布时间
562	夏威夷火山喷发, 周围居民紧急撤离	美食	2023-01-02 00:00:15
68107	日本“宝宝哭相扑”, 吓到“虎躯一震”	风俗民俗习惯	2022-03-29 21:20:00
734	沙特土豪请客剩百斤饭菜摆满全扔	奇人异事	2022-03-29 21:20:00
68214	夏威夷火山喷发后地面出现巨大裂缝	奇景	2022-03-29 21:20:00
68054	长滩岛临时关闭, 游客举行烟花派对	风俗民俗习惯	2022-03-29 21:20:00
68293	2男子24小时打卡美国23个州破纪录	奇人异事	2022-03-29 21:20:00
371	痛哭! 男童这样帮4岁女孩克服怕兔子	家居	2023-01-02 00:00:15
705	印度新郎骑马娶亲, 亲友盛装跳舞	奇人异事	2022-03-29 21:20:00
747	私建空中楼阁, 俄业主被告上法庭	奇人异事	2022-03-29 21:20:00
68076	日本推出橡皮筋瘦脸法, 10分钟见效	风俗民俗习惯	2022-03-29 21:20:00

共50条数据 1 2 3 4 5 10条/页

用户画像

概述

兴趣旅程提取组件将用户的历史行为映射到连贯的兴趣旅程中，同时支持查看用户兴趣点强度和查询某个兴趣点的时间变化趋势，从而挖掘用户的长短期兴趣。这有助于更好地理解用户的兴趣和行为，从而提供更深入、更可解释的用户理解。

前提条件

- 1应用已经创建完成。
- 2数据接入服务和推荐服务已经开启。
- 3该用户有使用行为。

操作步骤

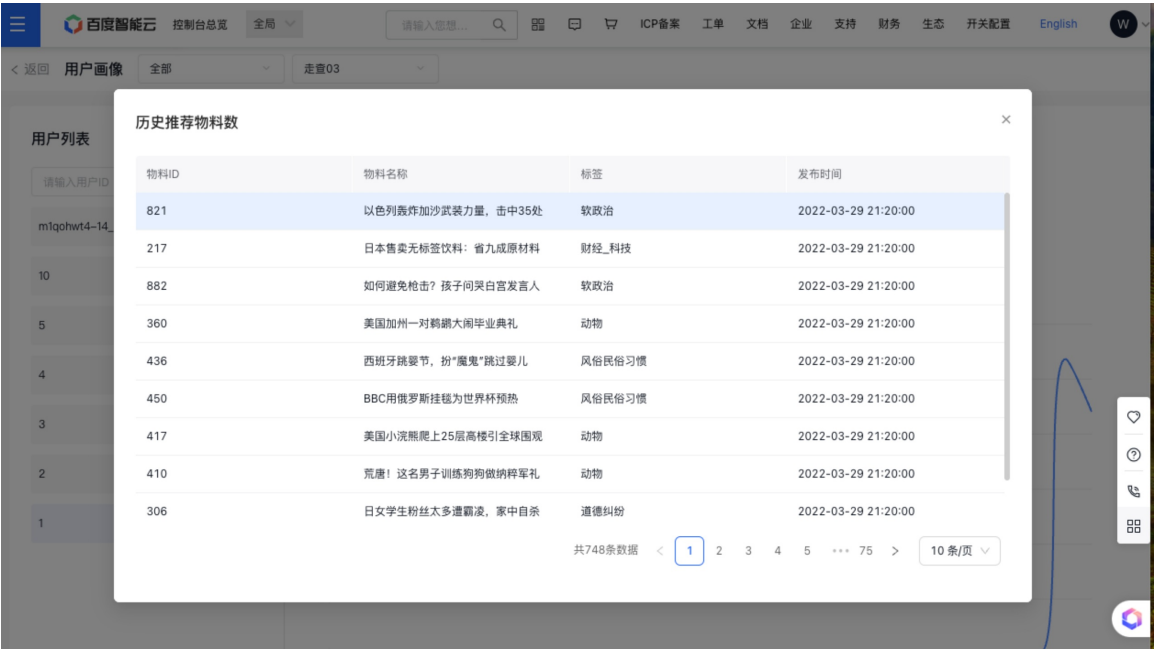
1登录进入智能推荐引擎，进入【服务中心】模块，点击【用户画像】卡片。



2输入用户id，点击【添加抽样】按钮，系统会抽样该用户行为数据进行分析，便于跟踪模拟用户使用效果和分析。



3点击历史推荐物料【详情】按钮，查看历史推荐列表。



业务指标监控

概述

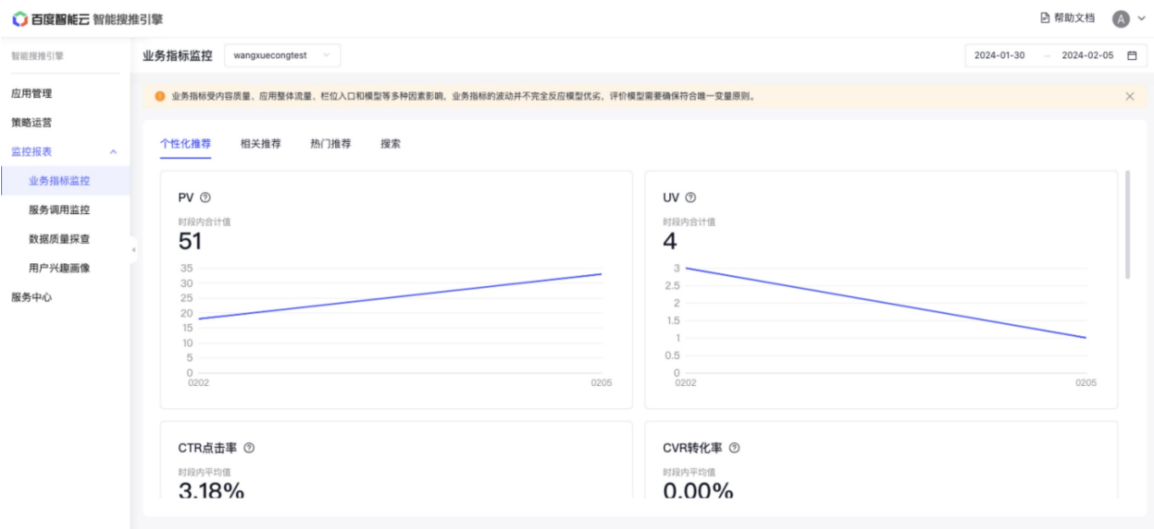
通过对个性化推荐、相关推荐、热门推荐和搜索等场景的业务效果指标监控，帮助用户了解推荐系统的运行情况和效果，为运营提供数据支持，提高用户满意度和转化率，同时为企业运营提供数据支持和参考。指标包括：UV/PV、点击率、转化率，用户覆盖率、物品覆盖率等。

前提条件

- 1应用已经创建完成。
- 2数据接入服务和推荐服务已经开启。
- 3该用户有使用行为。

操作步骤

1登录进入智能推荐引擎，选择【监控报表】，点击【业务指标监控】，选择需要数据的应用以及场景进行筛选查看。



2指标解释：统计范围涵盖个性化推荐、相关推荐、热门推荐和搜索，场景适用图文、商品、媒资。统计周期支持选择：近7天、近14天、近1月、自定义，自定义支持近3个月数据。

指标	定义
召回策略ctr分布	所选周期内各路召回策略的点击总数/曝光总数
召回策略cvr分布	所选周期内各路召回策略的购买总数/点击总数（该指标图文物料不含）
物品覆盖率	所选周期内推荐出的物品数占总物品的比例
用户覆盖率	所选周期内请求的用户数占总用户数的比例
pV	所选周期内调用推荐接口的次数，按行为粒度统计
uv	所选周期内调用推荐接口的人数，按用户粒度统计
ctr	所选周期内用户点击总数/曝光总数，按行为粒度统计
cvr	所选周期内用户购买总数/点击总数，按行为粒度统计，该指标图文物料不含
搜索无结果query数	所选周期内用户搜索的query无结果的数量
新增query数	所选周期内用户新增的query的数量

版本迭代

AB实验

概述

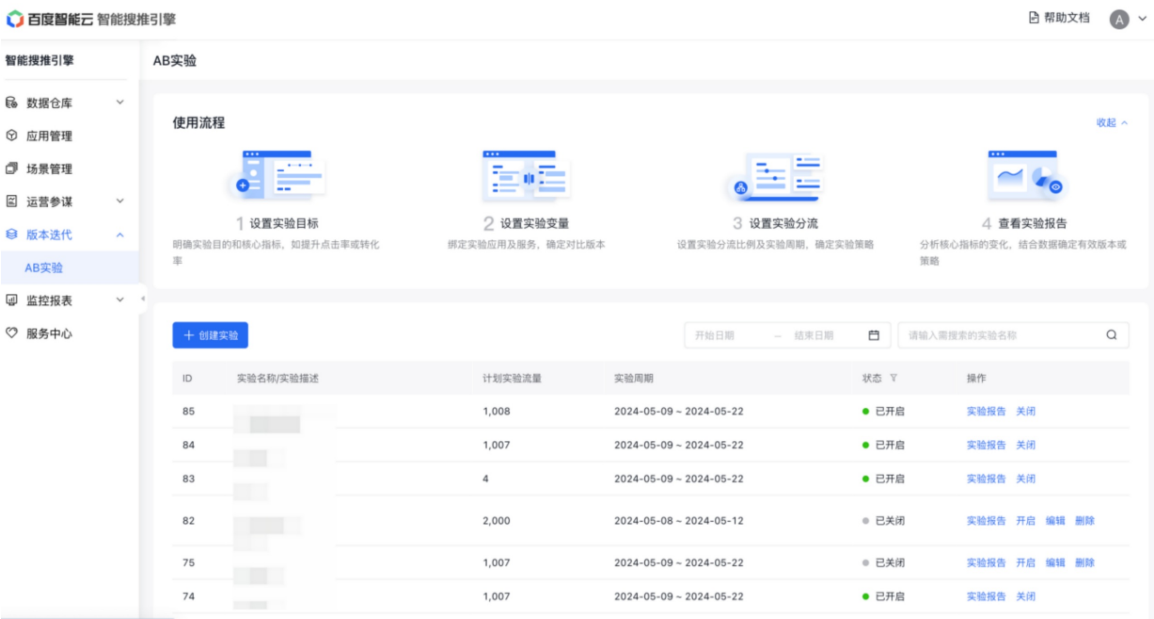
在线上流量中取出一小部分，完全随机地分给原策略A和新策略B，得到对于两种策略相对效果的准确估计。整体流程是设置实验目标，设置实验变量，设置实验分流，查看实验报告。

前提条件

- 1应用已经创建完成。
- 2数据接入服务和推荐服务已经开启。

操作步骤

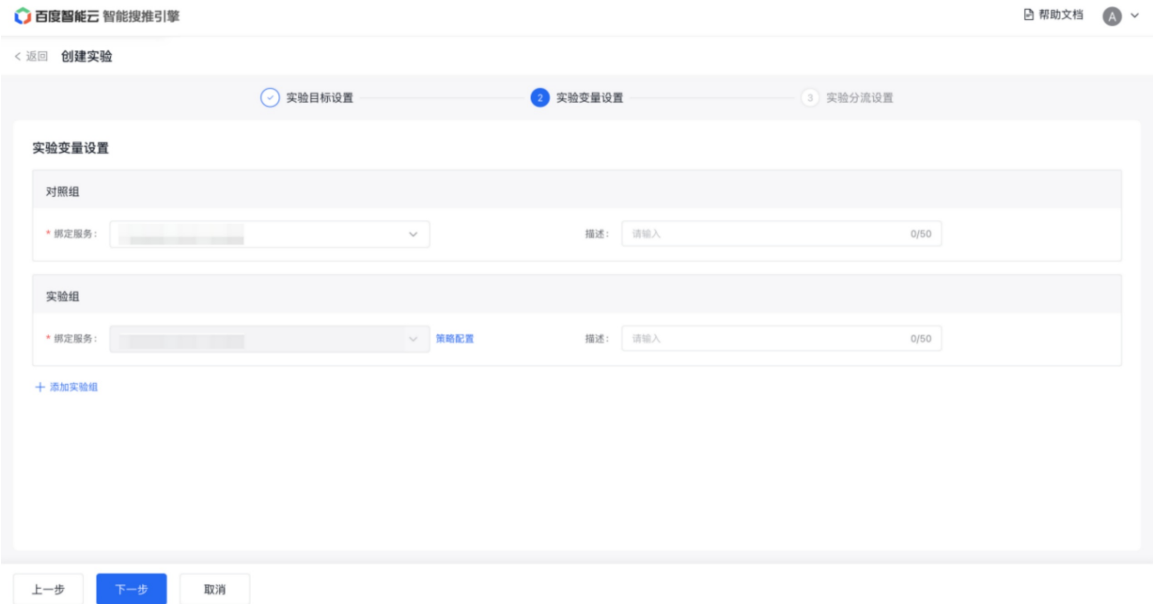
- 1登录进入智能推荐引擎，点击【版本迭代】下的【AB实验】。



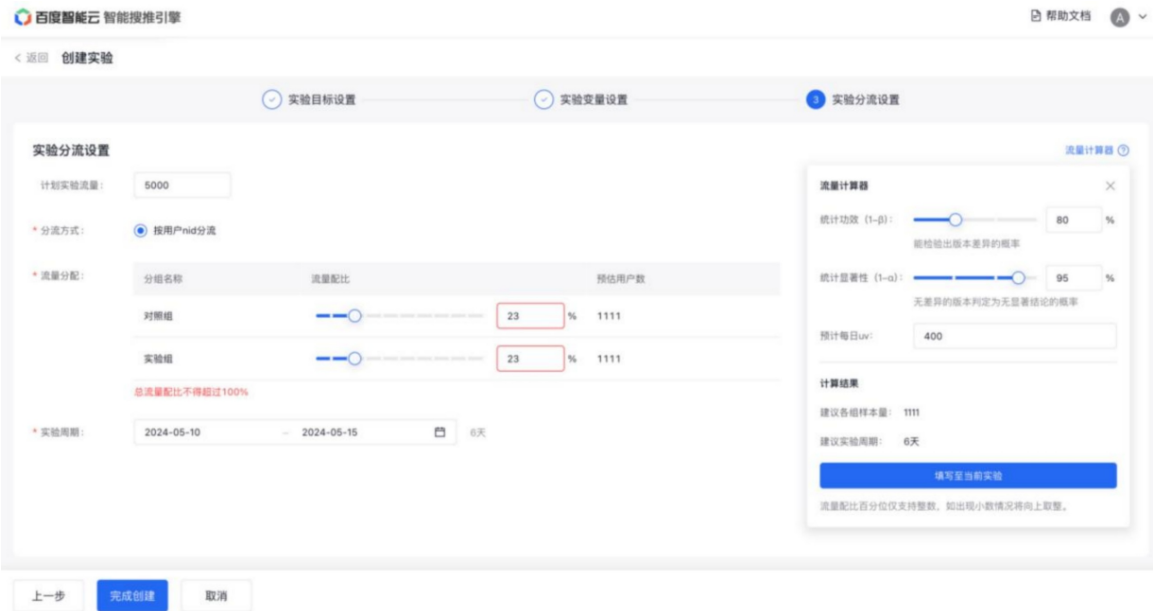
- 2创建实验：点击【+创建实验】按钮，弹出“创建实验”弹框。
 - (1) 实验目标设置：输入实验名称、核心指标基线值、预期提升指标值，并勾选需要的辅助指标。



(2) 实验变量设置：设立对照组和实验组，并绑定对应服务，对实验组进行【策略配置】操作。



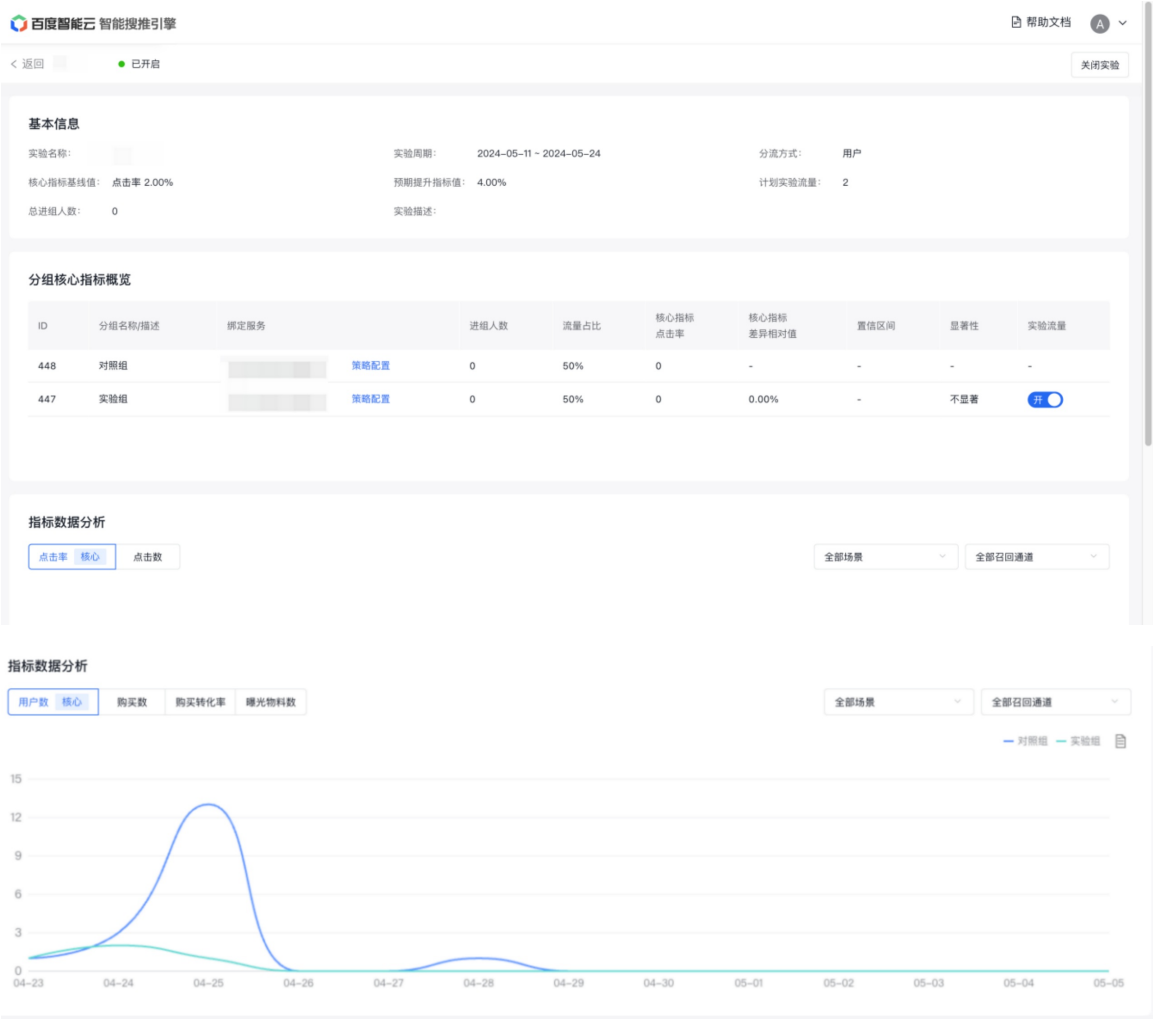
(3) 实验分流设置：填写计划实验流量，进行流量分配（实验组和流量组总量配比不超过100%），可使用流量计算器的计算结果作为参考来调整实验组和对照组的流量，并设定实验周期，点击【完成创建】。



