

知识引擎原子能力

DeepSeek OpenAI 兼容接口



腾讯云

【 版权声明 】

©2013–2025 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

【 商标声明 】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

【 服务声明 】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

【 联系我们 】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或 95716。

文档目录

- DeepSeek OpenAI 兼容接口
 - 腾讯云 DeepSeek OpenAI 对话接口
 - API KEY 管理

DeepSeek OpenAI 兼容接口

腾讯云 DeepSeek OpenAI 对话接口

最近更新时间：2025-05-01 23:41:42

腾讯云知识引擎原子能力 DeepSeek OpenAI 对话接口兼容了 OpenAI 的接口规范，这意味着您可以直接使用 OpenAI 官方提供的 SDK 来调用。您仅需要将 `base_url` 和 `api_key` 替换成相关配置，不需要对应用做额外修改，即可无缝将您的应用切换到相应的大模型。默认该接口下单账号限制并发数为 100 路。

- `base_url`: `https://api.lkeap.cloud.tencent.com/v1`
- `api_key`: 需在控制台 [API KEY 页面](#) 进行创建，操作步骤请参考 [API KEY 管理](#)。
- 接口请求地址完整路径: `https://api.lkeap.cloud.tencent.com/v1/chat/completions`
- 调用情况可在 [控制台](#) 中查看。计费详情请参考 [计费概述](#)。

在线体验

如您希望在网页内直接体验 DeepSeek 模型对话，推荐您前往 [腾讯云大模型知识引擎](#)，使用 [DeepSeek 联网助手](#)。

已支持的模型

模型	参数量	上下文长度	最大输出长度	默认输出长度
DeepSeek-R1	671B	64k	16k (不含思维链长度)	4k (不含思维链长度)
DeepSeek-V3	671B	64k	16k	4k
DeepSeek-V3-0324	671B	128k	16k	4k
DeepSeek-Prover-V2	671B	64k	16k	4k

DeepSeek-R1 (model 参数值为 `deepseek-r1`)

DeepSeek-R1 为 671B 模型，使用强化学习训练，推理过程包含大量反思和验证，思维链长度可达数万字。该系列模型在数学、代码以及各种复杂逻辑推理任务上推理效果优异，并为用户展现了完整的思考过程。

DeepSeek-V3 (model 参数值为 `deepseek-v3`)

DeepSeek-V3 为 671B 参数 MoE 模型，在百科知识、数学推理等多项任务上优势突出。

DeepSeek-V3-0324 (model 参数值为 `deepseek-v3-0324`)

DeepSeek-V3-0324 为671B 参数 MoE 模型，在编程与技术能力、上下文理解与长文本处理等方面优势突出。

📌 说明：

相比于 DeepSeek-V3，DeepSeek-V3-0324仅更新了模型权重，未增加参数量。总模型大小为685B，其中包括671B 的主模型权重和14B 的多令牌预测（MTP）模块权重，后续均描述主模型参数量。

DeepSeek-Prover-V2 (model 参数值为 deepseek-prover-v2)

DeepSeek-Prover-V2 为671B 参数 MoE 模型，在数学定理证明和复杂计算任务中表现出色。

快速开始

API 使用前提：已在腾讯云控制台 [API Key管理](#) 开通知识引擎原子能力并创建 API Key。如果通过 SDK 调用，需要安装 OpenAI 。

- 如果您首次使用知识引擎原子能力，请参考 [API Key管理](#) 进行知识引擎原子能力的开通，并将示例代码中的 model 参数修改为上表中您需要调用的模型名称。
- 由于 deepseek-r1 模型的思考过程可能较长，可能导致响应慢或超时，建议您优先使用**流式输出**方式调用。

