

# 知识库开放api

🐼 open\_api接口和前端应用接口略有不同，需要重新实现接口,主要是入参和返回数据的变化，内部逻辑不变

baseURL: <https://open.bigmodel.cn/api/llm-application/open>

## 统一鉴权

<https://open.bigmodel.cn/dev/api#nosdk>

## SDK调用

暂无

## HTTP调用

## 统一JSON返回格式

参数名称	类型	参数说明
data	object	接口响应数据
code	int	错误码，200为正常
message	string	错误信息

## 统一基础结构体

embeddingData 模型信息

参数名称	类型	参数说明
name	string	模型名称
description	string	模型描述

id	int	模型id
----	-----	------

knowledgeData 知识库数据

参数名称	类型	参数说明
id	string	知识库唯一id
embedding_id	int	知识库绑定的向量化模型
name	string	知识库名称
description	string	知识库描述
background	string	背景颜色
icon	string	知识库图标
word_num	int	知识库总字数
length	int	知识库总大小（字节）
document_size	int	知识文件数量

uploadDetail 上传详情

参数名称	类型	参数说明
key	string	上传文档接口返回的key
file_name	string	文档名称

documentData 知识数据

参数名称	类型	参数说明
id	string	知识唯一id
custom_separator	string	切片规则
sentence_size	string	切片大小
length	int	文件大小（字节）
word_num	int	文件字数

name	string	文件名
url	string	文件下载链接
embedding_stat	int	0:向量化中 1:向量化完成 2:向量化失败
failInfo	object	失败原因 向量化失败embedding_stat=2的时候 会有此值

failInfo:

参数名称	类型	参数说明
embedding_code	int	失败码
embedding_msg	string	失败原因

applicationData 应用数据

参数名称	类型	参数说明
id	string	应用唯一id
name	string	应用名称
desc	string	应用描述
prompt	string	模版,格式为 “ {{知识}}xxx{{用户}} ”,必须包含知识和用户, 且占位符 {{知识}} 有且只有一个, 占位符 {{用户}} 有且只有一个
top_p	string	默认0.7, 用温度取样的另一种方法, 称为核取样 取值范围是: (0.0, 1.0) 开区间, 不能等于 0 或 1, 默认值为 0.7 模型考虑具有 top_p 概率质量tokens的结果 例如: 0.1 意味着模型解码器只考虑从前 10% 的概率的候选集中取 tokens 建议您根据应用场景调整 top_p 或 temperature 参数, 但不要同时调整两个参数
temperature	string	默认0.95, 采样温度, 控制输出的随机性, 必须为正数 取值范围是: (0.0,1.0] , 不能等于 0,默认值为 0.95 值越大, 会使输出更随机, 更具创造性; 值越小, 输出会更加稳定或确定 建议您根据应用场景调整 top_p 或 temperature 参数, 但不要同时调整两个参数
knowledge_ids	list<Long>	知识库id列表

slice_count	int	分片数量, 默认是1
model	enum	取值范围: chatglm_pro,chatglm_std, chatglm_lite,chatglm_lite_32k
icon_color	enum	取值范围: blue, red, orange, purple, sky
icon_type	enum	取值范围: robot_assist、robot_navigate、robot_guardian、robot_service、robot_messenger、robot_chat、robot_engineer

## 向量化状态 status

状态枚举值:int	参数说明
0	向量化过程中
1	向量化成功
2	向量化失败

## 向量化失败状态码

状态枚举值:int	参数说明
10001	知识不可用, 知识库空间已达上限
10002	知识不可用, 知识库空间已达上限(字数超出限制)

## 知识类型 d\_type

枚举值:int	参数说明
1	txt
3	pdf
4	url
5	docx
6	xlsx

# 向量相关

## 向量模型列表

用于获取当前支持的向量模型列表

### 接口请求

传输方式	https
请求地址	/embedding
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	
响应格式	JSON
接口请求类型	GET

### 接口请求参数

参数名称	类型	是否必填	参数说明
------	----	------	------

### 请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-application/open/embedding' \  
2 --header 'accept: */*' \  
3 --header 'Authorization: eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfaWQiOiJlLCJjdXN0b21lc2VzZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEt0TkxMi00OTE0NTU5NGU1MDA1LCJjZXR5bmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2VZz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg'
```

### 接口响应参数


参数名称	类型	参数说明
list	list<embeddingData>	模型列表

total	int	模型总数量
-------	-----	-------

响应示例