参数	参数说明
核心指标	用来决策实验功能是否符合预期的成功指标。
辅助指标	辅助判断实验对其他因素的影响，与核心指标互斥。
统计功效 (1-β)	统计功效 = 1 - 第二类错误的概率（β），统计功效在现实中表现为：假设我的新策略是有效的，有多大概率在实验中检测出来。默认值为80%，支持调整为80%、90%、99%。
统计显著性 (1-α)	统计显著性=1 - 第一类错误的概率（α），也称“置信水平、置信度、置信系数”，描述实验结果的可信度。本系统置信度参数默认值设置为95%

流量计算器| 计算结果仅作为参考，不是实验真正的流量和试验周期。实验周期=n/每日uv | 核心指标差异值 当前实验版本相对于对照版本的绝对差异/基准版本值。 置信区间 计算核心指标差异值的置信区间，由样本统计量构造的总体参数的估计区间。

3 查看实验报告：分析核心指标的变化，结合数据确定有效版本或策略。当达到截止时间，出现实验结论，通过显著性判断，是否继续实验。



说明：

实验状态：默认开，状态为开且满足运行开始时间，流量开始进入实验；关闭实验流量不分流。

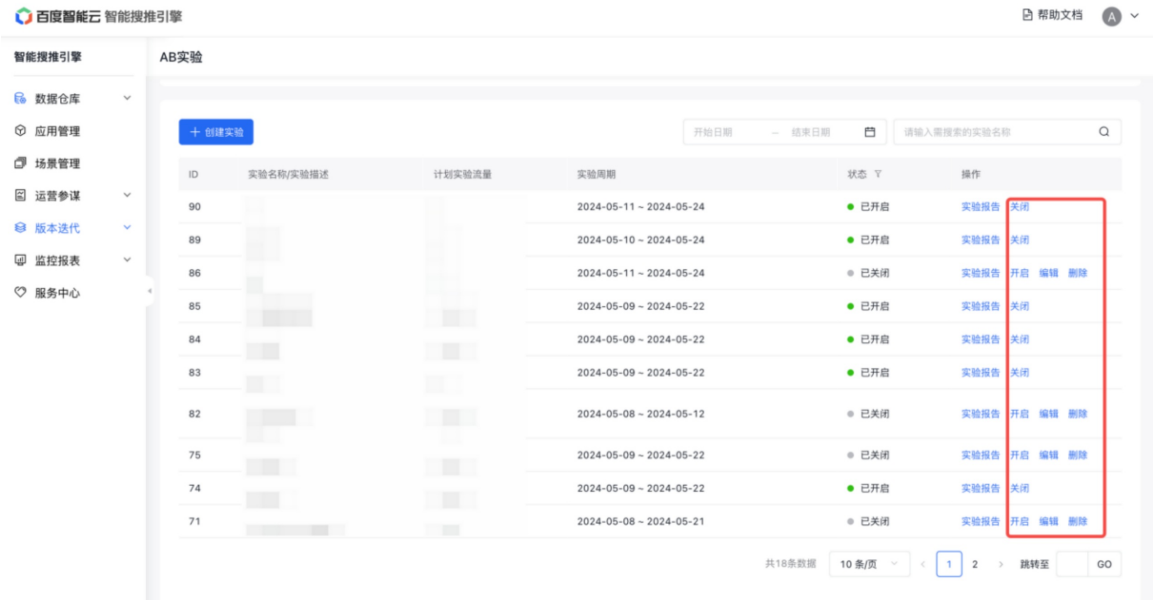
实验结论：当达到运行结束时间后生成最终实验结果，共有三种结果：

（1）正向显著：截止2024-01-24 01:00统计已正向显著，实验组n为优胜版本，建议线上应用此配置，可提高核心指标表现。

（2）均不显著：截止2024-01-24 01:00尚未统计显著，建议扩大流量或者延长实验时间继续观察。

（3）负向显著：截止2024-01-24 01:00统计业务负向显著，可结束实验。

4 开启关闭实验：可对实验报告进行筛选、编辑、开启、关闭等操作。若实验状态为“已开启”，可进行查看实验报告和关闭操作。若实验状态为“已关闭”，可进行查看实验报告、开启、编辑、删除的操作。



版权内容

创建内容集

百度内容资讯由百家号等内容平台提供资源支持，垂类内容和创作者都可根据需求灵活调整与订制。百家号拥有海量不同垂类的签约作者，现阶段约为340万，共同组成多样的内容创作者矩阵，涵盖媒体、MCN、PGC、UGC等各个方面，覆盖百种垂类，互动内容形态多样（图文、图集、视频、短视频、问答、互动等）。财经&科技类的自媒体作者数量约为34万，占比10%，涵盖个人作者、机构作者、媒体作者等（金融界、同花顺等）；支持日均生产财经&科技内容量超过3万篇。

概述

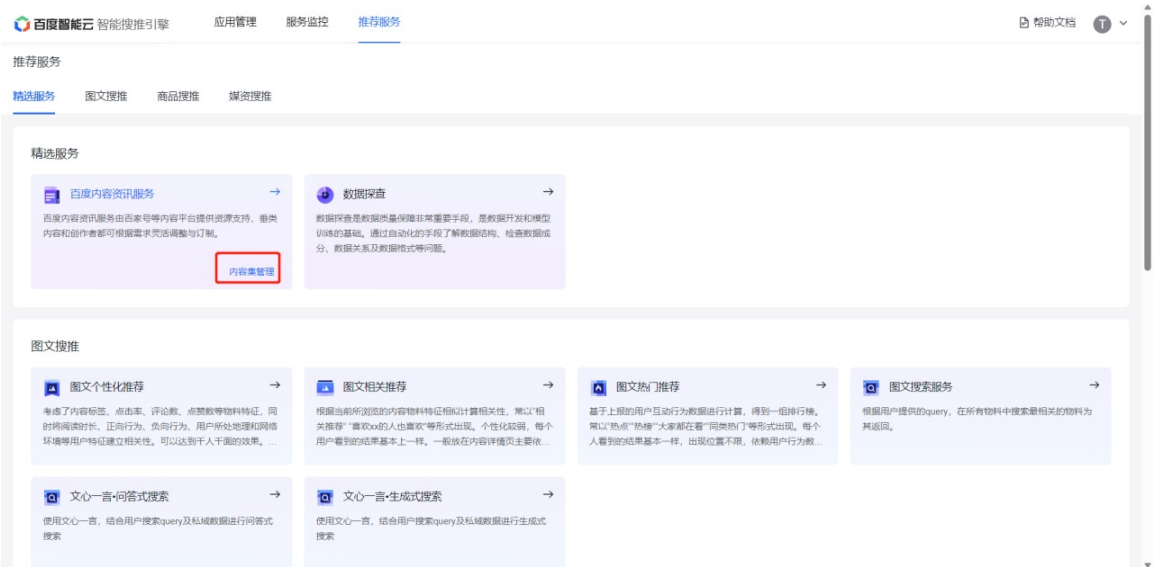
通过设定筛选条件圈选内容资讯，并可以将其定义为某类内容集，供推进应用绑定作为物料数据。用户可以根据实际需求按照内容和作者两个视角自主设定筛选规则。

前提条件

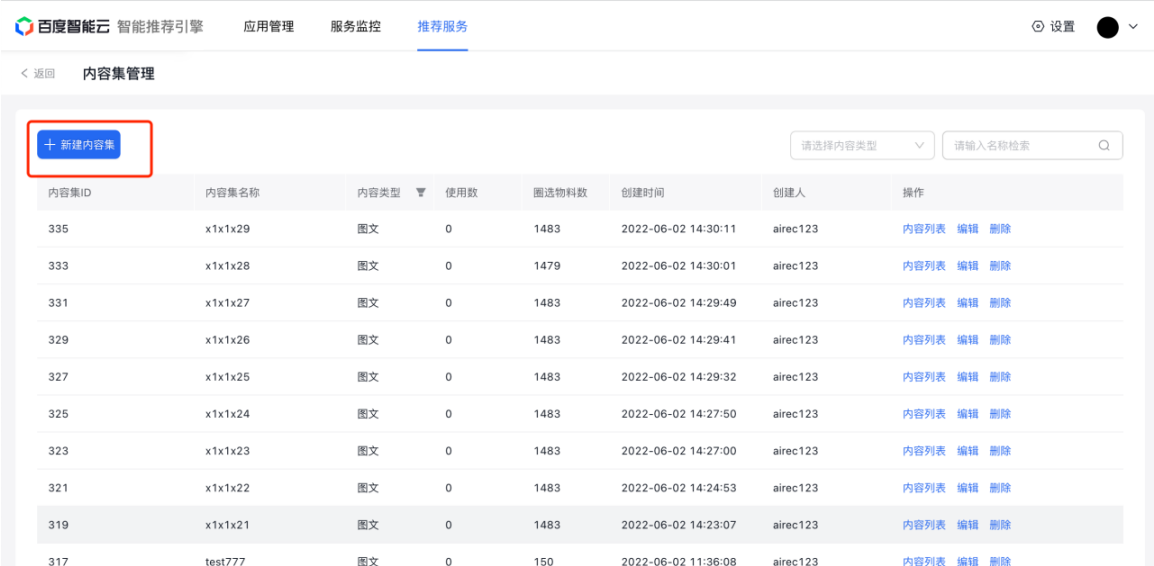
1租户开通了内容服务，内容集的入口是开放可用的。

操作步骤

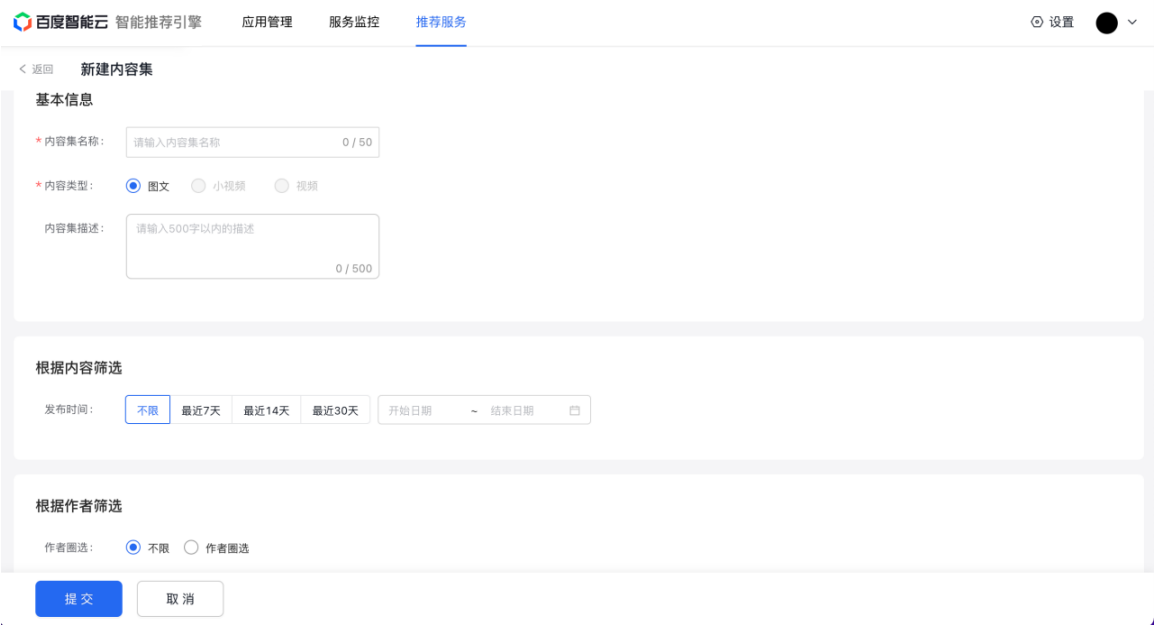
1登录进入智能推荐引擎，点击【推荐服务】找到【内容资讯服务】，点击【内容集管理】。



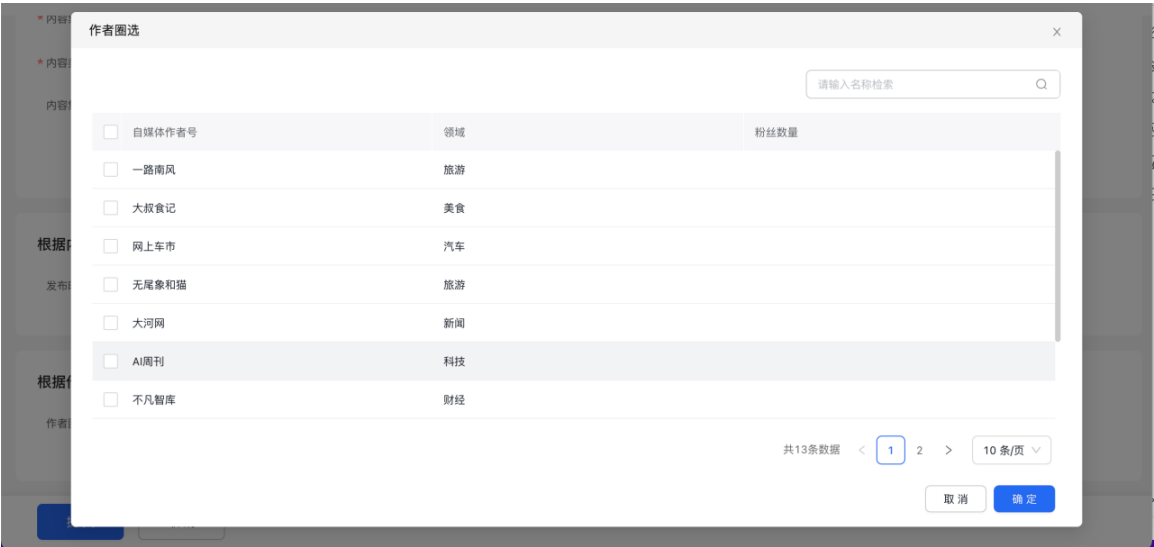
2进入内容集列表页面，点击左上角的【新建内容集】按钮，创建新的内容集。



3进入新建内容集页面，输入内容集基本信息和圈选条件，当前支持根据内容发布时间筛选和根据作者筛选内容。点击【保存】按钮，提示【保存成功】。



- (1) 发布时间可选近7天、近14天、近30天或者自定义时间段。
- (2) 作者可选不限，或者指定名称选择。



4进入内容集列表页，查看新建的内容集圈选的物料数，检查是否符合预期数，点击【内容列表】按钮，查看内容。

百度智能云 智能推荐引擎 应用管理 服务监控 推荐服务

设置

返回 内容集管理

+ 新建内容集

请选择内容类型 请输入名称检索

内容集ID	内容集名称	内容类型	使用数	圈选物料数	创建时间	创建人	操作
335	x1x1x29	图文	0	1483	2022-06-02 14:30:11	airec123	内容列表 编辑 删除
333	x1x1x28	图文	0	1479	2022-06-02 14:30:01	airec123	内容列表 编辑 删除
331	x1x1x27	图文	0	1483	2022-06-02 14:29:49	airec123	内容列表 编辑 删除
329	x1x1x26	图文	0	1483	2022-06-02 14:29:41	airec123	内容列表 编辑 删除
327	x1x1x25	图文	0	1483	2022-06-02 14:29:32	airec123	内容列表 编辑 删除
325	x1x1x24	图文	0	1483	2022-06-02 14:27:50	airec123	内容列表 编辑 删除
323	x1x1x23	图文	0	1483	2022-06-02 14:27:00	airec123	内容列表 编辑 删除
321	x1x1x22	图文	0	1483	2022-06-02 14:24:53	airec123	内容列表 编辑 删除

说明：
新建内容集后仅支持修改基本信息，暂不支持修改规则，确实需要修改建议删除后重新创建。

5进入内容列表页，查看内容是否符合预期。

百度智能云 智能推荐引擎 应用管理 服务监控 推荐服务

设置

返回 内容列表

内容集：x1x1x29

标题检索 请输入标题检索

ID	缩略图	标题	作者	发布时间	垂类	原文链接	是否上架	操作
449		移动支付市场要“变天”？对手不是“国家队”，创始人也姓“马”449	无尾象和猫	2022-05-29 09:49:17	科技-移动支付	http://baijiahao.baidu.com/s?id=1689823329416149921	是	详情 删除
448		移动支付市场要“变天”？对手不是“国家队”，创始人也姓“马”448	无尾象和猫	2022-05-29 09:49:17	科技-移动支付	http://baijiahao.baidu.com/s?id=1689823329416149921	是	详情 删除
447		移动支付市场要“变天”？对手不是“国家队”，创始人也姓“马”447	无尾象和猫	2022-05-29 09:49:17	科技-移动支付	http://baijiahao.baidu.com/s?id=1689823329416149921	是	详情 删除
446		移动支付市场要“变天”？对手不是“国家队”，创始人也姓“马”446	无尾象和猫	2022-05-29 09:49:17	科技-移动支付	http://baijiahao.baidu.com/s?id=1689823329416149921	是	详情 删除

使用内容集

概述

内容集是内容资讯的集合，主要有两个用途：一是可以查询或者管理内容集里的内容；二是可以绑定到应用上作为推荐引擎的物料数据，实现内容的个性化分发，本节主要介绍作为第二种用途的使用方法。

前提条件

- 1租户开通了内容服务，内容集的入口是开放可用的。
- 2内容集已经创建完成。
- 3当内容集用作推荐物料时，需要接入系统完成行为数据采集并将行为数据和用户数据接入推荐引擎中。

操作步骤

1登录进入智能推荐引擎，进入【应用管理】--点击【创建应用】依次填写应用各项参数--【推荐服务】选择【百家号图文推荐】，选择需要使用的内容集，点击【创建】按钮，进入创建应用提示页面。

百度智能云 智能搜推引擎

帮助文档

返回

创建应用

基本信息

* 应用名称:

请输入50字以内名称

0/50

* 推荐服务:

请选择推荐服务

* 部署环境:

开发测试环境

生产环境

QPS预估:

5次/秒

应用描述:

请输入500字以内的描述

0/500

确定

取消

2在应用创建成功提示页面点击【立即配置】，或者在应用列表页面选择相应的应用点击【配置】按钮。进入应用配置页面，在数据接入选项卡下点击数据API卡片右上角的【启用服务】按钮一键开启3个数据接入服务。点击【刷新状态】按钮，直到三个服务的状态都显示为【服务中】时，并且物料数据量和内容集数据量对应一致，表明内容资讯数据接入成功。

百度智能云 智能搜推引擎

应用管理

服务监控

推荐服务

配置应用

数据接入

服务调用

模型体验

数据API

启用服务

物料数据接入

服务中 已接入1000条数据

服务监控

物料管理

数据接入服务调用指南

更多接口使用说明, 请查阅《API接口文档》

第1步: 启用服务

点击右侧按钮一键开启3个数据接入服务。

第2步: 编写/获取代码

选择适合您的数据接入方式: 您可以使用CURL快速接入验证服务, 也可以选择相应语言复制代码展示的或者编写代码。

第3步: 替换模板中的参数

复制上方基本信息中圈选信息和数据API中的URL, 并参考代码示例进行替换。

第4步: 执行接入数据

返回"success:true"即表明接入成功, 跟踪数据状态, 接入必传数据即可

测试数据样例

字段名称	中文名	类型	是否必填	数据样例	备注
rid	物料唯一标识	string	是	NID_1001	一条物料的唯一标识, 物料id不能重复
title	物料标题	string	是	物料标题, 主要用来进行语义分析提取物料特征	标题
content	文章正文	string	是	示例文章正文	文章正文, 主要用来进行语义分析提取物料特征
publish_time	发布时间	long	是	1648561924764	物料发布时间, 可以用于最新物料召回的策略, 13位时间戳
ts	上传时间	long	是	1648561924764	物料上传时间, 格式: 13位时间戳
status	状态	int	是	1	物料上下架状态, 0 (下架) , 1 (上架) , 只有上架状态的物料可被推荐, 建议上架的物料占比不低于60%。