安装SDK

您需要确保已安装 Python 3.8或以上版本。

安装或更新 OpenAI Python SDK

运行以下命令：

```
pip install -U openai
```

如果运行失败，请将 pip 改为pip3。

示例代码片段

非流式请求

Python

```
import os
from openai import OpenAI

client = OpenAI(
    # 请用知识引擎原子能力API Key将下行替换为: api_key="sk-xxx",
```

```
api_key="LKEAP_API_KEY", # 如何获取API Key:
https://cloud.tencent.com/document/product/1772/115970
base_url="https://api.lkeap.cloud.tencent.com/v1",
)

completion = client.chat.completions.create(
    model="deepseek-r1", # 此处以 deepseek-r1 为例, 可按需更换模型名称。
    messages=[
        {'role': 'user', 'content': '9.9和9.11谁大'}
    ]
)

# 通过reasoning_content字段打印思考过程
print("思考过程: ")
print(completion.choices[0].message.reasoning_content)
# 通过content字段打印最终答案
print("最终答案: ")
print(completion.choices[0].message.content)
```

NodeJS

```
import OpenAI from "openai";

const openai = new OpenAI(
  {
    // 请用知识引擎原子能力API Key将下行替换为: apiKey: "sk-xxx",
    apiKey: "LKEAP_API_KEY", // 如何获取API Key:
    https://cloud.tencent.com/document/product/1772/115970
    baseUrl: "https://api.lkeap.cloud.tencent.com/v1"
  }
);

const completion = await openai.chat.completions.create({
  model: "deepseek-r1", // 此处以 deepseek-r1 为例, 可按需更换模型名称。
  messages: [
    { role: "user", content: "9.9和9.11谁大" }
  ],
});

console.log("思考过程: ")
console.log(completion.choices[0].message.reasoning_content)
console.log("最终答案: ")
console.log(completion.choices[0].message.content)
```

cURL

```
curl https://api.lkeap.cloud.tencent.com/v1/chat/completions \  
-H "Content-Type: application/json" \  
-H "Authorization: Bearer sk-xxxxxxxxxxxx" \  
-d '{  
  "model": "deepseek-r1",  
  "messages": [  
    {  
      "role": "user",  
      "content": "你好"  
    }  
  ],  
  "stream": false  
'
```

多轮对话

大模型知识引擎提供的 DeepSeek API 默认不会记录您的历史对话信息。多轮对话功能可以让大模型“拥有记忆”，满足如追问、信息采集等需要连续交流的场景。如果您使用 deepseek-r1 模型，会收到 **reasoning_content** 字段（思考过程）与 **content**（回复内容），您可以将 content 字段通过 `{'role': 'assistant', 'content': API 返回的 content}` 添加到上下文，无需添加 **reasoning_content** 字段。

Python

```
import os  
from openai import OpenAI  
  
client = OpenAI(  
    # 请用知识引擎原子能力 API Key 将下行替换为: api_key="sk-xxx",  
    api_key="LKEAP_API_KEY", # 如何获取 API Key:  
    https://cloud.tencent.com/document/product/1772/115970  
    base_url="https://api.lkeap.cloud.tencent.com/v1",  
)  
  
# 通过 messages 数组实现上下文管理  
messages = [  
    {'role': 'user', 'content': '你好'},  
    {'role': 'assistant', 'content': '你好，有什么可以帮助你的?'},  
    {'role': 'user', 'content': '9.9和9.11哪个大'}  
,
```

```
]

completion = client.chat.completions.create(
    model="deepseek-r1", # 此处以 deepseek-r1 为例，可按需更换模型名称。
    messages=messages
)

print("="*20+"第一轮对话"+"="*20)
# 通过 reasoning_content 字段打印思考过程
print("="*20+"思考过程"+"="*20)
print(completion.choices[0].message.reasoning_content)
# 通过 content 字段打印最终答案
print("="*20+"最终答案"+"="*20)
print(completion.choices[0].message.content)

messages.append({'role': 'assistant', 'content':
completion.choices[0].message.content})
messages.append({'role': 'user', 'content': '你是谁'})
print("="*20+"第二轮对话"+"="*20)
completion = client.chat.completions.create(
    model="deepseek-r1", # 此处以 deepseek-r1 为例，可按需更换模型名称。
    messages=messages
)
# 通过 reasoning_content 字段打印思考过程
print("="*20+"思考过程"+"="*20)
print(completion.choices[0].message.reasoning_content)
# 通过 content 字段打印最终答案
print("="*20+"最终答案"+"="*20)
print(completion.choices[0].message.content)
```