```
1 {
2   "data": {
3     "list": [
4       {
5         "name": "zhipu_embedding_v1",
6         "description": "zhipu_embedding_v1",
7         "id": 3
8       }
9     ],
10    "total": 1
11  },
12  "code": 200,
13  "message": "请求成功",
14  "timestamp": 1689649504996
15 }
16
```

知识库

 知识库可以不依赖于应用单独存在

创建知识库

用于创建个人知识库

接口请求

传输方式	https
请求地址	/knowledge
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	JSON

响应格式	JSON
接口请求类型	POST

接口请求参数

参数名称	类型	是否必填	参数说明
embedding_id	int	是	知识库绑定的向量化模型
name	string	是	知识库名称
description	string	否	知识库描述
background	string	否（默认）blue	背景颜色（给枚举）'blue','red','orange','purple','sky'
icon	string	否（默认）question	知识库图标（给枚举） question :问号、 book :书籍、 seal :印章、 wrench :扳手、 tag :标签、 horn :喇叭、 house :房子

请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-application/open/knowledge' \
2 --header 'accept: */*' \
3 --header 'Content-Type: application/json' \
4 --header 'Authorization:
eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfaWQiOiE2MzU4NiwiZXNlcl9rZXkiOiJlZTFhNTAwNC05MDZiLTQ4OTEtOTkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc21pZCI6Ijc2OTk3NTcwOTMyNzA0MjUzZmJQIjE2VybWZtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2VZz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg' \
5 --data '{
6   "embedding_id": 1,
7   "name": "测试知识库",
8   "description": "用于测试"
9 }'
```

## 接口响应参数

参数名称	类型	参数说明
id	string	生成的知识库唯一id

## 响应示例

```
1 {
2   "data": {
3     "id": "1734786514366738432"
4   },
5   "code": 200,
6   "message": "请求成功",
7   "timestamp": 1702440310459
8 }
9
```

## 编辑知识库

用于编辑已经创建好的个人知识库

## 接口请求

传输方式	https
请求地址	/knowledge/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	PATH
响应格式	JSON
接口请求类型	PUT

## 接口路径参数

参数名称	类型	是否必填	参数说明



id	string	是	知识库id
----	--------	---	-------

## 接口请求参数

仅传入要修改的字段即可

参数名称	类型	是否必填	参数说明
embedding_id	int	否	知识库绑定的向量化模型
name	string	否	知识库名称
description	string	否	知识库描述
background	string	否	背景颜色
icon	string	否	知识库图标

## 请求示例

```
1 curl --location --request PUT 'https://open.bigmodel.cn/api/llm-
  application/open/knowledge/1734786514366738432' \
2 --header 'accept: */*' \
3 --header 'Content-Type: application/json' \
4 --header 'Authorization:
  eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiaWJQRVJTT05BTCIsInVzZXJfaWQiOiE2MzU4NiwiZX
  Nlcll9rZXkiOiJlZTFlNTAwNC05MDZiLTQ4OTEtOTkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc
  l9pZ
  CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQ1LCJlc2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
  Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg' \
5 --data '{
6   "name": "test"
7 }'
```

## 接口响应参数

### 响应示例

```
1 {
2   "code": 200,
3   "message": "请求成功",
4   "timestamp": 1689649504996
5 }
6
```

# 知识库列表

获取个人知识库列表

## 接口请求

传输方式	https
请求地址	/knowledge
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON
接口请求类型	GET

## 接口请求参数

参数名称	类型	是否必填	参数说明
page	int	否	页码，默认 1
size	int	否	每页数量 默认10

## 请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-application/open/knowledge' \  
2 --header 'accept: */*' \  
3 --header 'Authorization: eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfaWQiOiJlcl9rZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEt0TkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc1pZCI6Ijc2OTk3NTcwOTMyNzA0MjUzMjQ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2VZz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg'
```

## 接口响应参数

参数名称	类型	参数说明

list	list<knowledgeData>	知识库列表
total	int	知识库总数量

响应示例