3点击查看内容，验证数据是否可用。按照3.1.2数据接入方法，接入用户数据和行为数据。

百度智能云 智能搜推引擎

应用管理

服务监控

推荐服务

配置应用

数据接入

服务调用

模型体验

用户数据接入

服务中 已接入0条数据

服务监控

数据接入方式:

请求地址

CURL

Python

Java

请求地址

http://18.137.19.146:8859/api/rec/ap/oml/ums/batch-sync

行为数据接入

服务中 已接入0条数据

服务监控

数据接入方式:

请求地址

CURL

Python

Java

请求地址

http://18.137.19.146:8859/api/rec/ap/oml/ums/batch-sync

管理内容集

概述

77

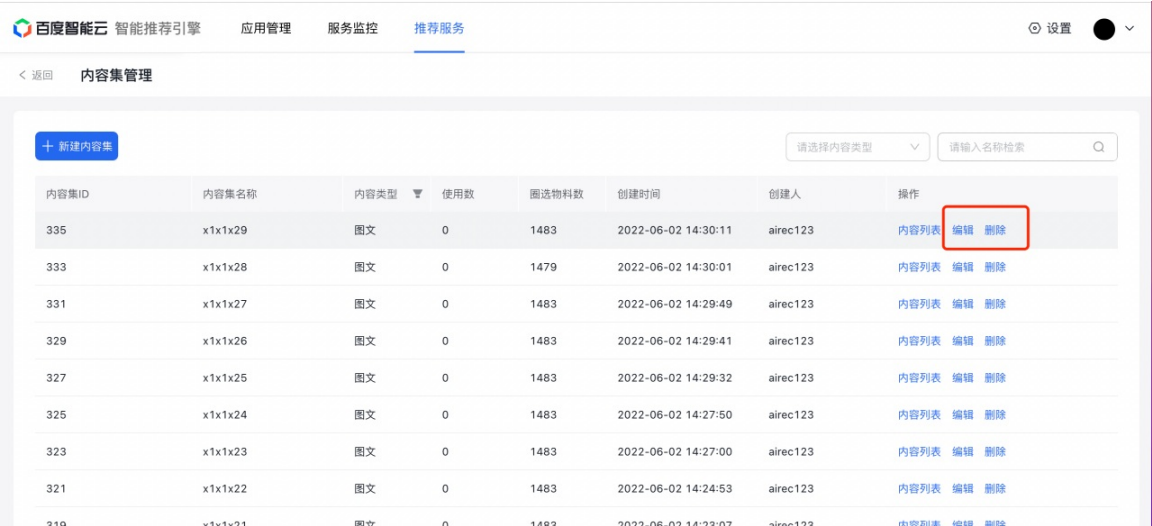
内容集管理涵盖了内容集从创建、使用、删除的全生命周期管理。

前提条件

1租户开通了内容服务，内容集的入口是开放可用的。 2内容集已经创建完成。

操作步骤

1登录进入智能推荐引擎，点击【推荐服务】--【内容资讯服务】，进入内容集列表页面，可以查询、编辑和删除内容集。



- (1) 查询内容集：用户在搜索框中输入查询内容集名称关键字，点击查询按钮即可。
- (2) 编辑内容集：仅支持编辑基本信息，不支持编辑圈选规则。
- (3) 删除内容集：如果内容集不存在后置依赖项（被应用引用），删除操作会删内容数据，请谨慎操作。误删后您可以重新建立内容集捞回；如果内容集存在后置依赖项，不允许删除，请删除后置依赖项后再执行此操作。
- (4) 内容管理：点击内容列表进入内容管理页面，对内容集内多条内容进行查询、上下架、删除等管理和维护，支持对内容的预览，方便运营人员操作。



- A、内容查询：用户在搜索框中输入查询内容集名称关键字，进行查询。
- B、内容上下架：内容下架后将不会在前台推荐。
- C、内容删除：该操作会删内容数据，请谨慎操作。
- D、内容详情：用户选择需要查看的内容，点击“详情”按钮，弹出详情信息框。

其他

多用户访问控制

一、介绍

多用户访问控制，主要用于帮助用户管理云账户下资源的访问权限，适用于企业内的不同角色，可以对不同的工作人员赋予使用产品的不同权限，当您的企业存在多用户协同操作资源时，推荐您使用多用户访问控制。 适用于下列使用场景：

- 中大型企业客户：对公司内多个员工授权管理；
- 中小开发者或小企业：添加项目成员或协作者，进行资源管理。

二、创建用户

1. 主账号用户登录后在控制台选择“多用户访问控制”进入用户管理页面。
2. 在左侧导航栏点击“用户管理”，在“子用户管理列表”页，点击“新建用户”。
3. 在弹出的“新建用户”对话框中，完成填写“用户名”和确认，返回“子用户管理列表”区可以查看到刚刚创建的子用户。

三、配置策略

AIREC支持系统策略，实现AIREC产品级权限控制。

- 系统策略：百度智能云系统为管理资源而预定义的权限集，这类策略可直接为子用户授权，用户只能使用而不能修改。
- 系统策略：系统策略包含管理权限和只读权限2种策略，权限范围详细如下：

策略名称	权限说明	权限范围
AIRECullControlPolicy	完全控制管理智能搜推引擎(airec)的权限	查看表、创建表、删除表，创建应用、删除应用、配置应用，创建场景、删除场景、配置场景
AIRECReadOnlyAccessPolicy	只读访问智能搜推引擎(airec)的权限	查看应用、查看表、查看场景

四、用户授权

在“用户管理->子用户管理列表页”的对应子用户的“操作”列选择“添加权限”，并为用户选择系统权限或自定义策略进行授权。 说明：如果在不修改已有策略规则的情况下修改某子用户的权限，只能通过删除已有的策略并添加新的策略来实现，不能取消勾选已经添加过的策略权限。

子用户登录



完成对子用户的授权后，可以将链接发送给子用户；子用户可以通过IAM用户登录链接登录主账号的管理控制台，根据被授权的策略对主账户资源进行操作和查看。

模型超参配置

一、模型配置（model_conf）

- 1、model: 指定了使用的模型类型。这里使用了built-in|KVFM，可能表示一个内置的键值对因子分解机（Key-Value Factorization Machines, KVFM）模型。注意，这里的命名可能不是标准的，因为常见的因子分解机模型包括FM（Factorization

Machines) 和 FFM (Field-aware Factorization Machines) , 而 KVFM 可能是一个特定实现或变种。

2、table_conf: 用于配置嵌入表 (embedding tables) 的参数, 这在处理分类特征时非常常见。每个嵌入表都有一个名称、维度、零索引列表、分片数以及初始化器。

```
* name: 嵌入表的名称。
* dim: 嵌入向量的维度。
* zero_indexes: 指定哪些索引在嵌入表中应该被初始化为零向量。这通常用于表示缺失值或特定的占位符。
* shard_num: 嵌入表的分片数, 用于分布式训练时分散嵌入表的计算。
* initializer: 嵌入向量的初始化方法, 这里使用了均匀随机初始化, 并指定了最小值和最大值。
```

二、编译配置 (compile_conf)

- 1、optimizer: 优化器的类型, 这里使用了Adam优化器。
- 2、lr: 学习率, 设置为 $1e-4$ 。
- 3、l2_decay: L2正则化项的系数, 用于防止过拟合。这里设置为1.0, 但通常这个值会远小于1, 可能需要调整。

三、运行配置 (run_conf)

- 1、runner: 指定了运行模式的类型, 这里使用了train_eval_save, 表示模型将进行训练、评估和保存。
- 2、epoch: 训练轮次。
- 3、train_print_period: 训练过程中打印日志的周期 (以样本数为单位) 。
- 4、train_debug: 是否在训练过程中启用调试模式。
- 5、valid_print_period: 验证过程中打印日志的周期 (以样本数为单位) 。
- 6、valid_debug: 是否在验证过程中启用调试模式。
- 7、save_dir: 模型保存目录。
- 8、save_mode: 模型保存模式, 这里使用了save_pipeline_model, 可能表示保存整个训练流程中的模型, 包括预处理和训练后的模型。
- 9、logdir: 日志文件保存的目录。

四、数据配置 (data_conf)

- 1、train_dataset 和 val_dataset: 分别配置了训练数据集和验证数据集的参数。

```
* mode: 数据集的处理模式, 这里都使用了InMemoryDataset, 表示数据将被加载到内存中。
* shuffle: 是否在训练前打乱数据。
* thread: 数据加载时使用的线程数。
* batch_size: 每个批次的数据量。
* dirs: 数据集所在的目录。注意这里训练集和验证集都指向了同一个目录 (./valid_dataset), 这在实际应用中可能不是最佳实践, 因为通常我们希望验证集是训练集的一个独立子集。
```

- 2、json_template: 指定了数据模板文件的路径 (sample_data.json) , 这个文件可能用于定义数据的格式和结构。

五、注意事项

- 1、配置文件中的某些参数 (如l2_decay的值) 可能需要根据实际情况进行调整。
- 2、训练集和验证集使用相同的目录可能不是最佳实践, 应该确保验证集是独立于训练集的。
- 3、嵌入表的维度和分片数应该根据问题的复杂性和可用的计算资源进行调整。
- 4、如果使用的是自定义模型 (如lr.py|MyLR) , 则需要确保相应的Python文件 (lr.py) 和类 (MyLR) 已经正确实现并可用。

六、模型参数

1. LR

```
fe_conf: slot_list.conf
model_conf:
  model: built-in|KVLr
  table_conf:
    - dim: 1
      initializer:
        attrs:
          max: 0.01
          min: -0.01
          type: uniform_random
        name: embedding
        shard_num: 20
        zero_indexes:
          - 0
  optimizer_conf:
    optimizer: adam
    lr: 1e-4
    l2_decay: 0.001

run_conf:
  runner: train_eval_save
  epoch: 1
  train_print_period: 100
  train_debug: false
  valid_print_period: 4
  valid_debug: false
  # load_dir: save_dir/epoch_0
  save_dir: save_dir
  save_mode: save_pipeline_model
  logdir: logs
  logfile: train.log

data_conf:
  train_dataset:
    mode: InMemoryDataset
    shuffle: True
    thread: 8
    batch_size: 128
    dirs:
      - /home/work/data/
  val_dataset:
    mode: InMemoryDataset
    shuffle: False
    thread: 8
    batch_size: 128
    dirs:
      - /home/work/data/
```

2. MMOE

```
fe_conf: slot_list.conf
model_conf:
  model: built-in|KVMMOE
table_conf:
  - name: tag_embedding
    dim: 16
    shard_num: 20
    zero_indexes: [0]
    initializer:
      type: uniform_random
      attrs:
        min: -0.01
        max: 0.01
  loss_weights: [1]
  task_modes: ["classification"]
  tower_sizes: [8]

optimizer_conf:
  optimizer: adam
  lr: 1e-4

run_conf:
  runner: train_eval_save
  epoch: 10
  train_print_period: 100
  train_debug: False
  valid_print_period: 100
  valid_debug: False
  load_dir: ""
  save_dir: "save_dir"

data_conf:
  json_template: json_template
  train_dataset:
    mode: InMemoryDataset
    shuffle: True
    thread: 1
    batch_size: 128
    dirs:
      - train-data
  val_dataset:
    mode: InMemoryDataset
    shuffle: False
    thread: 1
    batch_size: 128
    dirs:
      - train-data
```

3. FM

```
fe_conf: slot_list.conf
```

```
model_conf:
```

```
  model: built-in|KVFM
```

```
  table_conf: #remove table_conf if you don't use kv-table.
```

```
    - name: first_embedding
```

```
      dim: 1
```

```
      shard_num: 20
```

```
      zero_indexes: [0]
```

```
      initializer:
```

```
        type: uniform_random
```

```
        attrs:
```

```
          min: -0.01
```

```
          max: 0.01
```

```
    - name: second_embedding
```

```
      dim: 16
```

```
      shard_num: 20
```

```
      zero_indexes: [0]
```

```
      initializer:
```

```
        type: uniform_random
```

```
        attrs:
```

```
          min: -0.01
```

```
          max: 0.01
```

```
optimizer_conf:
```

```
  optimizer: adam
```

```
  lr: 1e-4
```

```
run_conf:
```

```
  runner: train_eval_save
```

```
  epoch: 10
```

```
  train_print_period: 100
```

```
  train_debug: False
```

```
  valid_print_period: 100
```

```
  valid_debug: False
```

```
  load_dir: "save_dir"
```

```
  save_dir: "save_dir"
```

```
data_conf:
```

```
  json_template: json_template
```

```
  train_dataset:
```

```
    mode: InMemoryDataset
```

```
    shuffle: True
```

```
    thread: 1
```

```
    batch_size: 128
```

```
    dirs:
```

```
      - /home/work/data/
```

```
  val_dataset:
```

```
    mode: InMemoryDataset
```

```
    shuffle: False
```

```
    thread: 1
```

```
    batch_size: 128
```

```
    dirs:
```

```
      - /home/work/data/
```

4. DssmRank

```
model_conf:
  model: dssm_rank|KvDssmRank
  left_hidden_size: [10]
  right_hidden_size: [10]
  table_conf:
    - dim: 64
      initializer:
        attrs:
          max: 0.01
          min: -0.01
          type: uniform_random
        name: embedding
      shard_num: 20
      zero_indexes:
        - 0

optimizer_conf:
  optimizer: adam
  lr: 1e-4
  l2_decay: 0.001

run_conf:
  runner: train_eval_save
  epoch: 1
  train_print_period: 100
  train_debug: false
  valid_print_period: 4
  valid_debug: false
  load_dir: save_dir/epoch_0
  save_dir: save_dir
  save_mode: save_pipeline_model
  logdir: logs
  logfile: train.log

data_conf:
  train_dataset:
    mode: InMemoryDataset
    shuffle: True
    thread: 8
    batch_size: 128
    dirs:
      - train-data
  val_dataset:
    mode: InMemoryDataset
    shuffle: False
    thread: 1
    batch_size: 128
    dirs:
      - train-data
```

5. DeepFM


```
e_conf: slot_list.conf

model_conf:
  model: built-in|KVDeepFM
  table_conf: #remove table_conf if you don't use kv-table.
  - name: embedding_1
    dim: 100
    zero_indexes: [0]
    initializer:
      type: uniform_random
      attrs:
        min: -0.01
        max: 0.01
  - name: embedding_2
    dim: 100
    zero_indexes: [0]
    initializer:
      type: uniform_random
      attrs:
        min: -0.01
        max: 0.01
  hidden_size: [256, 128, 64]

optimizer_conf:
  optimizer: adam
  lr: 1e-4

run_conf:
  runner: train_eval_save
  epoch: 10
  train_print_period: 100
  train_debug: False
  valid_print_period: 100
  valid_debug: False
  load_dir: "save_dir"
  save_dir: "save_dir"

data_conf:
  json_template: json_template
  train_dataset:
    mode: InMemoryDataset
    shuffle: True
    thread: 1
    batch_size: 128
    dirs:
      - /home/work/data/
  val_dataset:
    mode: InMemoryDataset
    shuffle: False
    thread: 1
    batch_size: 128
    dirs:
      - /home/work/data/
```

6. WideDeep

```
fe_conf: slot_list.conf

model_conf:
  model: built-in|KVNRRMS
  table_conf: #remove table_conf if you don't use kv-table.
    - name: title_embedding
      #load_dir: "kv_emb_all"
      dim: 300
      zero_indexes: [0]
      initializer:
        type: uniform_random
        attrs:
          min: -0.01
          max: 0.01

  n_heads: 20
  head_dim: 20
  atten_hidden_dim: 200
  hist_len: 5
  title_len: 3
  cand_len: 2
  #place: cuda

optimizer_conf:
  optimizer: adam
  lr: 1e-4
  # l2_decay: 0.001 nrms关闭l2_decay

run_conf:
  runner: train_eval_save
  epoch: 1
  train_print_period: 1
  train_debug: False
  valid_print_period: 1
  valid_debug: False
  logdir: "expr/log_nrms"
  load_dir: ""
  save_dir: "save_dir"
  save_mode: "save_pipeline_model"

data_conf:
  train_dataset:
    mode: InMemoryDataset
    shuffle: True
    thread: 8
    batch_size: 128
    dirs:
      - train-data
  val_dataset:
    mode: InMemoryDataset
    shuffle: False
    thread: 1
    batch_size: 128
    dirs:
      - train-data
```

7. NRMS

```
fe_conf: slot_list.conf

model_conf:
  model: built-in|KVNRRMS
  table_conf: #remove table_conf if you don't use kv-table.
    - name: title_embedding
      #load_dir: "kv_emb_all"
      dim: 300
      zero_indexes: [0]
      initializer:
        type: uniform_random
        attrs:
          min: -0.01
          max: 0.01

  n_heads: 20
  head_dim: 20
  atten_hidden_dim: 200
  hist_len: 5
  title_len: 3
  cand_len: 2
  #place: cuda

optimizer_conf:
  optimizer: adam
  lr: 1e-4
  # l2_decay: 0.001 nrms关闭l2_decay

run_conf:
  runner: train_eval_save
  epoch: 1
  train_print_period: 1
  train_debug: False
  valid_print_period: 1
  valid_debug: False
  logdir: "expr/log_nrms"
  load_dir: ""
  save_dir: "save_dir"
  save_mode: "save_pipeline_model"

data_conf:
  train_dataset:
    mode: InMemoryDataset
    shuffle: True
    thread: 8
    batch_size: 128
    dirs:
      - train-data
  val_dataset:
    mode: InMemoryDataset
    shuffle: False
    thread: 1
    batch_size: 128
    dirs:
      - train-data
```

8. Dssm

```
fe_conf: slot_list.conf
model_conf:
  model: dssm_rank|KvDssmRank
  left_hidden_size: [10]
  right_hidden_size: [10]
  table_conf:
    - dim: 64
      initializer:
        attrs:
          max: 0.01
          min: -0.01
          type: uniform_random
        name: embedding
      shard_num: 20
      zero_indexes:
        - 0

optimizer_conf:
  optimizer: adam
  lr: 1e-4
  l2_decay: 0.001

run_conf:
  runner: train_eval_save
  epoch: 1
  train_print_period: 100
  train_debug: false
  valid_print_period: 4
  valid_debug: false
  load_dir: save_dir/epoch_0
  save_dir: save_dir
  save_mode: save_pipeline_model
  logdir: logs
  logfile: train.log

data_conf:
  train_dataset:
    mode: InMemoryDataset
    shuffle: True
    thread: 8
    batch_size: 128
    dirs:
      - train-data
  val_dataset:
    mode: InMemoryDataset
    shuffle: False
    thread: 1
    batch_size: 128
    dirs:
      - train-data
```