NodeJS

```
import OpenAI from "openai";

const openai = new OpenAI(
    {
        // 请用知识引擎原子能力API Key将下行替换为: apiKey: "sk-xxx",
        apiKey: "LKEAP_API_KEY", // 如何获取API Key:
https://cloud.tencent.com/document/product/1772/115970
        baseUrl: "https://api.lkeap.cloud.tencent.com/v1"
    }
);
```

```
const completion = await openai.chat.completions.create({
  model: "deepseek-r1", // 此处以 deepseek-r1 为例，可按需更换模型名称。
  messages: [
    { role: "user", content: "你好"},
    { role: "assistant", content: "你好，有什么可以帮助你的？"},
    { role: "user", content: "9.9和9.11谁大" }
  ],
});
console.log("思考过程：")
console.log(completion.choices[0].message.reasoning_content)
console.log("最终答案：")
console.log(completion.choices[0].message.content)
```

cURL

```
curl https://api.lkeap.cloud.tencent.com/v1/chat/completions \
-H "Content-Type: application/json" \
-H "Authorization: Bearer sk-xxxxxxxxxxxx" \
-d '{
  "model": "deepseek-r1",
  "messages": [
    {
      "role": "user",
      "content": "你好"
    },
    {
      "role": "assistant",
      "content": "你好，有什么可以帮助你的？"
    },
    {
      "role": "user",
      "content": "9.9和9.11谁大"
    }
  ],
  "stream": true
}'
```

流式输出

deepseek-r1 模型可能会输出较长的思考过程，为了降低超时风险，建议您使用流式输出方式调用 deepseek-r1 模型。

Python

```
from openai import OpenAI
import os

# 初始化OpenAI客户端
client = OpenAI(
    # 请用知识引擎原子能力API Key将下行替换为: api_key="sk-xxx",
    api_key="LKEAP_API_KEY", # 如何获取API Key:
    https://cloud.tencent.com/document/product/1772/115970
    base_url="https://api.lkeap.cloud.tencent.com/v1",
)

def main():
    reasoning_content = "" # 定义完整思考过程
    answer_content = "" # 定义完整回复
    is_answering = False # 判断是否结束思考过程并开始回复

    # 创建聊天完成请求
    stream = client.chat.completions.create(
        model="deepseek-r1", # 此处以 deepseek-r1 为例，可按需更换模型名称
        messages=[
            {"role": "user", "content": "9.9和9.11谁大"}
        ],
        stream=True
    )

    print("\n" + "=" * 20 + "思考过程" + "=" * 20 + "\n")

    for chunk in stream:
        # 处理usage信息
        if not getattr(chunk, 'choices', None):
            print("\n" + "=" * 20 + "Token 使用情况" + "=" * 20 + "\n")
            print(chunk.usage)
            continue

        delta = chunk.choices[0].delta

        # 处理空内容情况
```

```
    if not getattr(delta, 'reasoning_content', None) and not
getattr(delta, 'content', None):
        continue

    # 处理开始回答的情况
    if not getattr(delta, 'reasoning_content', None) and not
is_answering:
        print("\n" + "=" * 20 + "完整回复" + "=" * 20 + "\n")
        is_answering = True

    # 处理思考过程
    if getattr(delta, 'reasoning_content', None):
        print(delta.reasoning_content, end='', flush=True)
        reasoning_content += delta.reasoning_content

    # 处理回复内容
    elif getattr(delta, 'content', None):
        print(delta.content, end='', flush=True)
        answer_content += delta.content

if __name__ == "__main__":
    try:
        main()
    except Exception as e:
        print(f"发生错误: {e}")
```

NodeJS

```
import OpenAI from "openai";

const openai = new OpenAI({
    // 请用知识引擎原子能力API Key将下行替换为: apiKey: "sk-xxx",
    apiKey: "LKEAP_API_KEY", //如何获取API Key:
https://cloud.tencent.com/document/product/1772/115970
    baseURL: "https://api.lkeap.cloud.tencent.com/v1"
});

async function main() {
    let reasoningContent = ""; // 定义完整思考过程
    let answerContent = ""; // 定义完整回复
    let isAnswering = false; // 判断是否结束思考过程并开始回复
```

```
const completion = await openai.chat.completions.create({
  model: "deepseek-r1", // 此处以 deepseek-r1 为例，可按需更换模型名称
  messages: [
    { role: 'user', content: '9.9和9.11谁大' }
  ],
  stream: true,
});

console.log("\n" + "=".repeat(20) + "思考过程" + "=".repeat(20) +
"\n");

for await (const chunk of completion) {
  // 处理usage信息
  if (!chunk.choices?.length) {
    console.log("\n" + "=".repeat(20) + "Token 使用情况" +
"=".repeat(20) + "\n");
    console.log(chunk.usage);
    continue;
  }

  const delta = chunk.choices[0].delta;

  // 处理空内容情况
  if (!delta.reasoning_content && !delta.content) {
    continue;
  }

  // 处理开始回答的情况
  if (!delta.reasoning_content && !isAnswering) {
    console.log("\n" + "=".repeat(20) + "完整回复" +
"=".repeat(20) + "\n");
    isAnswering = true;
  }

  // 处理思考过程
  if (delta.reasoning_content) {
    process.stdout.write(delta.reasoning_content);
    reasoningContent += delta.reasoning_content;
  }

  // 处理回复内容
  else if (delta.content) {
    process.stdout.write(delta.content);
    answerContent += delta.content;
  }
}
```

```
    }
  }
}

main().catch(console.error);
```

cURL

```
curl https://api.lkeap.cloud.tencent.com/v1/chat/completions \
-H "Content-Type: application/json" \
-H "Authorization: Bearer sk-xxxxxxxxxxxx" \
-d '{
  "model": "deepseek-r1",
  "messages": [
    {
      "role": "user",
      "content": "你好"
    }
  ],
  "stream": true
}'
```

注意事项

稳定性

若执行后出现“concurrency exceeded”的响应，则表明您的请求遭遇了限流。这通常是由于服务器资源暂时不足所致。建议您稍后再试，届时服务器负载可能已得到缓解。

DeepSeek-R1

- 不支持设置的参数和功能
 - Function Calling、JSON Output、对话前缀续写、上下文硬盘缓存。
- 不支持的参数
 - presence_penalty、frequency_penalty、logprobs、top_logprobs。
- 支持的参数
 - top_p、temperature、max_tokens。
- 参数默认值：
 - temperature: 0.6 (取值范围是[0:2])

- top_p: 0.6 (取值范围是(0:1])
- 不建议设置 System Prompt (来自官方说明)。

DeepSeek-V3

- **不支持设置的参数和功能**
 - Function Calling、JSON Output、对话前缀续写、上下文硬盘缓存。
- **不支持的参数**
 - presence_penalty、frequency_penalty、logprobs、top_logprobs。
- **支持的参数**
 - top_p、temperature、max_tokens。
- **参数默认值:**
 - temperature: 0.6 (取值范围是[0:2])
 - top_p: 0.6 (取值范围是(0:1])

敬请关注后续动态。

联网搜索

本接口支持联网搜索，前提是开启 DeepSeek API 的服务。若需要使用相关联网搜索功能，首先需要开启联网搜索后付费开关或者购买搜索服务的资源包，然后通过传入 enable_search 参数体验 DeepSeek 的联网搜索能力。

1. 首先开启 DeepSeek API 的服务，进入 [原子能力控制台](#) 开启 DeepSeek 的后付费开关。

后付费设置

1. 开启后，资源包耗尽时，以“后付费方式”按小时从账户余额中扣费，欠费时无法使用，[查看账户余额](#)（各开关每小时可修改1次）
 2. 扣费顺序：体验资源包>付费资源包>后付费，可在[控制台-费用中心](#) [查看账单](#)