八、训练日志

```
[test@dev-baidu-01 logs]$ cat train.log
2024-01-05 17:44:27,762-INFO: local: set random seed as 1000
2024-01-05 17:44:27,763-INFO: local: vdl records will be write in vdl_local
2024-01-05 17:44:27,864-INFO: local: running on host: ae84503c2387, ip: 192.168.42.5, dir:
/home/work/iptv/mmoe/model_5000w
2024-01-05 17:44:27,865-INFO: local: init running config done
2024-01-05 17:44:28,478-INFO: local: Loading checkpoint with only incremental table: run_conf.load_dir is empty
2024-01-05 19:15:46,795-INFO: local: init dataset with InMemoryDataset, data size: 53466444, load in memory using:
5437.415941, shuffle using: 40.845384
2024-01-05 19:15:56,411-INFO: local: init dataset with InMemoryDataset, data size: 50000, load in memory using:
9.547343, shuffle using: 0.060682
2024-01-05 19:15:56,411-INFO: local: init model and reader done and start running epochs
2024-01-05 19:37:43,719-INFO: local: {"time": "2024-01-05 19:37:43", "run_type": "train", "batch_id": 200000, task
0 auc[1]:[0.749406], task 1 auc[1]:[0.944328]}
2024-01-05 20:04:17,671-INFO: local: {"time": "2024-01-05 20:04:17", "run_type": "train", "batch_id": 400000, task
0 auc[1]:[0.763899], task 1 auc[1]:[0.964906]}
2024-01-05 20:25:09,876-INFO: local: {"time": "2024-01-05 20:25:09", "run_type": "train", "batch_id": 600000, task
0 auc[1]:[0.772776], task 1 auc[1]:[0.973489]}
2024-01-05 20:46:00,475-INFO: local: {"time": "2024-01-05 20:46:00", "run_type": "train", "batch_id": 800000, task
0 auc[1]:[0.779014], task 1 auc[1]:[0.977674]}
2024-01-05 20:49:43,556-INFO: local: finish training epoch 0 in 5627.152987s, metrics: {"task 0 auc":0.779906,"task
1 auc":0.978272}
2024-01-05 20:49:43,567-INFO: local: Saving checkpoint with save_dir/epoch_0
2024-01-05 20:49:43,578-INFO: local: save serving kv model
2024-01-05 20:49:43,589-INFO: local: Dumped tag_embedding, total feature number 865861, cost time 0.032322
2024-01-05 20:49:43,806-INFO: local: Dumped tag_embedding, total feature number 865861, cost time 0.05661
2024-01-05 20:49:43,906-INFO: local: finish saving epoch 0 save model at: save_dir/epoch_0
2024-01-05 20:49:44,207-INFO: local: {"time": "2024-01-05 20:49:44", "run_type": "valid", "batch_id": 10000, task 0
auc[1]:[0.783166], task 1 auc[1]:[0]}
2024-01-05 20:49:44,408-INFO: local: {"time": "2024-01-05 20:49:44", "run_type": "valid", "batch_id": 20000, task 0
auc[1]:[0.781714], task 1 auc[1]:[1]}
2024-01-05 20:49:44,709-INFO: local: {"time": "2024-01-05 20:49:44", "run_type": "valid", "batch_id": 30000, task 0
auc[1]:[0.786361], task 1 auc[1]:[1]}
2024-01-05 20:49:44,910-INFO: local: {"time": "2024-01-05 20:49:44", "run_type": "valid", "batch_id": 40000, task 0
auc[1]:[0.786127], task 1 auc[1]:[0.999788]}
2024-01-05 20:49:45,111-INFO: local: {"time": "2024-01-05 20:49:45", "run_type": "valid", "batch_id": 50000, task 0
auc[1]:[0.784533], task 1 auc[1]:[0.999859]}
2024-01-05 20:49:45,311-INFO: local: {"time": "2024-01-05 20:49:45", "run_type": "valid", "batch_id": 60000, task 0
auc[1]:[0.780083], task 1 auc[1]:[0.999906]}
2024-01-05 20:49:45,512-INFO: local: {"time": "2024-01-05 20:49:45", "run_type": "valid", "batch_id": 70000, task 0
auc[1]:[0.779144], task 1 auc[1]:[0.999907]}
2024-01-05 20:49:46,308-INFO: local: finish evaluating epoch 0 in 1.805703s, metrics: {"task 0 auc":0.779955,"task
1 auc":0.999904}
2024-01-05 20:49:46,309-INFO: local: epoch: 0 start_at: Fri Jan 5 19:15:56 2024 end_at: Fri Jan 5 20:49:45 2024
2024-01-05 20:50:25,716-INFO: local: all running finished
```

这个日志文件 (train.log) 记录了一个机器学习或深度学习模型训练过程中的详细信息。以下是对日志中关键信息的解析和概述：

1. 初始化设置：

- 日志开始时，设置了随机种子 (set random seed as 1000)，这有助于实验的复现性。
- 指定了VisualDL（一个可视化工具）的记录路径 (vdl_local)。
- 记录了运行的主机信息 (host: ae84503c2387, ip: 192.168.42.5) 和模型运行的目录 (/home/work/iptv/mmoe/model_5000w)。
- 初始化运行配置完成 (init running config done)。

2. 数据集加载：

- 加载了两个数据集，一个大小为53466444（可能是训练集），另一个大小为50000（可能是验证集或测试集）。这两个数据集都被加载到内存中，并记录了加载和打乱数据所需的时间。

3. 模型训练：

- 初始化模型和读取器后，开始运行训练周期（epochs）。
- 记录了每200000个批次（batch_id）的AUC（Area Under the Curve，一种常用的分类评估指标）值，分别针对两个不同的任务（task 0和task 1）。
- 完成了第一个训练周期（epoch 0），并记录了总训练时间（5627.152987s）。

4. 模型保存和评估：

- 训练完成后，保存了第一个周期的模型到save_dir/epoch_0。
- 保存了服务所需的KV模型。
- 记录了特征嵌入（tag_embedding）的保存过程，包括特征数量和所需时间。
- 进行了模型评估，记录了不同批次在验证集上的AUC值，并计算了整体的AUC值。

5. 运行结束：

- 记录了训练周期的开始和结束时间。
- 指出所有运行已完成（all running finished）。

API文档

数据接入接口

行为同步接口

🔗 行为同步接口

接口说明

百度提供的交互行为数据批量同步接口，适用于在线实时调用，回传用户行为。

请求示例

HTTP方法：POST

请求URL：https://airec.baidu.com/airec/api/omni/ms/batch-sync

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	数据表鉴权信息
Alias-Name	string	是	数据表名称
code	string	是	数据表类型
id	string	是	数据表id

请求格式

POST 方式调用

注意：要求使用由 JSON 组成的数组格式的结构体来描述一个请求的具体内容，最大支持20条。

请求参数

图文行为：

参数名	中文名	类型	是否必填	备注
nid	物料唯一标识nid	string	是	物料唯一标志，当nid非必需时，推荐传入空字符串""
uid	用户UID	string	是	用户唯一标志
event	用户行为	string	是	行为类型：show展现、click点击、collect收藏、follow关注、like喜欢、dislike不喜欢、search搜索、share分享、comment评论等
ts	行为发生时间	long	是	行为发生时间，毫秒，格式: 13位时间戳，如1648561924764
traceid	推荐traceid	string	是	推荐请求唯一标志, 来自百度推荐接口的traceid;如果是非百度推荐的行为数据则传selftrace
query	搜索query	string	否	用户的搜索query
scene_type	行为场景类型	string	否	用户行为场景，p_rec:个性化推荐场景、r_rec:相关推荐场景、h_rec:热门推荐场景、search:搜索场景
event_scene	行为的场景标识	string	否	行为的场景，可以传入客户自定义的场景标识，如首页，详情页等场景id
event_value	行为信息	string	否	用户行为信息。支持将nid、tags、author_id加入黑名单，仅当event值为dislike时，该配置才会生效，示例传参：'dislike_class:tags=科技'

商品行为：

参数名	中文名	类型	是否必填	备注
nid	物料唯一标识nid	string	是	物料唯一标志，当nid非必需时，推荐传入空字符串""
uid	用户UID	string	是	用户唯一标志
event	用户行为	string	是	行为类型：show展现、click点击、collect收藏、follow关注、like喜欢、dislike不喜欢、search搜索、share分享、comment评论等
ts	行为发生时间	long	是	行为发生时间，毫秒，格式: 13位时间戳，如1648561924764
traceid	推荐traceid	string	是	推荐请求唯一标志, 来自百度推荐接口的traceid;如果是非百度推荐的行为数据则传selftrace
query	搜索query	string	否	用户的搜索query
scene_type	行为场景类型	string	否	用户行为场景，p_rec:个性化推荐场景、r_rec:相关推荐场景、h_rec:热门推荐场景、search:搜索场景
event_scene	行为的场景标识	string	否	行为的场景，可以传入客户自定义的场景标识，如首页，详情页等场景id
event_value	行为信息	string	否	用户行为信息。支持将nid、tags、author_id加入黑名单，仅当event值为dislike时，该配置才会生效，示例传参：'dislike_class:tags=科技'

媒资行为：

参数名	中文名	类型	是否必填	备注
nid	物料唯一标识nid	string	是	物料唯一标志，当nid非必需时，推荐传入空字符串""
uid	用户UID	string	是	用户唯一标志
event	用户行为	string	是	行为数据类型, 购买buy 搜索点击search_click 曝光rec_show 推荐点击rec_click 直播tv_playing 点播vod_playing回看tvod_playing 时移timeshift 不喜欢dislike等
ts	行为发生时间	long	是	行为发生时间, 毫秒, 格式: 13位时间戳, 如1648561924764
traceid	推荐trace id	string	是	推荐请求唯一标志, 来自百度推荐接口的traceid;如果是非百度推荐的行为数据则传selftrace
context_id	详情页媒资id	string	否	详情页媒资id
event_value	行为信息	string	否	用户行为信息。支持将nid、tags、author_id加入黑名单, 仅当event值为dislike时, 该配置才会生效, 示例传参: 'dislike_class:tags=科技'
scene_type	行为场景类型	string	否	用户行为场景, p_rec:个性化推荐场景、r_rec:相关推荐场景、h_rec:热门推荐场景、search:搜索场景
base_tags	媒资标签	strings	否	内容标签,支持多个值
actors	演员列表	strings	否	演员列表, 支持多个值
directors	创作者	strings	否	媒资创作者/导演/出品方id,支持多个值
channels	频道	strings	否	用户自定义标签, 可在推荐接口作为推荐条件进行数据过滤
episode_type	媒资类型	string	否	物料类型, program单剧集、series连续剧/系列剧、episode子集、channel频道、schedule节目单、album专栏、subject专题
type	类型标识	string	是	包含电视剧、电影、综艺、儿童动漫、新闻、音乐、体育、纪录片、游戏、直播、短内容合辑、戏曲、公开课, 其他类型填未知, 用作特征

请求示例

```
{
  "uid": "uid_1001",
  "nid": "nid_1001",
  "event": "dislike",
  "ts": 1609768813182,
  "traceid": "traceid_1001",
  "query": "王维",
  "scene_type": "0",
  "event_scene": "scene_1",
  "event_value": "dislike_class:tags=科技"
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
success	boolean	是	成功标识
code	string	是	响应码：0正常，其他异常
message	string	是	响应描述
result	dict	是	响应数据

返回示例

```
{
  "success": true,
  "code": "0",
  "message": null,
  "result": null
}
```

行为数据traceid获取方法

- (1) 历史行为数据

默认传selftrace

- (2) 新行为数据

- 调用百度推荐接口的场景

系统集成关系



推荐系统会为每一次推荐请求结果分配唯一id，用户在调取个性化推荐或者相关推荐接口时请保存此id，将推荐给用户的物料反馈行为回传给推荐系统。

```
using time: 0.211582, status 200, ret content: {"result":{"traceid":"924c678a-aca8-495d-9d58-a7d7c8158fb5","size":30,"item_list":[{"youku_XnTg5MjU1NjU1Mg==","youku_XnTg5MjU1NjU1O0A==","youku_XnTg5MTIyNTYlNA==","youku_XnTg_YInTzXZmG==","youku_XnTg4Nzc0ODczMg==","youku_XnTg4NzcxNDI3Ng==","youku_XnTg4Nzc0OTQyOA==","youku_XnTg4NjU0Mjg4NA==","youku_XnTgzMg4NjUwZmE2MA==","youku_XnTg4NDg4NDIzM3Mg==","8272206250600000","6142206250600000","8272206250555000","6142206250555000","8272206250544000","6142206250544000","6142206250544000","6142206250544000","8272206250544000","6142206242344000","8272206242344000","6142206242254000","6142206242254000","6192206242225000","8272206242206000","6142206242206000","8272206242200000","6142206242200000","8272206242145000","6142206242145000","6142206242145000","8272206242100000"},"time":"","trace_id":"924c678a-aca8-495d-9d58-a7d7c8158fb5","code":0,"message":"bottom"}]}
```

- 非调用百度推荐接口的场景 默认传selftrace

用户同步接口

🌀 用户同步接口

接口说明

百度提供的用户信息同步接口,可以实现用户库、正排库批量导入。

请求示例

HTTP方法：POST

请求URL：<https://airec.baidu.com/airec/api/omni/ms/batch-sync>

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	数据表鉴权信息
Alias-Name	string	是	数据表名称
code	string	是	数据表类型
id	string	是	数据表id

请求格式

POST 方式调用

注意：要求使用由 JSON 组成的数组格式的结构体来描述一个请求的具体内容，最大支持20条。

请求参数

图文用户：

参数名	中文名	类型	是否必填	备注
uid	用户UID	string	是	用户唯一标志
uid_type	用户注册类型	string	否	用户注册类型:1 app注册类型; 2 手机号; 3 微信号; 4 其它
register_time	用户注册时间	long	否	用户注册时间, 格式: 13位时间戳
tags	兴趣标签	strings	否	兴趣标签
subscribe_author_ids	订阅作者	strings	否	用户订阅的作者id

商品用户：

参数名	中文名	类型	是否必填	备注
uid	用户UID	string	是	用户唯一标志
uid_type	用户注册类型	string	否	用户注册类型(根据实际注册场景传入即可，无特殊含义，仅作为用户注册标记): 1 app注册类型; 2 手机号; 3 微信号; 4 其它
register_time	用户注册时间	long	否	用户注册时间, 格式: 13位时间戳
tags	兴趣标签	strings	否	兴趣标签
age	用户年龄	string	否	用户年龄
gender	用户性别	string	否	用户性别：男性0、女性1、未知2转
subscribe_shop_ids	订阅店铺	strings	否	用户订阅的店铺id

媒资用户

参数名	中文名	类型	是否必填	备注
uid	用户UID	string	是	用户唯一标志
area	地区	string	否	用户所在地区
uid_type	用户注册类型	string	否	用户注册类型:1 app注册类型; 2 手机号; 3 微信号; 4 其它
register_time	用户注册时间	long	否	用户注册时间, 格式: 13位时间戳
base_tags	兴趣标签	strings	否	兴趣标签
subscribe_director_ids	订阅的创作者	strings	否	用户订阅的媒资创作者/导演/出品方id
collection_list	收藏列表	strings	否	收藏列表，例如["NID_1001", "NID_1002"]
watch_list	观看列表	strings	否	观看列表，例如["NID_1001", "NID_1002"]

请求示例

```
{
  "uid": "uid_1001",
  "uid_type": "1",
  "register_time": 1648561924764,
  "tags": ["标签1", "标签2"]
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
success	boolean	是	成功标识
code	string	是	响应码：0正常，其他异常
message	string	是	响应描述
result	dict	是	响应数据

返回示例

```
{
  "success": true,
  "code": "0",
  "message": null,
  "result": null
}
```

用户数据详情接口

 用户数据详情接口

接口说明

百度提供查询用户数据详情的接口。

请求示例

HTTP方法：POST

请求URL：https://airec.baidu.com/airec/api/omni/ms/detail/search

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	数据表鉴权信息
Alias-Name	string	是	数据表名称
code	string	是	数据表类型
id	string	是	数据表id

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	类型	是否必填	说明
id	string	是	物料数据的nid

请求示例

```
{
  "id": "nid_1"
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
success	boolean	是	成功标识
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
result	object	是	响应结果，数据详情

返回示例