子产品名称	启用状态	生效状态	操作
原子能力_Embedding后付费	<input type="checkbox"/>	未启用	-
原子能力_多轮改写后付费	<input type="checkbox"/>	未启用	-
原子能力_Rerank后付费	<input type="checkbox"/>	未启用	-
原子能力_文档解析后付费	<input type="checkbox"/>	未启用	-
原子能力_RAG_知识库检索后付费	<input type="checkbox"/>	未启用	-
原子能力_文档拆分后付费	<input type="checkbox"/>	未启用	-
原子能力_DeepSeek API-R1后付费	<input checked="" type="checkbox"/>	已生效	-
原子能力_DeepSeek API-V3后付费	<input checked="" type="checkbox"/>	已生效	-
原子能力_搜索服务后付费	<input checked="" type="checkbox"/>	已生效	-

2. 进入 [原子能力控制台](#) 打开联网搜索后付费开关或者购买搜索服务的资源包。

开启搜索服务的后付费开关：

腾讯云 控制台
支持通过实例ID、IP、名称等搜索资源
快捷键 /
集团账号 备案 工具 客服支持 费用
1

知识引擎原子能力

- DeepSeek API
 - 立即接入
 - API key 管理
- RAG原子能力
 - 在线体验
 - 立即接入
- 系统管理
 - 调用明细
 - 资源包管理
 - 设置
- 常用工具
 - API Explorer
 - 自助诊断

设置

后付费设置

1. 开启后，资源包耗尽时，以“后付费方式”按小时从账户余额中扣费，欠费时无法使用，[查看账户余额](#)（各开关每小时可修改1次）

2. 扣费顺序：体验资源包>付费资源包>后付费，可在[控制台-费用中心](#)查看账单

子产品名称	启用状态	生效状态	操作
原子能力_Embedding后付费	<input type="checkbox"/>	未启用	-
原子能力_多轮改写后付费	<input type="checkbox"/>	未启用	-
原子能力_Rerank后付费	<input type="checkbox"/>	未启用	-
原子能力_文档解析后付费	<input type="checkbox"/>	未启用	-
原子能力_RAG_知识库检索后付费	<input type="checkbox"/>	未启用	-
原子能力_文档拆分后付费	<input type="checkbox"/>	未启用	-
原子能力_DeepSeek API-R1后付费	<input checked="" type="checkbox"/>	已生效	-
原子能力_DeepSeek API-V3后付费	<input checked="" type="checkbox"/>	已生效	-
原子能力_搜索服务后付费	<input checked="" type="checkbox"/>	已生效	-

购买搜索服务的资源包：

腾讯云 | 选购其他云产品
搜索
备案 | Alberthch...
控制台

知识引擎原子能力

返回产品详情
产品文档

购买须知

- 购买说明** 根据您的需要购买对应的原子能力套餐包，对话原子能力专属并发购买前需要联系对接的客户经理确认资源部署情况，如无对接的客户经理可联系[售前咨询](#)。详情参考[计费概述](#)
- 使用说明** 购买完成后资源为生效状态，通过调用API使用该产品
- 退费说明** 购买前请详细了解[退费说明](#) 腾讯云有权根据退费说明拒绝相应退款要求

选择配置

采购类型

DeepSeek API 新

多轮改写

Embedding

文档解析

Rerank

RAG套件

资源类型

搜索服务包

可选套餐包

<p>10万次</p> <p style="font-size: 8px;">有效期自购买之日一年内</p>	<div style="display: flex; align-items: center; gap: 5px;"> - 0 + </div>
<p>100万次</p> <p style="font-size: 8px;">有效期自购买之日一年内</p>	<div style="display: flex; align-items: center; gap: 5px;"> - 0 + </div>
<p>1000万次</p>	<div style="display: flex; align-items: center; gap: 5px;"> - 0 + </div>

配置费用 0.00元

立即购买

联系
销售

3. 调用 DeepSeek API 接口的时候，增加 enable_search 参数体验带联网搜索能力的 DeepSeek API。
 输入示例：

```

client = OpenAI(
# 请用知识引擎原子能力API Key将下行替换为: api_key="sk-xxx",
api_key="LKEAP_API_KEY", # 如何获取API Key:
https://cloud.tencent.com/document/product/1772/115970
base_url="https://api.lkeap.cloud.tencent.com/v1",)

completion = client.chat.completions.create(
    model="deepseek-v3", # 此处以deepseek-r1为例，可按需更换模型名称。
    messages=[
        {
            "role": "user",
            "content": "深圳今日天气"
        }
    ]
    )
    
```

```
],  
  extra_body={  
    "enable_search": True, # 开启联网搜索  
  }  
)
```

输出示例:

```
{'id': '0c9eba283a56d0add6084e24263056b7', 'choices': [{'finish_reason':  
'stop', 'index': 0, 'logprobs': None, 'message': {'content': '根据深圳市气  
象局(台)的预报, 2025年4月29日深圳的天气情况如下: \n\n今天深圳多云间阴天, 早晚有轻  
雾, 部分时间可见阳光, 偏东风最大阵风6-7级, 气温22-29℃。相对湿度在40%-70%之间。 \n\n  
具体来说, 今天的气温预计在22℃到29℃之间, 风向为偏东风, 风力为2-3级, 沿海、高地和海区阵  
风可达5-6级。此外, 今天日出时间为05:52, 日落时间为18:49。 \n\n需要注意的是, 今天早晚  
有轻雾, 部分时间可见阳光, 建议关注清劲偏东风的影响。 \n\n总结: 今天深圳的天气以多云间阴  
天为主, 气温适中, 早晚有轻雾, 风力较大, 适合外出但需注意防风。', 'refusal': None,  
'role': 'assistant', 'annotations': None, 'audio': None,  
'function_call': None, 'tool_calls': None, 'search_results': [{'index':  
1, 'url': 'http://weather.sz.gov.cn/?COLLCC=2354144265&', 'name': '深圳市  
气象局(台)', 'snippet': '深圳市气象局(台) 今日预报 实况 29.4℃ 12-20时 多云;气温  
27-30℃;东风2-3级,沿海、高地和海区阵风5-6级;相对湿度40%-70%。 05:52 日出 18:49 日  
落 6天 距立夏剩 06:40 月出 20:41 月落 4月29日12时56分 深圳福田国家基本气象台 东南  
偏东风 小于三级 相对湿度 0mm 24小时降雨量 十天预报 逐时预报 展开 【天气提示】 预计29  
日多云间阴天,早晚有轻雾,部分时间可见阳光,偏东风最大阵风6-7级,气温22-29℃;30日多云到阴  
天,局地有短时阵雨,早晚清凉;展望五一假期,初期和末期有(雷)阵雨,局地雨势较大,中期以多云  
为主,间中有短时阵雨,午间较热。建议关注29日清劲偏东风的影响。 明天 4-30 周四 5-1 周五  
5-2 周六 5-3 周日 5-4 周一 5-5 周二 5-6 周三 5-7 周四 5-8 13时 多云 14时 多云  
15时 多云 16时 多云 17时 多云 18时 多云 19时 多云 20时 多云 21时 少云 热点推荐  
工作动态 公告公示 重要资讯 公开目录 媒体聚焦 庆祝中华全国总工会成立100周年暨全国劳动  
模范和先进工作者表彰大会隆重举行 习近平发表重要讲话 04/29 广交会上看外贸新动能 04/29  
新,高质量发展看动能 04/29 一季度规模以上工业企业利润由降转增 04/29 暴涨96%!一季  
度“中国游 中国购”持续升温 04/29 李强主持召开国务院常务会议 部署开展美丽河湖保护与建设  
行动 研究进一步加强困境儿童福利保障有关举措 讨论《中华人民共和国医疗保障法(草案)》 决  
定核准浙江三门三期工程等核电项目 04/28 经济日报:消费市场保持升温势头|中国经济新看点  
04/27 规范涉企执法,怎样防止问题反弹、提振企业信心?', 'icon': '', 'site':  
'weather.sz.gov.cn', 'published_time': 1745856000}}]}], 'created':  
1745929940, 'model': 'deepseek-v3-aisearch', 'object':  
'chat.completion', 'service_tier': None, 'system_fingerprint': None,  
'usage': {'completion_tokens': 189, 'prompt_tokens': 3366,  
'total_tokens': 3555, 'completion_tokens_details': None,  
'prompt_tokens_details': None}}
```