```
{
  "success": true,
  "code": "0",
  "message": "success",
  "result": {
    "uid": "UID_1001",
    "uid_type": "1",
    "register_time": "2022-03-11 01:32:13",
    "tags": ["标签1", "标签2"],
    "hist_click": ["nid01", "nid02"],
    "hist_tags": ["nba", "cba"],
    "hist_baidu_labels": ["梅西", "iphone"]
  }
}
```

物料同步接口

🔗 物料同步接口

接口说明

百度提供的物料内容同步接口，用于批量同步物料库数据。

请求示例

HTTP方法：**POST**

请求URL：**https://airec.baidu.com/airec/api/omni/ms/batch-sync**

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	数据表鉴权信息
Alias-Name	string	是	数据表名称
code	string	是	数据表类型
id	string	是	数据表id

请求格式

POST 方式调用

注意：要求使用由 JSON 组成的数组格式的结构体来描述一个请求的具体内容，最大支持20条。

请求参数

图文物料：

参数名	中文名	类型	是否必填	备注
nid	物料唯一标识	string	是	一条物料的唯一标识，物料id不能重复
site_name	站点名称	string	否	站点名称
title	物料标题	string	是	标题
content	文章正文	string	是	文章正文，主要用来进行语义分析提取物料特征
publish_time	发布时间	long	是	物料发布时间，可以用于最新物料召回的策略，13位时间戳：示例 1648561924764
ts	上传时间	long	是	物料上传时间, 格式: 13位时间戳，示例 1648561924764
status	状态	int	是	物料上下架状态， 0（下架）、1（上架），只有上架状态的物料可被推荐，建议上架状态的物料占比不低于60%。
author_id	作者ID	string	否	当前图文物料的作者id，暂不支持多个
author	作者昵称	string	否	当前图文物料的作者名称，暂不支持多个
channels	频道	strings	否	频道列表，数组格式，["体育", "美食"]
tags	自有标签	strings	否	业务方运营的标签，主要用作标签召回，支持多个值
category	类别	string	否	文章类别，用于文章分类和筛选。
sub_category	子类别	string	否	子类别，用于文章存在级联子分类和筛选。
url	图文地址	string	否	图文地址
cities	城市名称或编码列表	strings	否	所属城市列表：["beijing", "shanghai"]
content_type	物料类型	string	否	图文为article、视频为video，不传默认为article
summary	物料摘要	strings	否	物料摘要
search_hint	搜索提示词	string	否	搜索提示词，["西红柿炒鸡蛋怎么做", "西红柿炒鸡蛋好吃吗"]

商品物料：

参数名	中文名	类型	是否必填	备注
nid	商品唯一标识	string	是	一条商品的唯一标识，商品id不能重复
title	商品标题	string	是	商品标题，主要用来进行语义分析提取物料特征
content	商品描述	string	否	商品描述，主要用来进行语义分析提取商品特征
publish_time	商品上架时间	long	是	商品上架时间，可以用于最新商品召回的策略，13位时间戳
ts	上传时间	long	是	商品上传时间, 格式: 13位时间戳
status	商品状态	int	是	商品上下架状态， 0（下架）、1（上架），只有上架状态的商品可被推荐，建议上架状态的商品占比不低于60%
shop_id	店铺ID	string	否	当前商品物料的店铺id，暂不支持多个
brand	商品品牌	string	否	商品品牌
tags	商品自有标签	strings	否	商品标签，如首页展示标签“特惠”，“热卖”,或详情页“满50包邮”,“假一赔三”等
category	类别	string	否	商品类别，用于商品分类和筛选。
sub_category	子类别	string	否	子类别，用于商品存在级联子分类和筛选。
source	商品来源类型	string	否	商品来源类型，如自营/第三方
price	商品价格	float	否	商品价格
rating	商品评分	string	否	商品的评分
cities	城市名称或编码列表	strings	否	物料所属的城市列表，如["beijing", "shanghai"]

媒资物料：

参数名	中文名	类型	是否必填	备注
nid	物料唯一标识	string	是	一条物料的唯一标识，物料id不能重复
title	媒资标题	string	是	电影、剧集、节目等标题，主要用来进行语义分析提取物料特征
display_run_time	媒资时长	int	否	播放时长(秒)
content	商品描述	string	否	商品描述，主要用来进行语义分析提取商品特征
publish_time	商品上架时间	long	是	商品上架时间，可以用于最新商品召回的策略，13位时间戳
ts	上传时间	long	是	商品上传时间, 格式: 13位时间戳
status	商品状态	int	是	商品上下架状态， 0（下架）、1（上架），只有上架状态的商品可被推荐，建议上架状态的商品占比不低于60%
shop_id	店铺ID	string	否	当前商品物料的店铺id，暂不支持多个
brand	商品品牌	string	否	商品品牌
tags	商品自有标签	strings	否	商品标签，如首页展示标签“特惠”，“热卖”,或详情页“满50包邮”,“假一赔三”等
category	类别	string	否	商品类别，用于商品分类和筛选。
sub_category	子类别	string	否	子类别，用于商品存在级联子分类和筛选。

source	商品来源类型	string	否	商品来源类型，如自营/第三方
price	商品价格	float	否	商品价格
rating	商品评分	string	否	商品的评分
cities	城市名称或编码列表	strings	否	物料所属的城市列表，如["beijing", "shanghai"]
country	国家地区	string	否	发布国家
year	上映年份	string	否	上映年份，用作特征，如生成某年的排行榜。
desc	媒资描述	string	否	视频描述，主要用来进行语义分析提取物料特征
actors	演员列表	strings	否	演员列表，支持多个值
directors	创作者	strings	否	媒资创作者/导演/出品方id,支持多个值
view_point	看点	string	否	内容看点
score	评分	string	否	评分
data_source	物料来源(电信/移动/联通/其它)	string	是	运营商或运营商编码: 电信/移动/联调/其它
epi_type	媒资类型	string	否	物料类型，program单剧集、series连续剧/系列剧、episode子集、channel频道、schedule节目单、album专栏、subject专题
content_type	内容类型	string	否	指定内容是视频还是图文类型
base_tags	媒资标签	strings	否	内容标签,支持多个值
op_tags	运营标签	strings	否	业务方运营的标签
show_tags	展示标签	strings	否	展示标签
valid_time	生效时间	long	否	物料生效时间, 用于控制物料可被推荐的生效时间，13位时间戳
expire_time	过期时间	long	否	物料失效时间，用于控制物料过期后不被推荐的场景，13位时间戳
publish_time	发布时间	long	是	发布时间, 格式: 13位时间戳
operator_id	运营商ID	string	否	运营商(电信、联通、移动等)
operator_name	运营商名称	string	否	运营商(电信、联通、移动等)
provider_id	供应商ID, 直播情况下为直播的频道ID	string	否	供应方ID,媒资提供商或者来源,用作特征
provider_name	供应商, 直播情况下为直播的频道名称	string	否	供应方名称,媒资提供商或者来源,用作特征
need_pay	是否付费	int	否	是否付费，0（不付费）、1（付费）
channels	频道	strings	否	用户自定义标签，可在推荐接口作为推荐条件进行数据过滤
ts	上传时间	long	是	物料上传时间， 格式: 13位时间戳
url	视频地址	string	否	视频地址
language	影视语言	string	否	媒资所用语言，用作特征

award	获奖	string	否	用作特征
definition	清晰度	string	否	用作特征
cities	城市名称或编码列表	strings	否	物料所属的城市列表
live_tags	直播时间戳标签列表	strings	否	直播标签
media_code	媒资编码	string	否	媒资编码
base_tag_codes	媒资编码	strings	否	媒资编码
compere	主持人	strings	否	主持人
type	类型标识	string	是	包含电视剧、电影、综艺、儿童动漫、新闻、音乐、体育、纪录片、游戏、直播、短内容合辑、戏曲、公开课，其他类型填未知，用作特征
type_code	类型标识编码	string	否	类型标识编码
heat	热度	float	否	热度
is_subset	是否是子集（如花千骨第二集）	int	否	是否是子集（如花千骨第二集），0（不是）、1（是）
series_type	系列剧类型	string	否	连续剧/系列片
vod_type	剧集类型	string	否	vod：单剧集 episode：连续剧子集 series2：系列剧子集
biz_domain	业务类型	string	否	0：IPTV 业务(默认) 2：WEBTV 业务 3：混合 IPTV 和 WEBTV业务
poster	竖图文件路径	string	否	海报，对应内容的竖图文件在图片服务器中保存路径
still	横图文件路径	string	否	海报，对应内容的横图文件在图片服务器中保存路径
opimg1	运营图1文件路径	string	否	/op/op1.img 运营图1文件在图片服务器中保存路径
opimg2	运营图2文件路径	string	否	/op/op2.img 运营图2文件在图片服务器中保存路径
base_mark	基础角标标识	string	否	基础角标标识，如/base/mark.img
op_mark	运营角标标识	string	否	运营角标标识，如/op/mark.img
ext1	string类型拓展字段1	string	否	string类型拓展字段1
ext2	string类型拓展字段2	string	否	string类型拓展字段2
ext3	string类型拓展字段3	string	否	string类型拓展字段3
ext_list1	string list类型拓展字段1	strings	否	string list类型拓展字段1
ext_list2	string list类型拓展字段2	strings	否	string list类型拓展字段2
ext_list3	string list类型拓展字段3	strings	否	string list类型拓展字段3
status	物料上下架状态	int	是	物料状态：0下架，1上架

请求示例

```
{
  "nid": "nid_1",
  "site_name": "mr.yesterday",
  "title": "杂诗·王维 1",
  "content": "双燕初命子，五桃新作花，来日绮窗前，寒梅著花未。",
  "author_id": "author_123456_id_1",
  "publish_time": 1648561924764,
  "status": 1,
  "ts": 1609768813182
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
success	boolean	是	成功标识
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
result	dict	是	响应数据

返回示例

```
{
  "success": true,
  "code": "0",
  "message": null,
  "result": null
}
```

物料简单查询接口

🔗 物料简单查询接口

接口说明

百度提供的更加自由的方式查询。

请求示例

HTTP方法：POST

请求URL：https://airec.baidu.com/airec/api/omni/ms/general/search

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	数据表鉴权信息
Alias-Name	string	是	数据表名称
code	string	是	数据表类型
id	string	是	数据表id

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	类型	是否必填	说明
pageNo	int	否	当前页码，默认1，最小1
pageSize	int	否	每页条数，默认20,最大100
retype	boolean	否	是否做retype操作,默认false,为true需开启etl（将所有返回字段值转string类型）
totalCount	boolean	否	为空则返回总数和数据，true仅返回总数，false仅返回数据
conditions	array	否	所查询的自定义查询条件
field	string	是	查询的字段名
fieldType	string	否	字段类型，string，number，list_s，list_n，boolean
matchType	string	是	匹配类型，is精准查询、in包含查询、gte/lte范围查询（大于等于/小于等于）、match匹配查询、not match非匹配查询、like模糊查询、null为空、not null不为空
value	object	是	要匹配的值，注意传输时是字符类型还是数值类型，in、match、not match为数组
resultFields	array	否	返回字段列表,为空时返回所有
pageSort	object	否	分页排序参数,为空时采用cms默认配置
direction	string	是	分页查询排序：ASC、DESC
fields	array	是	排序字段列表

请求示例

```
{
  "pageNo":1,
  "pageSize":20,
  "conditions":[{"field":"online",
    "fieldType": "number",
    "matchType":"is",
    "value": 1
  },{
    "field":"publish_time",
    "matchType":"gte",
    "value": 1609813338000
  },{
    "field":"nid",
    "fieldType":"list_s",
    "matchType":"in",
    "value": "vLEStZZ5jMvAlnUy2R2eBfrX2AeB28RI,vLEStZZ5jMvAlnUy2R2eBfrX2AeB28R2"
  }],
  "resultFields":["nid","title","publish_time"],
  "pageSort":{"direction": "DESC",
    "fields":["publish_time"]}
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
success	boolean	是	成功标识
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
page	object	是	响应分页结果
pageNo	int	是	当前页码
pageSize	int	是	每页条数
totalPages	int	是	总页码
totalCount	int	是	总条数
result	array	否	分页数据
其他	object	是	resultFields参数所指定的字段，该字段为空时返回所有字段

返回示例

```
{
  "success": true,
  "code": "0",
  "message": "success",
  "page": {
    "pageNo": 1,
    "pageSize": 20,
    "totalPages": 1,
    "totalCount": 2,
    "result": [
      {
        "nid": "vLEStZZ5jMvAlnUy2R2eBfrX2AeB28RI",
        "title": "还有比韩庚版关羽更辣眼睛的关羽吗？",
        "publish_time": 1609768813182
      },
      {
        "nid": "vLEStZZ5jMvAlnUy2R2eBfrX2AeB28R2",
        "title": "还有比韩庚版关羽更辣眼睛的关羽吗？",
        "publish_time": 1609768813182
      }
    ]
  }
}
```

物料数据详情接口

🔗 物料数据详情接口

接口说明

百度提供查询数据详情的接口。

请求示例

HTTP方法：POST

请求URL：https://airec.baidu.com/airec/api/omni/ms/detail/search

Path参数如下：

参数名	类型	是否必填	说明
-----	----	------	----

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	数据表鉴权信息
Alias-Name	string	是	数据表名称
code	string	是	数据表类型
id	string	是	数据表id

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	类型	是否必填	说明
id	string	是	数据的id

请求示例

```
{
  "id": "nid_1"
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
success	boolean	是	成功标识
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
result	object	是	响应结果，数据详情

返回示例

```
{
  "success": true,
  "code": "0",
  "message": "success",
  "result": {
    "nid": "nid_1",
    "site_name": "mr.yesterday",
    "title": "杂诗-王维-1",
    "content": "双燕初命子，五桃新作花，来日绮窗前，寒梅著花未。",
    "mthid": "device_123456_id_1",
    "publish_time": 1609768813182,
    "status": 1,
    "ts": 1609768813182,
    "baidu_labels": ["杂诗", "王维"]
  }
}
```

物料条件批量调权接口

 物料条件批量调权接口

接口说明

百度提供的基于一定条件调整物料权重的接口。

请求示例

HTTP方法：`POST`

请求URL：`http://airec.baidu.com/airec/api/weight/update`

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	数据表鉴权信息
Alias-Name	string	是	数据表名称

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	中文名	类型	是否必填	备注
conditions	筛选条件	list	是	筛选条件列表，最大支持20条
type	且/或逻辑	string	否	且/或逻辑
filters	过滤条件	list	是	过滤条件
cond	要进行的比较操作	string	是	如"contain"、"not_contain"、"equal"、"not_equal"
field	要筛选的字段名	string	是	如"tags"、"channel"
value	要筛选的目标值	-	是	如"军事"、"周星驰"
weight	物料全局权重	float	否（weight & channel_weight 两者二选一）	物料全局权重，值在1-10之间，默认以该权重做排序
channel_weights	频道下的权重	list	否（weight & channel_weight 两者二选一）	频道下的权重，请求有频道情况下，会优先以该权重做排序
channel	频道列表	string	否	频道列表
weight	频道下的权重	float	否	频道下的权重，值在1-10之间
scene_name	应用场景	string	是	枚举值：tuwen/goods/media

请求示例1

```
{
  "conditions": [
    {
      "filters": [
        {
          "cond": "equal", // 匹配类型
          "field": "nid", // 字段名
          "value": "nid_1" // 匹配值
        }
      ],
      "weight": 1.0, // 物料全局权重
      "channel_weights": [ // 物料在不同频道下的权重
        {
          "channel": "频道一",
          "weight": 1.1
        },
        {
          "channel": "频道二",
          "weight": 2.1
        },
        {
          "channel": "频道三",
          "weight": 3.0
        }
      ]
    },
    {
      "filters": [
        {
          "cond": "equal",
          "field": "nid",
          "value": "nid_2"
        }
      ],
      "weight": 1.3, // 物料全局权重
      "channel_weights": [ // 物料在不同频道下的权重
        {
          "channel": "频道一",
          "weight": 1.3
        },
        {
          "channel": "频道二",
          "weight": 2.3
        },
        {
          "channel": "频道三",
          "weight": 3.3
        }
      ]
    }
  ],
  "scene_name": "tuwen"
}
```

请求示例2

物料批量删除权重:


```
{
  "conditions": [
    {
      "filters": [
        {
          "cond": "not_equal",
          "field": "nid",
          "value": "xxxx"  // 这里设置一个不存在的nid或置空即可
        }
      ],
      "weight":1.0,  // 物料全局权重，默认为1.0
      "channel_weights":[  // 物料在不同频道下的权重，清空
      ]
    }
  ],
  "scene_name": "tuwen"
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
traceid	string	是	推荐一次请求的唯一标识

返回示例

```
{
  "code": 0,
  "message": "",
  "trace_id": "286262231970967396559654621012708541514"
}
```

上下架删除接口

 上下架删除接口

接口说明

百度提供的物料上下架、删除接口。

请求示例

HTTP方法：POST

请求URL：https://airec.baidu.com/airec/api/omni/ms/batch/operation

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	数据表鉴权信息
Alias-Name	string	是	数据表名称
code	string	是	数据表类型
id	string	是	数据表id

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	类型	是否必填	说明
ids	list	是	需要删除的物料nid列表
opt	string	是	"DELETE"代表删除，"ON"代表上线，"OFF"代表下线

请求示例

```
{
  "ids": [
    "8995826958503099674"
  ],
  "opt": "DELETE"
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
success	boolean	是	成功标识
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
result	array	否	分页数据

返回示例

```
{
  "success": true,
  "code": "0",
  "message": null,
  "result": null
}
```

推荐服务接口

个性化推荐接口

接口说明

百度提供的个性化推荐接口，结合用户历史行为，推荐当前用户可能感兴趣的物料集合。

请求示例

HTTP方法：POST

请求URL：https://airec.baidu.com/airec/api/rec/p_rec

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	中文名	类型	是否必填	备注
uid	用户标识uid	string	是	用户唯一标志
page_id	推荐请求的翻页, 默认1	int	否	当前推荐页数
req_cnt	推荐数量	int	否	单次请求推荐接口获取结果的数量，默认为10
channels	频道	string	否	频道推荐, 只返回这个频道下的物料,多个值逗号分割,如vod,体育
is_personalized	是否使用个性化推荐	int	否	是否使用个性化推荐, 使用为1，不使用为0，默认为1
timestamp	时间戳	int	否	支持默认当前时间
city	当前用户所在城市	string	否	城市名称或城市编码，如：北京
filter	过滤条件	dict	否	过滤条件
scene_key	场景管理key	string	否	场景管理key

请求示例1

首页推荐请求示例：

```
{
  "uid": "xsfe123455",
  "page_id": 1,
  "req_cnt": 20,
  "is_personalized": 1
}
```

请求示例2

频道推荐请求示例：只返回物料channels字段中包含体育或科技的物料。

```
{
  "uid": "xsfe123455",
  "page_id": 1,
  "req_cnt": 20,
  "channels": "体育,科技"
}
```

请求示例3

城市推荐请求示例：只推荐物料cities字段包含全国或北京的内容。

```
{
  "uid": "xsfe123455",
  "page_id": 1,
  "req_cnt": 20,
  "city": "北京",
  "filter": {
    "cond": "contain",
    "field": "cities",
    "value": ["全国", "北京"]
  }
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
trace_id	string	是	推荐一次请求的唯一标识
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
item_list	list	是	返回的物料 nid列表
size	int	是	返回的物料数量
top_list	list	是	置顶的物料 nid列表

返回示例

```
{
  "code": 0,
  "message": "",
  "result": {
    "item_list": [
      "176",
      "2223",
      "211",
      "2046",
      "0__687639449669",
      "0__687639449672",
      "0__687639449671",
      "0__687639449673",
      "0__687639449674",
      "0__687639449675"
    ],
    "recall_result": {
      "搜索需求召回": [
        "176",
        "2046"
      ],
      "最新发布召回": [
        "2223",
        "0__687639449962",
        "NID_1001",
        "NID_1002"
      ],
      "热门物料召回": [
        "211",
        "2046"
      ],
      "短期兴趣召回": [
        "0__687639449673",
        "0__687639449674"
      ],
      "订阅作者召回": [
        "0__687639449675"
      ],
      "语义相似召回": [
        "0__687639449669",
        "0__687639449672"
      ],
      "长期偏好召回": [
        "0__687639449671"
      ]
    },
    "size": 10,
    "time": "2024-06-07 11:15:13",
    "top_list": [
      "NID_1003",
      "NID_1005"
    ],
    "traceid": "traceid-lrh67"
  },
  "trace_id": "traceid-lrh67"
}
```

相关推荐接口

 相关推荐接口

接口说明

百度提供的相关推荐接口，推荐和用户指定物料相关的物料集合。

请求示例

HTTP方法：POST

请求URL：http://airec.baidu.com/airec/api/rec/r_rec

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	中文名	类型	是否必填	备注
nids	相关物料ID	string	是	物料唯一标志
uid	用户ID	string	是	用户唯一标志
req_cnt	请求个数	int	否	单次请求物料个数, 不传默认为10
page_id	当前推荐页数	string	否	当前推荐页数，默认1
filter	过滤条件	dict	否	过滤条件
scene_key	场景管理key	string	否	场景管理key

请求示例

```
{
  "uid": "xsfe123455",
  "nids": "dsxxsfe123455",
  "page_id": 1,
  "req_cnt": 40
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
trace_id	string	是	推荐一次请求的唯一标识
item_list	list	是	返回的物料 nid列表
size	int	是	返回的物料数量

返回示例

```
{
  "code": 0,
  "message": "",
  "result": {
    "item_list": [
      "NID_1001",
      "NID_1002"
    ],
    "recall_result": {
      "热度TOP召回": [
        "2828",
        "0__647639449832"
      ],
      "短期兴趣召回": [
        "NID_1001",
        "NID_1002",
        "NID_1003",
        "NID_1004"
      ],
      "语义相似召回": []
    },
    "size": 2,
    "time": "2024-06-07 11:22:35",
    "traceid": "traceid-lrh67"
  },
  "trace_id": "traceid-lrh67"
}
```

热门推荐接口

🔗 热门推荐接口

接口说明

百度提供的热门推荐接口，基于用于点击历史计算的日热榜数据。

请求示例

HTTP方法：POST

请求URL：http://airec.baidu.com/airec/api/rec/h_rec

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	中文名	类型	是否必填	备注
uid	用户ID	string	是	用户唯一标志
hot_id	热搜榜单ID	string	否	热搜榜单ID
req_cnt	请求个数	int	否	单次请求物料个数, 不传默认为10
page_id	当前推荐页数	string	否	当前推荐页数，默认1
is_config	热搜榜配置	boolean	否	是否需要展示不在当前时间段的置顶物料，默认为false
scene_key	场景管理key	string	否	场景管理key

请求示例

```
{
  "uid": "xsfe123455",
  "hot_id": "100"
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
result	object	是	返回数据
result.item_list	object[]	是	返回的热搜榜单结果列表
result.item_list.nid	string	是	返回的物料nid
result.item_list.title	string	是	返回的物料标题
result.item_list.weight	int	是	返回的物料权重，排序权重越高，曝光几率越大
result.item_list.heat	int	是	返回的物料自然热度
result.item_list.heat_display	int	是	返回的物料展示热度，展示热度=自然热度×权重
result.item_list.label	string	是	返回的物料展示标签
result.item_list.is_top	int	是	返回的物料是否置顶，1置顶、0未置顶
result.item_list.top_n	int	否	置顶位置
result.item_list.start_time	string	否	置顶开始时间，13位时间戳
result.item_list.end_time	string	否	置顶结束时间，13位时间戳
result.item_list.hotlist_index	string	否	非置顶物料根据展示热度排序后的索引
result.size	int	是	返回的物料数量
result.time	string	是	请求时间点
result.total_count	int	是	榜单数据总量
result.traceid	string	是	推荐一次请求的唯一标识
trace_id	string	是	推荐一次请求的唯一标识

返回示例