深度思考

只要调用 `deepseek-r1` 模型即代表开启深度思考（深度思考过程通过 `reasoning_content` 返回）。

查看更多完整示例

[Python](#)

[NodeJS](#)

[Go](#)

错误码

错误码	错误信息	说明
20031	not enough quota	您的账号目前没有可用资源。为了继续使用，请先 开通 并完成付费。
20034	concurrency exceeded	您的请求遭遇了限流。这通常是由于服务器资源暂时不足所致。建议您稍后再试，届时服务器负载可能已得到缓解。
20059	input content too long	输入长度超过上下文长度，请减小输入内容的长度

错误示例

```
{"error":{"message":"not enough quota","type":"runtime_error","param":null,"code":"20031"}}
```

安全审查示例

`finish_reason = content_filter` 表示输出内容触发了安全审核机制。这通常发生在系统检测到某些输入或输出内容可能包含敏感信息或不适当的语言，因此自动启动了审核流程以确保内容的安全性和适宜性。在这种情况下，系统会对相关内容进行仔细审查，以防止不当信息的传播。

非流式输出示例：

```
{"id":"26a58a8ab6e7712937ad542436b4b97a","object":"chat.completion","created":1740379897,"model":"deepseek-r1","choices":[{"index":0,"message":{"role":"assistant","content":"你好，我无法给到相关内容。"},"finish_reason":"content_filter"}],"usage":{"prompt_tokens":0,"completion_tokens":0,"total_tokens":0}}
```

流式输出示例：

```
data:
{"id":"d2d486bfdb31b1b6f55c8b5cbeb492d3","object":"chat.completion.chunk",
,"created":1740379627,"model":"deepseek-r1","choices":
[{"index":0,"delta":{"role":"assistant","content":"你好，我无法给到相关内
容。"},"finish_reason":"content_filter"}],"usage":
{"prompt_tokens":0,"completion_tokens":0,"total_tokens":0}}
```

API KEY 管理

最近更新时间：2025-02-06 00:49:42

⚠ 注意：

需要主账号才能进行创建、删除操作，API KEY 删除后将无法恢复。

步骤1：登录主账号

注册并通过个人实名认证或企业认证后，登录 [腾讯云](#)。如果没有账号，请参考 [注册腾讯云](#)。

步骤2：开通服务

知识引擎原子能力大模型对话 API 已对外开放，可前往 [控制台](#) 开通服务。

步骤3：管理 API

进入 [控制台](#) > [立即接入](#) 管理，单击 [创建 API KEY](#)。

知识引擎原子能力 立即接入

使用腾讯云SDK方式接入

- 1 创建密钥**
进入API密钥管理界面，点击新建密钥，即可生成API/SDK调用所需的签名 APPID、SecretId与SecretKey信息。
[创建密钥](#)
- 2 选择接入方式**
通过API接入 [查看文档](#)
- 3 快速调试**
API Explorer 提供了在线调用、签名验证、SDK代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成SDK调用示例。
[点击调试](#)

使用OpenAI SDK方式接入

- 1 创建API KEY**
进入兼容OpenAI API KEY的创建页面，点击新建即可生成API KEY
[创建API KEY](#)
- 2 选择接入方式**
按照兼容OpenAI的方式，通过API接入 [查看文档](#)

创建完成后，进入 [API KEY 管理](#)，进行新增、查看、删除操作。

知识引擎原子能力

立即接入

数据报表

解析拆分demo

资源包管理

设置

API KEY管理

常用工具

API KEY管理

• API KEY是大模型API请求时的安全鉴权凭证。为了您的财产和服务安全，请妥善保管API KEY，请勿通过任何方式上传或则公开分享您的密钥信息。

创建API KEY

API KEY	创建时间	最后使用时间	操作
暂无数据			