```
{
  "code": 0,
  "message": "",
  "result": {
    "item_list": [
      {
        "heat": 1,
        "heat_display": 1,
        "hotlist_index": 1,
        "is_top": 0,
        "label": "热",
        "nid": "20",
        "title": "理财书籍套装",
        "weight": 1
      },
      {
        "heat": 1,
        "heat_display": 1,
        "hotlist_index": 2,
        "is_top": 0,
        "label": "无",
        "nid": "19",
        "title": "银行卡转账",
        "weight": 1
      }
    ],
    "size": 10,
    "time": "2024-06-06 21:24:59",
    "total_count": 100,
    "traceid": "160852490735699660090165944675305530960-lrh67",
    "trace_id": "160852490735699660090165944675305530960-lrh67"
  }
}
```

新推荐接口

🔗 新推荐接口

接口说明

百度提供的新推荐接口，按照发布时间推荐最新的物料。

请求示例

HTTP方法：POST

请求URL：https://airec.baidu.com/airec/api/rec/n_rec

Header如下：

参数名	类型	是否必填	说明
Content-Type	-	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

参数名	中文名	类型	是否必填	备注
uid	用户标识uid	string	是	用户唯一标志
page_id	当前推荐页数	int	否	当前推荐页数
req_cnt	请求个数	int	否	单次请求推荐接口获取结果的数量，默认为10
filter	过滤条件	dict	否	过滤条件
user_info_str	用户信息	string	否	{}
sort_field	排序字段	string	否	排序字段
mapKey	场景	string	否	场景

请求示例

```
{
  "uid": "xsfe123455",
  "req_cnt": 2
}
```

返回格式

JSON格式

返回参数

参数名	类型	是否必填	说明
trace_id	string	是	推荐一次请求的唯一标识
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
item_list	list	是	返回的物料 nid列表
size	int	是	返回的物料数量

返回示例

```
{
  "code": 0,
  "message": "",
  "result": {
    "item_list": [
      {
        "channels": [
          "抖屏游戏"
        ],
        "cities": [
          "beijing"
        ],
        "country": "中国",
        "data_source": "vod",
        "display_run_time": 61,
        "epi_type": "program",
        "expire_time": "1978831694000",
        "language": "中文",
        "need_pay": 1,
        "nid": "9_020005120000000012022032599001774",
        "operator_id": "CMCC",

```

```
    "operator_name": "移动",
    "provider_id": "iqiyi",
    "provider_name": "爱奇艺",
    "publish_time": "1648561924764",
    "score": "8.5",
    "status": 1,
    "title": "国乐大典 李乐苏园园《万里长城永不倒》",
    "ts": "1668768945874",
    "type": "电影",
    "valid_time": "1648561924764",
    "view_point": "看点",
    "year": "2019"
  },
  {
    "actors": [
      "广东粤剧院"
    ],
    "base_tags": [
      "粤剧"
    ],
    "channels": [
      "戏曲"
    ],
    "cities": [
      "beijing"
    ],
    "country": "中国",
    "data_source": "vod",
    "desc": "粤剧《八和会馆》描写了清朝同治年间，清政府对本地班禁戏“亡行”之后，以新华为首的年轻粤剧子弟于艰难困苦的绝境中，坚强不屈、矢志不渝，更兼学习领悟当时广东改革创新的思想，终使得官府解除禁戏。粤剧由此复兴而走向繁荣，而新华等更团结起广大粤剧艺人，以创新精神建立起保障艺人权利，组织粤剧生产，和衷共济、共存共荣的“八和会馆”，使得粤剧发展进入到崭新的阶段。",
    "display_run_time": 61,
    "epi_type": "program",
    "expire_time": "1978831694000",
    "language": "中文",
    "need_pay": 1,
    "nid": "9__02000078000000012021062399001883",
    "op_tags": [
      "粤剧"
    ],
    "operator_id": "CMCC",
    "operator_name": "移动",
    "provider_id": "iqiyi",
    "provider_name": "爱奇艺",
    "publish_time": "1648561924764",
    "score": "8.5",
    "status": 1,
    "title": "粤剧《八和会馆》黄春强版",
    "ts": "1668768945863",
    "type": "电影",
    "valid_time": "1648561924764",
    "view_point": "看点",
    "year": "2019"
  }
],
"size": 2,
"time": "2024-02-22 08:00:08",
"traceid": "156874020880905940353710878700496761108_hc5sh"
},
"trace_id": "156874020880905940353710878700496761108_hc5sh"
}
```

自定义筛选使用说明

自定义筛选使用说明

自定义筛选功能支持对物品的单个属性字段的单值过滤，也支持对多个属性字段的组合过滤。

使用限制：1、总的单值过滤规则不超过10个；2、条件组合深度不超过3层。

1、单值过滤：

支持的过滤条件有：

包含/不包含（contain/not_contain），仅支持多值字段

包含/不包含（in/not_in），仅支持单值字段

相等/不相等（equal/not_equal），仅支持单值字段

大于/小于（greater_than/less_than），仅支持数字或日期型字段

单值过滤JSON格式的过滤规则的具体语法：

```
{
  "cond": "contain" | "not_contain" | "equal" | "not_equal", // 要进行的比较操作
  "field": "<field_name>", // 要筛选的字段名，如"tags"、"channel"等
  "value": "<compare_value>" // 要筛选的目标值，如"美食"。
}
```

2、组合过滤：

对单值过滤或其他组合过滤的结果进行“且/或”的逻辑组合。过滤规则通过一个JSON字符串来表示。

组合过滤JSON格式的过滤规则的具体语法：

```
{
  "type": "and" | "or",
  "filters": [
    { 组合过滤规则 | 单值过滤规则 },
    { 组合过滤规则 | 单值过滤规则 },
    ...
  ]
}
```

3、实例：

过滤来源为“百家号”、作者为“北京人家”，且文章中有美食或科技标签的物料。

```
{
  "uid": "uid_1",
  "req_cnt": 30,
  "filter": {
    "type": "and",
    "filters": [{
      "cond": "equal",
      "field": "brand_id",
      "value": "百家号"
    },
    {
      "cond": "equal",
      "field": "anthor",
      "value": "北京人家"
    },
    {
      "type": "or",
      "filters": [{
        "cond": "contain",
        "field": "tags",
        "value": ["美食"]
      },
      {
        "cond": "contain",
        "field": "tags",
        "value": ["科技"]
      }
    ]
  }
}
```

自定义召回支持第三方服务API

一、自定义召回支持第三方服务API功能说明

支持业务方通过标准化的接口接入外部数据，将第三方数据生成的候选集与系统内置召回融合，实现多源异构召回能力的灵活扩展。

该功能适用于以下场景：

- 1、补充召回源：集成垂直领域服务（如实时新闻API），覆盖系统未触达的长尾内容。
- 2、动态数据调整：通过外部实时计算物料数据。

二、接口对接示例

请求示例

HTTP方法：POST

请求URL：http://{ip:port}/{serviceName}/{apiName}

1、输入规范样例

参数	类型	示例	说明
user_info	dict	{"uid": 10, "query", "测试"}	个性化推荐、相关推荐、智能搜索、猜你想搜uid必填，其余非必需；智能搜索、联想词、相关搜索query为必填
req_cnt	int	10	需要返回的召回物料个数
trace_id	str	"213987372302138"	traceid为必填项

2、输出规范样例

参数	类型	示例	说明
recall_item_list	list	[{"nid": "xxx", "score": 1.0}]	每个包含两个属性，nid、score为必填项

3、请求示例

• 请求示例

```
{
  "user_info": {
    "uid": "uid_1",
    "hist_rec": [
      "193"
    ],
    "query": "x",
    "channels": [
      ""
    ],
    "_timestamp": "1745821017000",
    "filter": "{\"bool\": {\"must\": [{\"term\": {\"status\": 1}}, {\"range\": {\"expire_time\": {\"gt\": 1745821017000}}}}}",
    "cur_city": ""
  },
  "req_cnt": 38,
  "trace_id": "237794112466636887852941159840767345765-8nqpx"
}

- **返回示例**
```python
{
 "recall_item_list":[
 {
 "score":1.0,
 "nid":"1"
 },
 {
 "score":0.9,
 "nid":"2"
 }
]
}
```

##### 三、注意事项

1、数据合规：第三方结果需符合内容安全规范，禁止返回违禁内容。

2、字段对齐：需与系统内容库映射，避免重复或无效ID。

```
搜索服务接口

智能搜索

搜索接口
接口说明

百度提供的物料搜索接口。

请求示例

HTTP方法：`POST`

请求URL：`http://airec.baidu.com/airec/api/search/main`

Header如下：

|参数名|类型|是否必填|说明|
|----|-----|----|-----|
|Content-Type|-|是|application/json;charset=UTF-8|
|X-Gw-Ak| string|是|推荐服务鉴权信息|
|Alias-Name| string|是|推荐服务应用名|

请求格式

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

请求参数

|参数名|中文名|类型|是否必填|备注| |
|---|---|---|---|---|---|
|uid|用户标识uid| string|是|用户唯一标识|
|query|搜索词| string|是|搜索词|
|page_id|页码| int|否|页码，默认为1|
|page_size|每页返回数量| int|否|每页返回数量，默认为10|
|filter|过滤条件| dict|否|过滤条件，使用说明可参考自定义筛选使用说明|(https://cloud.baidu.com/doc/AI_REC/s/Kluamvg6d)|
|sort_field|排序字段| string|否|后置排序字段，如果设置了，会将最终结果根据指定字段排序|
|reverse|是否逆序排列| bool|否|按照指定字段排序时，是否按照逆序排列，默认为true|
|field_weight|搜索权重| dict|否|需要搜索的字段及语义分打权重，打分权重大于0，示例：{"title":3,"content":1.8}|
|query_source|搜索来源| string|否|搜索来源，用于追踪请求来源|
|scene_key|场景管理key| string|否|场景管理key，用于获取场景对应的搜索配置|
|rank_type|排序方式| string|否|排序方式，仅当sort_field未指定时生效，支持related（按照相关性排序）和hybrid（按照综合排序）|
|query_rule_enable|query干预开关| bool|否|是否开启query规则干预，默认开启|

请求示例
```

```
{ "uid": "xsfe123455", "query": "王维" }
```



**\*\*返回格式\*\***

JSON格式

**\*\*返回参数\*\***

参数名	类型	是否必填	说明	
----	-----	----	-----	
code	string	是	响应码：0正常，其他异常	
message	string	否	响应描述	
result	object	是	返回数据	
result.final_query	string	是	最终使用的搜索词	
result.highlight	list	是	用于高亮效果展示	
result.is_correct	bool	是	是否存在纠错词	
result.is_sensitive	bool	是	是否存在敏感词	
result.item_list	object[]	是	返回的物料列表	
result.size	int	是	返回的物料数量	
result.time	string	是	请求时间点	
result.total	int	是	查询结果总数	
result.traceid	string	是	推荐一次请求的唯一标识	
trace_id	string	是	推荐一次请求的唯一标识	

**\*\*返回示例\*\***

```
{ "code": 0, "message": "", "result": { "final_query": "春游", "highlight": ["春游"], "is_correct": false, "is_sensitive": false, "item_list": [{ "title": "和晋陵路丞早春游望", "publish_time": 1735202961839, "tags": ["古诗", "诗词"] }, {"size": 1, "time": "2025-01-08 19:36:35", "total": 1, "traceid": "148180624964103756784967419111169300637-5krl2" }, {"trace_id": "148180624964103756784967419111169300637-5krl2" }
```

## ### 搜索联想词接口

## ##### 搜索联想词接口

## \*\*接口说明\*\*

百度提供的基于搜索query给出对应联想词。

## \*\*请求示例\*\*

HTTP方法：`POST`

请求URL：`http://airec.baidu.com/airec/api/query-suggest`

Header如下：

|参数名|类型|是否必填|说明|

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

|Content-Type|是|application/json;charset=UTF-8|

|X-Gw-Ak| string |是|推荐服务鉴权信息|

|Alias-Name| string| 是|推荐服务应用名|

## \*\*请求格式\*\*

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

## \*\*请求参数\*\*

| 参数名 | 中文名| 类型 | 是否必填 | 备注 |

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

|uid|用户标识uid| string| 否|用户唯一标识|

|query|请求query|string|是|请求query|

|req\_cnt|请求个数|string|否|单次请求物料个数, 最多20条, 不传默认为10|

|page\_id|当前搜索页数|string|否|当前搜索页数, 不传默认为1|

## \*\*请求示例\*\*

```
{ "uid": "uid_1", "query": "题破山寺后禅院" }
```

**\*\*返回格式\*\***

JSON格式

**\*\*返回参数\*\***

参数名	类型	是否必填	说明	
----	-----	----	-----	
code	string	是	响应码：0正常，其他异常	
message	string	否	响应描述	
result	object	是	返回数据	
result.item_info_list	object[]	是	返回的联想词结果列表	
result.item_info_list.nid	string	是	召回类型	
result.item_info_list.title	string	是	联想词结果	
result.time	string	是	请求时间点	
result.traceid	string	是	推荐一次请求的唯一标识	
trace_id	string	是	推荐一次请求的唯一标识	

**\*\*返回示例\*\***

```
{ "code": 0, "message": "", "result": { "item_info_list": [{ "nid": "前缀召回:0", "title": "阿富汗" }], "time": "2023-06-21 17:59:41", "traceid": "286262231970967396559654621012708541514" }, "trace_id": "286262231970967396559654621012708541514" }
```

## ### 相关搜索接口

## ##### 相关搜索接口

## \*\*接口说明\*\*

百度提供的相关搜索接口，在搜索结果页面推荐与query相关的其他query；在详情页推荐与该物料相关的query，激发搜索需求减少输入成本。

## \*\*请求示例\*\*

HTTP方法：`POST`

请求URL：`http://airec.baidu.com/airec/api/related-query`

Header如下：

参数名	类型	是否必填	说明
Content-Type	是	application/json;charset=UTF-8	
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

## \*\*请求格式\*\*

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

## \*\*请求参数\*\*

参数名	中文名	类型	是否必填	备注
query	搜索词	string	是	搜索结果页传query或者详情页传物料标题
page_id	当前搜索页数	int	否	当前搜索页数，不传默认为1
req_cnt	请求个数	int	否	单次请求物料个数，不传默认为10

## \*\*请求示例\*\*

```
{ "query": "故人" }
```

## \*\*返回格式\*\*

JSON格式

## \*\*返回参数\*\*

参数名	类型	是否必填	说明
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
result	object	是	返回数据
result.item_list	list	是	返回的相关搜索结果列表
result.querys	object	是	返回的相关搜索结果列表
result.time	string	是	请求时间点
result.traceid	string	是	推荐一次请求的唯一标识
trace_id	string	是	推荐一次请求的唯一标识

## \*\*返回示例\*\*

```
{ "code": 0, "message": "", "result": { "item_list": ["故人的诗句", "故人归"], "query": { "query热门召回":["故人的诗句"], "高共现概率":["故人归"] } "time": "2024-02-22 08:00:08", "traceid": "295403186087133234193866971937309743415" }, "trace_id": "295403186087133234193866971937309743415" }
```

### 热门搜索接口

##### 热门搜索接口

\*\*接口说明\*\*

百度提供的热门搜索接口，根据用户搜索行为配置计算主榜和分类榜单，帮助用户发现平台搜索需求和兴趣热点。

\*\*请求示例\*\*

HTTP方法：`POST`

请求URL：`http://airec.baidu.com/airec/api/search/h\_search`

Header如下：

参数名	类型	是否必填	说明
Content-Type	string	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

\*\*请求格式\*\*

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

\*\*请求参数\*\*

参数名	中文名	类型	是否必填	备注
uid	用户标识uid	string	是	用户唯一标识
hot_id	热搜榜单ID	string	否	热搜榜单ID
req_cnt	请求个数	int	否	单次请求物料个数, 最多100条, 不传默认为10
page_id	当前搜索页数	int	否	当前搜索页数, 不传默认为1
is_config	热搜榜配置	bool	否	是否需要展示未来时间段的置顶物料, 默认为false

\*\*请求示例\*\*

```
{ "uid": "uid_1", "hot_id": "65" }
```

**\*\*返回格式\*\***

JSON格式

**\*\*返回参数\*\***

参数名	类型	是否必填	说明
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
result	object	是	返回数据
result.hot_list	object[]	是	返回的热搜榜单结果列表
result.hot_list.nid	string	是	返回的物料nid
result.hot_list.title	string	是	返回的物料标题
result.hot_list.weight	int	是	返回的物料权重，排序权重越高，曝光几率越大
result.hot_list.heat	int	是	返回的物料自然热度
result.hot_list.heat_display	int	是	返回的物料展示热度，展示热度=自然热度×权重
result.hot_list.label	string	是	返回的物料展示标签
result.hot_list.is_top	int	是	返回的物料是否置顶，1置顶、0未置顶
result.hot_list.top_n	int	否	置顶位置
result.hot_list.start_time	string	否	置顶开始时间，13位时间戳
result.hot_list.end_time	string	否	置顶结束时间，13位时间戳
result.hot_list.hotlist_index	string	否	非置顶物料根据展示热度排序后的索引
result.time	string	是	请求时间点
result.total_count	int	是	榜单数据总量
result.traceid	string	是	推荐一次请求的唯一标识
trace_id	string	是	推荐一次请求的唯一标识

**\*\*返回示例\*\***

```
{ "code": 0, "message": "", "result": { "hot_list": [{ "end_time": 1708191981901, "heat": 1, "heat_display": 1, "is_top": 1, "label": "无", "nid": "41", "start_time": 1708796785901, "title": "子夜四时歌：秋歌", "top_n": 2, "weight": 1 }, { "end_time": 1708019754166, "heat": 1, "heat_display": 1, "is_top": 1, "label": "无", "nid": "26", "start_time": 1709142956166, "title": "贼退示官吏并序", "top_n": 3, "weight": 1 }], "time": "2024-02-22 08:00:08", "total_count": 100, "traceid": "101169684362981887882145214525820211904" }, "trace_id": "101169684362981887882145214525820211904" }
```

## ### 猜你想搜接口

## ##### 猜你想搜接口

## \*\*接口说明\*\*

根据用户的搜索历史和偏好，推荐相关的搜索query，提供更加个性化的搜索体验。

## \*\*请求示例\*\*

HTTP方法：`POST`

请求URL：`https://airec.baidu.com/airec/api/query-discovery`

Header如下：

参数名	类型	是否必填	说明
Content-Type	string	是	application/json;charset=UTF-8
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

## \*\*请求格式\*\*

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

## \*\*请求参数\*\*

参数名	中文名	类型	是否必填	备注
uid	用户标识uid	string	是	用户唯一标识
page_id	页码	int	否	页码，默认为1
req_cnt	每页返回数量	int	否	每页返回数量，默认为10

## \*\*请求示例\*\*

```
{ "uid": "uid_1" }
```

## \*\*返回格式\*\*

JSON格式

## \*\*返回参数\*\*

参数名	类型	是否必填	说明
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
result	object	是	返回数据
result.item_list	list	是	返回的相关query列表
result.size	int	是	返回的相关query数量
result.time	string	是	请求时间点
result.traceid	string	是	推荐一次请求的唯一标识
trace_id	string	是	推荐一次请求的唯一标识

## \*\*返回示例\*\*

```
{ "code": 0, "message": "", "result": { "item_list": ["古诗", "星座", "导演",], "size": 3, "time": "2024-06-07 01:29:09",
"traceid": "054324039656044763419466023962854299769-pgvns" }, "trace_id":
"054324039656044763419466023962854299769-pgvns" }
```

### 历史搜索接口

##### 历史搜索接口

\*\*接口说明\*\*

用户可以查看自己过去的搜索词或查询，从而快速找到之前查找过的信息或重新访问之前感兴趣的内容。

\*\*请求示例\*\*

HTTP方法：`POST`

请求URL：`http://airec.baidu.com/airec/api/search/h\_search`

Header如下：

参数名	类型	是否必填	说明
----   -----   ----   -----			
Content-Type		是	application/json;charset=UTF-8
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

\*\*请求格式\*\*

POST 方式调用

注意：要求使用 JSON 格式的结构体来描述一个请求的具体内容。

\*\*请求参数\*\*

参数名	中文名	类型	是否必填	备注
----   ----   ----   -----   -----				
uid	用户标识uid	string	是	用户唯一标识

\*\*请求示例\*\*

```
{ "uid": "uid_1" }
```



```

返回格式

JSON格式

返回参数

| 参数名 | 类型 | 是否必填 | 说明 |
| ---- | ---- | ---- | ---- |
| code | string | 是 | 响应码：0正常，其他异常 |
| message | string | 否 | 响应描述 |
| result | object | 是 | 返回数据 |
| result.hist_query | list | 是 | 用户历史搜索列表 |
| result.time | string | 是 | 请求时间点 |
| result.traceid | string | 是 | 推荐一次请求的唯一标识 |
| trace_id | string | 是 | 推荐一次请求的唯一标识 |

返回示例

```

```

{ "code": 0, "message": "", "result": { "hist_query": ["银行卡转账", "理财书籍套装1", "理财"], "time": "2024-06-07
01:21:54", "traceid": "078528520092713688324668088542794041419-pgvns" }, "trace_id":
"078528520092713688324668088542794041419-pgvns" }

```

```

自然语言搜索配置

自然语言搜索配置
使用场景

将用户输入自然语言转化为机器可理解的查询语言，而非关键字检索，例如用户可输入“高收益且支持现金分红的基金产
品”。

输入要求

指令需包含以下变量：

| 变量名 | 描述 | 是否必须 |
| ---- | ---- | ---- |
| ${fields} | 索引字段，在自然语言搜索配置中添加的'问答字段' | 是 |
| ${document} | 索引字段说明，对'问答字段'的详细描述 | 是 |
| ${query} | query | 是 |

提示词示例

> 请将以下自然语言查询转换为符合ElasticSearch语法的查询语句，并以JSON格式输出。
>
> 要求：如果query中包含match，则需要把match放入bool.must里；禁止在查询语句中携带注释和说明；时间范围查询必
须使用13位时间戳；如果信息不足以回答问题，请输出：这个问题我暂时回答不了。
以下是索引字段说明，并且查询字段只能包含${fields}
>
> ${document}
>
> 示例输入:搜索作者ID不为UID_1001，价格在50元左右的手機,结果根据价格降序排序
>
> 示例输出:
>

```

```

{ "query": { "bool": { "must": [{ "range": { "price": { "gte": 45, "lt": 55 } } }, { "term": { "tags.keyword": { "value": "手机" } } }],
"must_not": [{ "term": { "author_id": { "value": "UID_1001" } } }] }, "sort": [{ "price": { "order": "desc" } }] }

```

> 现在请转换:

>

> 输入:\${query}

>

> 输出:

### 自然语言搜索接口

##### 自然语言搜索接口

\*\*接口说明\*\*

将用户输入自然语言转化为机器可理解的查询语言，而非关键字检索，例如用户可输入“高收益且支持现金分红的基金产品”。

\*\*请求示例\*\*

HTTP方法: `POST`

请求URL: `http://airec.baidu.com/airec/api/language-query`

Header如下:

参数名	类型	是否必填	说明
Content-Type	是	application/json;charset=UTF-8	
X-Gw-Ak	string	是	推荐服务鉴权信息
Alias-Name	string	是	推荐服务应用名

\*\*请求格式\*\*

POST 方式调用

注意: 要求使用 JSON 格式的结构体来描述一个请求的具体内容。

\*\*请求参数\*\*

参数名	中文名	类型	是否必填	备注
uid	用户标识uid	string	是	用户唯一标识
query	搜索词	string	是	搜索词
filter	过滤条件	dict	否	过滤条件，使用说明可参考[自定义筛选使用说明](https://cloud.baidu.com/doc/AI_REC/s/Kluamvg6d)
page_id	页码	int	否	页码，默认为1
page_size	每页返回数量	int	否	每页返回数量，默认为10
query_source	搜索来源	string	否	搜索来源，用于追踪请求来源
scene_key	场景管理key	string	否	场景管理key，用于获取场景对应的搜索配置
timestamp	时间戳	int	否	时间戳

\*\*请求示例\*\*

```
{ "uid": "xsfe123455", "query": "低风险且支持现金分红的基金" }
```

**\*\*返回格式\*\***

## JSON格式

**\*\*返回参数\*\***

参数名	类型	是否必填	说明
code	string	是	响应码：0正常，其他异常
message	string	否	响应描述
success	bool	是	请求是否被成功处理
trace_id	string	是	一次请求的唯一标识
result	object	是	返回数据
result.is_success	bool	是	问题是否被回答
result.answer	string	是	问题回答结果
result.item_list	object[]	是	返回的物料列表
result.size	int	是	返回的物料数量
result.time	string	是	请求时间点
result.total	int	是	查询结果总数
result.traceid	string	是	一次请求的唯一标识

**\*\*返回示例\*\***

```
{ "success": true, "message": { "global": "ok" }, "code": 0, "result": { "size": 6, "time": "2025-04-17 12:33:51", "total": 6,
"traceid": "207683259491340058310980645227791028219-s7qr4", "answer": "{\"query\": {\"bool\": {\"must\": [{\"term\":
{\"risk_level.keyword\": \"低风险\"}}, {\"term\": {\"dividend.keyword\": \"现金分红\"}}]}}\", \"is_success\": true, \"item_list\": [{ \"nid\":
\"14429\", \"title\": \"[存单AAA指数7天持有期证券投资基金]\", \"risk_level\": \"低风险\", \"dividend\": \"现金分红\" }] }, \"trace_id\":
\"207683259491340058310980645227791028219-s7qr4\" }
```

## # SDK文档

## ## PythonSDK

```
1、数据接入
1.1、物料数据接入
```

```
import os
import time
import requests
import json
```

```
def post_m(dic):
 ctx = json.dumps(dic, ensure_ascii=False)
 ctx = ctx.encode("utf-8")
 rex = requests.post(url, ctx, headers=headers)
 return rex
```

```
url = "https://airec.baidu.com/airec/api/omni/cms/batch-sync"
ak = "323475515d4bfccc1af3f3676ec0256d"
```

```
headers = {"Content-Type": "application/json", "X-Gw-Ak": ak, "Alias-Name": "in9iamvk-26"}
```

```
demo = [{ "nid": "nid_1", "site_name": "mr.yesterday", "title": "杂诗-王维-1", "content": "双燕初命子，五桃新作花，来日绮窗前，寒梅著花未。", "publish_time": 1648561924764, "status": 1, "tags": ["杂诗王维"], # 如果需要频道推荐则使用此字段
"channels": ["科技"], "ts": int(time.time() * 1000) }
```

```
begin = time.time()
rex = post_m(demo)
end = time.time()
print("using time: {}, status {}, ret content: {}".format(round(end - begin, 6), rex.status_code, rex.text))
```

## ##### 1.2、用户数据接入

```
import time import requests import json

def post_m(dic): ctx = json.dumps(dic, ensure_ascii=False) ctx = ctx.encode("utf-8") rex = requests.post(url, ctx,
headers=headers) return rex

url = "https://airec.baidu.com/airec/api/omni/ums/batch-sync" ak = "323475515d4bfccc1af3f3676ec0256d"

headers = {"Content-Type": "application/json", "X-Gw-Ak": ak, "Alias-Name": "in9iamvk-26"}

demo = [{"uid": "f"8-b8xvi89a_1", "city": "北京", "uid_type": "1", "tags": ["杂诗王维"]}]]

begin = time.time() rex = post_m(demo) end = time.time() print("using time: {}, status {}, ret content: {}".format(round(end -
begin, 6), rex.status_code, rex.text))
```

### ##### 1.3、行为数据接入

```
import time import requests import json

def post_m(dic): ctx = json.dumps(dic, ensure_ascii=False) ctx = ctx.encode("utf-8") rex = requests.post(url, ctx,
headers=headers) return rex

url = "https://airec.baidu.com/airec/api/omni/ims/batch-sync" ak = "323475515d4bfccc1af3f3676ec0256d"

headers = {"Content-Type": "application/json", "X-Gw-Ak": ak, "Alias-Name": "in9iamvk-26"}

demo = [{"uid": "uid_1", "nid": "nid_1", "event": "show", "traceid": "232808585383426618326892988534043392144",
"ts": int(time.time() * 1000) }] print("data:", json.dumps(demo, ensure_ascii=False)) begin = time.time() rex = post_m(demo)
end = time.time() print("using time: {}, status {}, ret content: {}".format(round(end - begin, 6), rex.status_code, rex.text))
```

### ##### 2、服务调用

#### ##### 2.1、个性化推荐接口

```
import os import time import requests import json

def post_m(dic): rex = requests.post(url, json=dic, headers=headers) return rex

url = "https://airec.baidu.com/airec/api/rec/p_rec" ak = "323475515d4bfccc1af3f3676ec0256d"

headers = {"Content-Type": "application/json", "X-Gw-Ak": ak, "Alias-Name": "in9iamvk-26"}

body = { "uid": "uid_1", # 如果使用频道推荐, 则传入此字段, 同时确保物料中已经传入此字段 # "channels": "tv,科技",
"req_cnt": 20 }

begin = time.time() rex = post_m(body) end = time.time() print("using time: {}, status {}, ret content: {}".format(round(end -
begin, 6), rex.status_code, rex.text))
```

#### ##### 2.2、相关推荐

```
import os

import time import requests import json

def post_m(dic): rex = requests.post(url, json=dic, headers=headers) return rex

url = "https://airec.baidu.com/airec/api/rec/r_rec" ak = "323475515d4bfccc1af3f3676ec0256d"

headers = {"Content-Type": "application/json", "X-Gw-Ak": ak, "Alias-Name": "in9iamvk-26"}
```

```
body = { "uid": "uid_1", "nids": "nid_1", "req_cnt": 20 }
```

```
begin = time.time() rex = post_m(body) end = time.time() print("using time: {}, status {}, ret content: {}".format(round(end -
begin, 6), rex.status_code, rex.text))
```

### ##### 2.3、智能搜索

```
import os import time import requests import json
```

```
def post_m(dic): rex = requests.post(url, json=dic, headers=headers) return rex
```

```
url = "https://airec.baidu.com/airec/api/search/main" ak = "323475515d4bfccc1af3f3676ec0256d" headers = {"Content-
Type": "application/json", "X-Gw-Ak": ak, "Alias-Name": "in9iamvk-26"}
```

```
body = { "uid": "uid_1", "query": "王维" }
```

```
begin = time.time() rex = post_m(body) end = time.time() print("using time: {}, status {}, ret content: {}".format(round(end -
begin, 6), rex.status_code, rex.text))
```

```
PHP SDK
```

```
一、数据接入
```

```
1. 物料数据接入
```

```
"nid_1", "site_name" => "mr.yesterday", "title" => "杂诗-王维-1", "summary" => "双燕初命子", "content" => "双燕初命子，
五桃新作花，来日倚窗前，寒梅著花未。", "publish_time" => 1648561924764, "status" => 1, "tags" => ["杂诗王维", "杂
诗"], "channels" => ["科技", "文学"], "author" => "王维", "content_type" => "图文", "type" => "古诗", "source" => "1", "ts"
=> time() * 1000]; $begin = microtime(true); $rex = post_m($demo); $end = microtime(true); echo "using time: " .
round($end - $begin, 6) . ", ret content: " . $rex; ``` **2. 用户数据接入** ``` 'dd5ym1b8-26_1', 'city' => '北京', 'uid_type' =>
'1', 'tags' => ['杂诗王维']]; $begin = microtime(true); $rex = post_m($demo); $end = microtime(true); echo "using time: " .
round($end - $begin, 6) . ", ret content: " . $rex; ``` **3. 行为数据接入** ``` 'uid_1', 'nid' => 'nid_1', 'event' => 'show',
'traceid' => '232808585383426618326892988534043392144', 'ts' => round(microtime(true) * 1000)]; $begin =
microtime(true); $rex = post_m($demo); $end = microtime(true); echo "using time: " . round($end - $begin, 6) . ", ret content:
" . $rex; ``` ##### 二、搜索接口 **1. 图文智能搜索** ``` 'uid_1', 'query' => '王维', 'sort_field' => 'publish_time', # 按照时
间排序，不传默认按照相关性排序 'reverse' => true, # 不传默认true # 支持的筛选类型：
equal,contain,greater_than,less_than,not_equal,not_contain 'filter' => ['type' => 'and', 'filters' => [['cond' => 'less_than',
'field' => 'publish_time', 'value' => 1748561924765], ['cond' => 'greater_than', 'field' => 'publish_time', 'value' =>
1648561924767], ['cond' => 'equal', 'field' => 'source', 'value' => '1'], ['cond' => 'not_equal', 'field' => 'type', 'value' =>
'abc']]]; $begin = microtime(true); $rex = post_m($body); $end = microtime(true); echo "using time: " . round($end -
$begin, 6) . ", ret content: " . $rex; ``` **2. 搜索联想词** ``` 'uid_1', 'query' => '王维']; $begin = microtime(true); $rex =
post_m($body); $end = microtime(true); echo "using time: " . round($end - $begin, 6) . ", ret content: " . $rex; ``` **3. 相关搜
索** ``` 'query_1']; $begin = microtime(true); $rex = post_m($body); $end = microtime(true); echo "using time: " .
round($end - $begin, 6) . ", ret content: " . $rex; ``` **4. 热门搜索** ``` 'uid_1', 'type' => '7day', 'req_cnt' => 20, 'page_id'
=> 1, 'sort_field' => 'click', 'update_rate' => '24hour']; $begin = microtime(true); $rex = post_m($body); $end =
microtime(true); echo "using time: " . round($end - $begin, 6) . ", ret content: " . $rex; ``` ##### 三、推荐接口 **1. 图文个性
化推荐** ``` 'uid_1', // 如果使用频道推荐，则传入此字段，同时确保物料中已经传入此字段 // 'channels' => 'tv,科技',
'req_cnt' => 20]; $begin = microtime(true); $rex = post_m($body); $end = microtime(true); echo "using time: " . round($end -
$begin, 6) . ", ret content: " . $rex; ``` **2. 图文相关推荐** ``` 'uid_1', 'nids' => 'nid_1', 'req_cnt' => 20]; $begin =
microtime(true); $rex = post_m($body); $end = microtime(true); echo "using time: " . round($end - $begin, 6) . ", ret content:
" . $rex; ``` **3. 图文热门推荐** ``` 'uid_1', 'page_id' => 1, 'req_cnt' => 10]; $begin = microtime(true); $rex =
post_m($body); $end = microtime(true); echo "using time: " . round($end - $begin, 6) . ", ret content: " . $rex; ``` # 数据规范
```

## 媒资数据字段规范 | 字段名称 | 中文名 | 类型 | 是否必传 | 数据样例 | 备注 | | --- | --- | --- | --- | --- | --- | |nid|物料标识|string|是|NID\_1001|一条物料的唯一标识, 物料id不能重复 title|媒资标题|string|是|示例媒资标题|电影、剧集、节目等标题, 主要用来进行语义分析提取物料特征 display\_run\_time|媒资时长|int|否|61|播放时长(秒)| country|国家地区|string|否|大陆|发布国| year|上映年份|string|否|2002|上映年份, 用作特征, 如生成某年的排行榜。| desc |媒资描述| string| 否| 示例描述| 视频描述, 主要用来进行语义分析提取物料特征 actors |演员列表 |strings|否|["成龙"] |演员列表, 支持多个值 directors| 创作者 |strings |否| ["袁和平"] |媒资创作者/导演/出品方id,支持多个值 view\_point |看点 |string |否| 样例描述| 内容看点 score |评分 |string| 否| 8.5 |评分 data\_source| 物料来源(电信/移动/联通/其它)| string |是| CTC |运营商或运营商编码: 电信/移动/联通/其它 epi\_type |媒资类型 |string| 否 |program |物料类型, program单剧集、series连续剧/系列剧、episode子集、channel频道、schedule节目单、album专栏、subject专题 content\_type| 内容类型 |string| 否| video| 指定内容是视频还是图文类型 base\_tags| 媒资标签 |strings |否| ["动作"] |内容标签,支持多个值 op\_tags |运营标签| strings| 否| ["VIP", "院线"] |业务方运营的标签 show\_tags |展示标签 |strings| 否| ["HD"] |展示标签 valid\_time| 生效时间 |long| 否 |1648561924764| 物料生效时间, 用于控制物料可被推荐的生效时间, 13位时间戳 expire\_time| 过期时间| long| 否 |1978831694000 |物料失效时间, 用于控制物料过期后不被推荐的场景, 13位时间戳 publish\_time |发布时间| long| 是| 1648561924764| 发布时间, 格式: 13位时间戳 operator\_id |运营商ID| string |否 |CMCC| 运营商(电信、联通、移动等) operator\_name |运营商名称 |string |否 |移动| 运营商(电信、联通、移动等) provider\_id |供应商ID, 直播情况下为直播的频道ID| string |否 |qiqi| 供应方ID,媒资提供商或者来源,用作特征 provider\_name |供应商, 直播情况下为直播的频道名称 |string| 否 |爱奇艺| 供应方名称,媒资提供商或者来源,用作特征 need\_pay |是否付费| int| 否| 1 |是否付费, 0 (不付费)、1 (付费) channels| 频道| strings |否| ["体育", "科技", "vod"] |用户自定义标签, 可在推荐接口作为推荐条件进行数据过滤 ts |上传时间 |long |是 |1648561924764 |物料上传时间, 格式: 13位时间戳 url |视频地址 |string |否 |视频地址 language |影视语言 |string| 否 |中文 |媒资所用语言, 用作特征 award |获奖| string |否 |金马奖 |用作特征 definition |清晰度| string| 否 |高清 |用作特征 cities| 城市名称或编码列表 |strings| 否| ["beijing", "shanghai"] |物料所属的城市列表 live\_tags |直播时间戳标签列表 |strings| 否| [] |直播标签 media\_code| 媒资编码 |string |否 | |媒资编码 base\_tag\_codes| 媒资编码 |strings| 否| ["0042", "0006", "0008"] |媒资编码 compere |主持人 |strings |否| ["柳敬东", "喻明"] |主持人 type| 类型标识 |string |否 |电影 |包含电视剧、电影、综艺、儿童动漫、新闻、音乐、体育、纪录片、游戏、直播、短内容合辑、戏曲、公开课, 其他类型填未知, 用作特征 type\_code |类型标识编码 |string| 否| 001 |类型标识编码 heat |热度| float| 否| 1.0 |热度 is\_subset |是否是子集 (如花千骨第二集) | int| 否| 0 |是否是子集 (如花千骨第二集), 0 (不是)、1 (是) # 最佳实践 ## 图文个性化推荐案例 ##### 一、业务背景 某生活APP是以提供本地生活服务为主的一站式本地生活服务平台, 覆盖餐饮、出行、公共服务、娱乐、购物、健康等生态, 提供点餐、买单、外卖、等服务, 以美食餐饮为主场景, 为了提高APP商业转化和用户活跃度, 需要以种草内容推荐带动各个板块业务目标。本案例主要介绍了使用智能推荐引擎实现图文类物料的个性化场景推荐。 ##### 二、数据准备 > 1、物料数据schema规范 > ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/super\_admin/airec/image\_84c39f1.png) > 数据样例: 144 rows × 9 columns 家居40条、科技40条、美食60 > > 2、行为数据schema规范 > ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/super\_admin/airec/image\_80e726a.png) > 数据样例: 155条 > > 3、用户数据schema规范 (非必传) > ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/super\_admin/airec/image\_b5a2533.png) > 样例数据: 18条 ##### 三、创建应用 在应用管理页, 单击顶部【创建应用】按钮, 进入创建应用页面, 依次填写应用各项参数, 选择图文推荐。点击【创建】按钮, 进入创建应用提示页面, 进入应用配置页面。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/super\_admin/airec/image\_6efda6f.png) ##### 四、接入数据 进入应用配置页面, 在数据接入选项卡下点击数据API卡片右上角的【启用服务】按钮一键开启3个数据接入服务。点击【刷新状态】按钮, 直到三个服务的状态都显示为“服务中”时, 即可根据生成的请求地址接入数据了。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/super\_admin/airec/image\_53926d9.png) 1、了解使用SDK方式接入数据的流程: ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/super\_admin/airec/image\_cb13a28.png) > 第1步: 下载代码模版, 下载并解压代码模版 (python文件), 请仔细阅读readme.md文件。 > > 第2步: 下载并替换鉴权信息, 点击下载鉴权信息, 并参考代码模板中readme.md文件进行修改。 > > 第3步: 接入数据, 返回'success':true即表明接入成功, 刷新数据状态, 接入必传数据即可。 \*\*2、分别接入三类数据\*\* \*\*第一步: 接入csv中的物料数据\*\* 注意: 本案例脚本本代码与sdk提供的存在差异仅供参考, 您需要根据实际业务使用sdk中的代码或者开发代码。 `` `` 从csv中发送物料数据给cms `` import os import time import requests import json import pandas as pd def post\_m(dic): ctx = json.dumps(dic, ensure\_ascii=False) ctx = ctx.encode("utf-8") rex = requests.post(url, ctx, headers=headers) return rex url = "http://airec.baidu.com/airec/api/1-s6r5gitg/omni/cms/batch-sync"

```

ak = "f5695cc1c9544ccd9088b29b417d395b" headers = {"Content-Type": "application/json", "X-Gw-Ak": ak} cmsdata =
pd.read_csv('/Users/wangxuecong01/Desktop/1产品管理/1智能推荐引擎/演示数据/CMSdemo.csv') for idx, data in
cmsdata.iterrows(): demo_cms= [{"nid": str(data['nid']), "site_name":data['site_name'], "title":data['title'],
"content":data['content'], "mthid":str(data['mthid']), "publish_time":data['publish_time'], "att":[data['att']],
"status":data['status'], "ts":data['ts']}] begin = time.time() rex = post_m(demo_cms) end = time.time() print("using time: {},
status {}, ret content: {}".format(round(end - begin, 6), rex.status_code, rex.text)) ``` **第二步：接入csv中的用户数据** ```
""" 发送用户数据给ums """ import os import time import requests import json import pandas as pd def post_m(dic): ctx =
json.dumps(dic, ensure_ascii=False) ctx = ctx.encode("utf-8") rex = requests.post(url, ctx, headers=headers) return rex url =
"http://airec.baidu.com/airec/api/1-fd8wpdx0/omni/ums/batch-sync" ak = "f5695cc1c9544ccd9088b29b417d395b"
headers = {"Content-Type": "application/json", "X-Gw-Ak": ak} umsdata = pd.read_csv('/Users/文件路径/演示数
据/UMSdemo.csv') for idx, data in umsdata.iterrows(): demo_ums = [{"uid":str(data['uid'])}] begin = time.time() rex =
post_m(demo_ums) end = time.time() print("using time: {}, status {}, ret content: {}".format(round(end - begin, 6),
rex.status_code, rex.text)) ``` **第三步：接入csv中的用户数据** ``` """ 发送用户数据给ums """ import os import time
import requests import json import pandas as pd def post_m(dic): ctx = json.dumps(dic, ensure_ascii=False) ctx =
ctx.encode("utf-8") rex = requests.post(url, ctx, headers=headers) return rex url = "http://airec.baidu.com/airec/api/1-
fd8wpdx0/omni/ims/batch-sync" ak = "f5695cc1c9544ccd9088b29b417d395b" headers = {"Content-Type":
"application/json", "X-Gw-Ak": ak} imdata = pd.read_csv('/文件路径演示数据/IMSdemo.csv') for idx, data in
imdata.iterrows(): demo_ims = [{"nid":str(data['nid']), "uid":str(data['uid']), "event":data['event'], "ts":data['ts']}] begin =
time.time() rex = post_m(demo_ims) end = time.time() print("using time: {}, status {}, ret content: {}".format(round(end - begin,
6), rex.status_code, rex.text)) ``` **3、验证是否成功接入数据** 在配置应用页面基础信息卡片查看数据接入条数是否符合预
期。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-
cdn.bj.bcebos.com/doc/super_admin/airec/image_5820149.png) ##### 五、服务调用 **1、启用服务** 进入智能推荐引擎
配置页面，在服务调用选项卡下点击推荐API卡片【启用】按钮开启需要调用的服务。 ![image.png](http://bjhw-bce-online-
public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-
cdn.bj.bcebos.com/doc/super_admin/airec/image_c5559f3.png) 点击【刷新状态】按钮，直到服务的状态都显示为“服务
中”时，即可根据生成的请求地址调用接口。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-
external/?url=http://bce-cdn.bj.bcebos.com/doc/super_admin/airec/image_a0bef8e.png) **2、了解调用流程** !
[image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-
cdn.bj.bcebos.com/doc/super_admin/airec/image_08ecc1d.png) >> 第1步：下载代码模版，下载并解压代码模版（python文
件），请仔细阅读readme.md文件。> 第2步：下载并替换鉴权信息，点击下载鉴权信息，并参考代码模板中readme.md文件
进行修改。> 第3步：调用服务，获取返回的推荐结果。> 第4步：验证结果，正确返回值类似，status值应为 200。 **3、开
始调用接口** 调用个性化接口 ``` """ 个性化推荐接口 """ import os import time import requests import json def post_m(dic):
rex = requests.get(url, dic, headers=headers) return rex url = "http://10.137.16.168:8081/airec/api/3-zo179t-l/rec/p_rec"
ak = "f5695cc1c9544ccd9088b29b417d395b" headers = {"Content-Type": "application/json", "X-Gw-Ak": ak} body = { "uid":
"uid_1", "req_cnt": 20 } begin = time.time() rex = post_m(body) end = time.time() print("using time: {}, status {}, ret content:
{}".format(round(end - begin, 6), rex.status_code, rex.text)) ``` ##### 六、模拟体验 在应用列表页面选择相应的应用点击【配
置】按钮进入配置页面，切换到服务调用页面，点击【模拟体验demo】按钮。 **1、体验个性化PC端推荐** 点击【刷新】按
钮，可以浏览推荐内容，您可以点击【阅读（曝光）】、【点击】、【评论】、【收藏】、【转发】按钮模拟用户的点击行
为，每次点击都会生成一条行为数据用于给当前用户推荐。 ![image.png](http://bjhw-bce-online-
public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-
cdn.bj.bcebos.com/doc/super_admin/airec/image_790c6a3.png) 查看用户兴趣点变化趋势：点击模拟用户的兴趣点可以查看
该兴趣点（标签）在每天的变化趋势。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?
url=http://bce-cdn.bj.bcebos.com/doc/super_admin/airec/image_e55407b.png) 查看推荐内容分布和用户兴趣分布是否吻
合，或者查询模拟行为记录。 **2、体验手机端个性化推荐** 在入pc demo页面点击【二维码】图标弹出二维码页面，选择模
拟用户，手机微信或者其他应用扫一扫打开H5页面。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-
external/?url=http://bce-cdn.bj.bcebos.com/doc/super_admin/airec/image_6473767.png) 下拉刷新可以浏览推荐更多的内
容，您可以点击【阅读（曝光）】、【点击】、【评论】、【收藏】、【转发】按钮模拟用户的点击行为，每次点击都会生成
一条行为数据用于给当前用户推荐，暂不支持取消行为。同样的，您可以上滑查看历史推荐内容。 ![image.png](http://bjhw-
bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-
cdn.bj.bcebos.com/doc/super_admin/airec/image_2d7ab74.png) ![image.png](http://bjhw-bce-online-

```



public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-

cdn.bj.bcebos.com/doc/super\_admin/airec/image\_72878f4.png) # 常见问题 ## 系统类问题 \*\* (1)平台对浏览器的要求

是? \*\* 答: 产品支持 Chrome 内核的浏览器, 暂不支持 IE 浏览器, 建议您使用谷歌或者火狐浏览器进行访问。 \*\* (2)如何申

请测试和试用? \*\* 答: 您可以选择试用云上 SaaS 版本或私有部署版本。目前云上 SaaS 版为所有用户提供99元30 天的poc

试用期, 试用期间您将体验到「标准版」的大部分功能; 每个百度智能云账号只提供一次poc试用机会。 \*\* (3)是否支持私有

化, 私有部署是否支持产品的版本升级? \*\* 答: 支持, 私有部署情况下支持产品的版本升级。 ## 功能问题 \*\* (1) 什么是

推荐系统? \*\* 答: 推荐系统是一种基于用户行为和偏好的计算机系统, 旨在预测和推荐用户可能感兴趣的内容或产品。它通

过分析用户的历史行为、购买记录、浏览记录和其他相关信息, 来预测用户可能喜欢或需要的产品或内容。 \*\* (2) 对于未登

录的游客, 怎么推送内容? \*\* 答: 系统有兜底策略, 按照最热、最新内容推荐。 \*\* (3) 推荐系统中算法有哪些? \*\* 答: 推荐

系统中常用的算法包括协同过滤算法、基于内容的推荐算法、混合推荐算法等。这些算法各有优缺点, 需要根据具体应用场景

选择合适的算法。 \*\* (4) 如何优化推荐系统的? \*\* 答: 优化推荐系统需要从多个方面入手, 包括数据收集、特征选择、模

型训练和结果展示等。可以通过改进算法、增加训练数据、调整模型参数等方式来提高推荐系统的准确性和效果。此外, 还可

以通过A/B测试等方式来评估不同优化策略的效果。 \*\* (5) 百度推荐系统的用户行为数据结构中, 行为类型: show 展现、

click 点击、collect 收藏、follow 关注。能否增加点赞行为? \*\* 答: 支持上传其他行为数据, 可以增加其他行为。 # 更新日志

## 智能搜推引擎V\_3.8.0版本新特性 智能搜推引擎V\_3.8.0版本计划于2025年1月16日上线公有云, 欢迎体验。本次升级内容

主要有: ##### 一、本期升级能力 ##### 1、新增搜索结果追溯, 提高badcase问题定位效率 \* 搜索badcase闭环运营流程

新增归因分析工具, demo升级按照业务需求设置过滤条件及排序结果, 可实时查看query解析过程并评估物料匹配。 \* 新增搜

索链路分析便于快速定位问题, 并可以采取相应干预措施, 增强用户体验, 提供精准搜索结果。 ![image.png](http://bjhw-bce-

online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-

share/AI\_REC/image\_5846de8.png) ##### 2、扩展营销触达能力: 支持场景粒度的内容植入 支持场景级别的强插与置顶内

容, 满足信息流营销场景植入广告或者上热门场景, 赋能营销物料在信息流场景精准触达和有效曝光转化。 ![image.png]

(http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-

share/AI\_REC/image\_62e37bb.png) ##### 3、支持离线任务调度管理, 促进运维监控和提效 \* 任务调度引擎应用的离线任

务的计划和执行, 可以追踪任务下每个实例的运行状态, 运行实例、运行日志及任务关系, 即时捕获任务异常。 \* 避免黑盒状

态下任务执行失败到再重试过程的滞后性, 及时发现并响应异常, 显著提升系统运营效率, 增强系统的可靠性和稳定性。 \* !

![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-

cloud-share/AI\_REC/image\_21dbd4f.png) ##### 二、新版预告 \* 支持模型管理, 可选择系统预置模型和自定义深度学习模

型, 支持模型训练记录查询; \* 应用内关联多场景, 场景级分发服务, 减少未使用服务离线任务空转; \* 增强自然语言搜索功

能, 提高语义理解能力, 精准匹配并满足用户需求。 ## 智能搜推引擎V\_3.6.0版本新特性 百度智能搜推引擎V\_3.6.0版本已于

2024年10月8日上线公有云。体验产品新特性请新建应用, 届时WEB端会有5分钟左右不可用, 已有数据服务及推荐服务不受

影响。本次升级主要包括: ##### 一、推荐和搜索升级打散策略, 增加多样性策略 打散策略支持对物料自定义打散维

度和窗口大小, 防止相同物料聚集, 有效避免用户对同一类内容多次浏览, 增强了搜索结果的多样性和新鲜感; ![image.png]

(http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-

share/AI\_REC/image\_2c138a9.png) 多样性规则支持根据业务需求灵活配置物料的出现频次, 为用户提供不同类型、不同风格

的内容, 能够减少用户的审美疲劳和信息茧房效应, 使用户在探索过程中保持新鲜感和好奇心, 从而提升整体的用户体验。

##### 二、query意图理解: 搜索组件升级分层分级检索策略, 提高搜索准确度和召回率 为提升用户搜索结果丰富度和准确

性, 本次升级在混合检索框架全文检索链路新增分层分级检索策略, 根据检索数据集和词权重对query进行智能改写, 从而实现

动态寻找最优检索方法。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-

cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_3b3753d.png) \* 词权重分析: 通过分析query中每个词 (term) 的重要

程度, 权重较低的词不参与召回。避免query中权重低的词限制召回范围, 导致命中结果过少或出现不相关结果。 \* 紧密度分

析: 分析query中相邻或者相隔的几个词 (term) 的关系紧密程度, 如果词 (term) 在query中是紧密的在搜索结果中也必须是

紧密的, 该策略可以控制搜索结果的准确度。 ##### 三、新版本预告 \* 推荐支持全局定向召回和针对用户群的定向召回。 \*

新增物料集管理和用户分群运营, 满足精细化运营需求; \* 搜索支持自定义召回, 解决漏召回问题, 提升精准度; ## 智能搜

推引擎V\_3.5.0版本新特性 百度智能搜推引擎V\_3.5.0版本将于2024年8月13日上线公有云。 体验产品新特性请新建应用, 届

时WEB端会有5分钟左右不可用, 已有数据服务及推荐服务不受影响。 本次升级主要包括: ##### 一、新增优质内容等

多召回, 支持流量定向倾斜 \* 新增运营精选内容召回和基于冷启动阶段优质内容召回, 优质内容支持二次运营, 便于挖掘高

潜, 增加优质内容召回机会扩大流量, 提升平台整体收益。 ![image.png](http://bjhw-bce-online-

public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-

share/AI\_REC/image\_d613657.png) \* 个性化推荐和相关推荐支持调权重排序策略, 系统内置时效性、召回通道和指定内容集

合定向调权。简化运营流程的同时, 能够有效提升用户留存率、活跃度和转化率。 ![image.png](http://bjhw-bce-online-



public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_0e2cec0.png) ##### 二、优化兜底策略，支持多兜底融合和终极兜底，保证必出结果 为了解决部分推荐请求的结果数量不稳定的问题，通过多元化的内容融合，有效提升了推荐的广度和深度。即使在极端情况下，当常规推荐策略失效时，启动终极兜底机制确保每次请求必出结果，保障系统的稳定性和可靠性。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_8c8dde5.png) ##### 三、新增内容粒度搜索推荐调优工具，提升精细化运营力度 推荐运营支持内容粒度的内容精选和生命周期设置，可以灵活控制推优和物料退出时机，便于定向运营。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_427d07c.png) 搜索支持物料分词结果优化，人工可干预标题、描述等分词结果，快速解决badcase提升搜索结果的准确性和相关性。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_8d76382.png) ##### 四、混合检索各路召回增加阈值控制，提高搜索准确率 系统支持全文、语义和拼音混合检索，确保在海量信息中能够全面而准确地捕捉用户的搜索意图。同时对各路召回结果设定合理的阈值，以过滤掉低质量搜索结果，提高了搜索结果的准确性，减少了用户筛选信息的时间成本。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_b2bacb0.png) ##### 五、新版本预告 \* 推荐和搜索升级打散策略，增加多样性策略。 \* 提升电商场景的搜索效果，优化分词准确度； \* 新增query意图理解，升级分层分级检索策略，提高搜索准确度和召回率； \* 多租户：私有化版本支持多租户、角色及权限管理，满足央国企客户IT建设集约化路线的发展要求。 ## 智能搜推引擎V\_3.4.0版本新特性 百度智能搜推引擎V\_3.4.0版本将于2024年6月xx日封版，于2024年6月xx日上线公有云。体验产品新特性请新建应用，届时WEB端会有5分钟左右不可用，已有数据服务及推荐服务不受影响。本次升级主要包括： ##### 一、新增模型中心一级模块，推荐场景开放模型选择配置 个性化和相关推荐场景支持排序策略，可以配置选择模型排序和规则排序，模型中心上线深度学习和多目标排序等10个模型。 \* 可以根据实际需求和运营策略，精准地调整模型和规则的参数，大大提高了运营的干预度和自主性。 \* 能够显著提升推荐效果，有效提高用户活跃度和使用时长，提高业务转化率。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_a57aa4f.png) ##### 二、搜索数据飞轮新增case处理节点，支持query结果干预 支持query粒度对搜索结果中的物料进行排序置顶或屏蔽操作。 \* 能够让客户更具针对性地处理搜索结果，对特定内容增加曝光，同时解决召回结果相关性和排序不理想的问题， \* 完善了搜索效果优化工具链，快速解决线上badcase，减少线上影响。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_ff5314b.png) ##### 三、开放数据字段自定义，优化数据取值分布的可视化表现 支持数据表字段的添加、删除、修改以及数据高级设定，支持9种语言分词配置。 \* 此版本后将不再受限于固定的字段结构，满足不同业务场景下的数据接入需求，能使客户更直观地了解数据状况辅助做特征工程。 \* 帮助客户快速接入数据，提高数据处理效率，同时保持系统的灵活性和可扩展性。 ![image.png](http://bjhw-bce-online-public0.bjhw.baidu.com:8109/proxy-external/?url=http://bce-cdn.bj.bcebos.com/doc/ai-cloud-share/AI\_REC/image\_831a8ef.png) ##### 四、新版本预告 \* 支持个性化、相关推荐中优质内容召回链路的内容运营，提升优质内容曝光； \* 个性化、相关推荐后置兜底策略，支持多兜底融合和终极兜底保证必出结果。解决一次推荐请求的结果数量不稳定的问题，防止刷新多次后数据为空的问题。 \* 搜索各路召回增加阈值控制，提高搜索准确率。 \* 搜索运营支持seo工具，推荐运营支持推优和生命周期设置。