```
1 {
2   "data": {
3     "list": [
4       {
5         "id": "12121212111212",
6         "embedding_id": 1,
7         "name": "",
8         "description": "",
9         "background": "",
10        "icon": "",
11        "document_size": 10,
12        "length":10000,
13        "word_num":100000
14      },
15    ],
16    "total": 1
17  },
18  "code": 200,
19  "message": "请求成功",
20  "timestamp": 1689649504996
21 }
```

知识库详情

获取个人知识库详情

接口请求

传输方式	https
请求地址	/knowledge/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	PATH
响应格式	JSON

接口请求类型	GET
--------	-----

接口路径参数

参数名称	类型	是否必填	参数说明
id	string	是	根据知识库id

请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-
  application/open/knowledge/1734786514366738432' \
2 --header 'accept: */*' \
3 --header 'Authorization:
eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOiJlMzU4NiwiZmZlcnRlZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEtOTkxMi00OTE0NTU5NGU1MDAiLCJjdXN0b21lc1pZ
CI6Ijc2OTk3NTcwOTMyZnZAMjUzZmJjLCJlc2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg'
```

接口响应参数参考

knowledgeData

响应示例

```
1 {
2   "data": {
3     "id": "1734786514366738432",
4     "name": "test",
5     "description": "用于测试",
6     "icon": "question",
7     "background": "blue",
8     "embedding_id": 1,
9     "word_num": 0,
10    "length": 0,
11    "document_size": 0
12  },
13  "code": 200,
14  "message": "请求成功",
15  "timestamp": 1702440707110
16 }
```

# 删除知识库

用于删除知识库

## 接口请求

传输方式	https
请求地址	/knowledge/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	PATH
响应格式	JSON
接口请求类型	DELETE

## 接口路径参数

参数名称	类型	是否必填	参数说明
id	string	是	知识库id

## 请求示例

```
1 curl --location --request DELETE 'https://open.bigmodel.cn/api/llm-
  application/open/knowledge/1734786514366738432' \
2 --header 'accept: */*' \
3 --header 'Authorization:
eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOjE2MzU4NiwiZ
Nlcll9rZXkiOiJlZTFNTAwNC05MDZiLTQ4OTEtOTkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc
l9pZCI6Ijc2OTk3NTcwOTMyNzA0MjUzMjQlLCJlc2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufX
sv0vR2VZz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYh
QyS7Cg'
```

## 接口响应参数

参数名称	类型	参数说明
------	----	------

## 响应示例

```
1 {
2   "code": 200,
3   "message": "请求成功",
4   "timestamp": 1689649504996
5 }
```

知识库使用量详情

接口请求

传输方式	https
请求地址	/knowledge/capacity
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	
响应格式	JSON
接口请求类型	GET

接口请求参数

参数名称	类型	是否必填	参数说明
------	----	------	------

请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-
  application/open/knowledge/capacity' \
2 --header 'accept: */*' \
3 --header 'Authorization:
  eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOiJlMzU4NiwiZXNlcl9rZXkiOiJlZTFNTAwNC05MDZiLTQ0TEt0TkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc19pZCI6Ijc2OTk3NTcwOTMyNzA0MjUzMjQiLCJ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2VZz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg'
```

接口响应参数

--	--	--

参数名称	类型	参数说明
used	object<T>	已使用量
total	object<T>	总量

T:


参数名称	类型	参数说明
word_num	integer	字数
length	integer	字节数

示例

```
1 {
2   "data":{
3     "used":{
4       "word_num":100,
5       "length":2000
6     },
7     "total":{
8       "word_num":100000,
9       "length":200000000
10    }
11  },
12  "code":200,
13  "message":"请求成功",
14  "timestamp":1689649504996
15 }
16
```

知识

提供按照切片删除的接口,增加切片大类

 知识依托于知识库存在，在创建知识之前必须有指定知识库

上传文档构建知识

使用备注:

- 1

@param id 知识库sid
- 2

@param files 文件数组
- 3

@param knowledge\_type 知识类型不传则动态解析,传了则按照传参进行解析
- 4

@param custom\_separator 知识类型为自定义情况下 可传分隔符, 默认\n,多个逗号分割或者多个k
- 5

@param sentence\_size 知识类型为自定义情况下 可传切片 ,默认300

接口请求

传输方式	https
请求地址	/document/upload_document/{id}
调用方式	同步调用, 等待返回结果
字符编码	UTF-8
接口请求格式	multipart/form-data
响应格式	JSON
接口请求类型	POST

接口路径参数

参数名称	类型	是否必填	参数说明
id	string	是	知识库id

接口请求参数

参数名称	类型	是否必填	参数说明
files	文件数组	是	文件
knowledge_type	int	否	不传: 会动态解析知识类型 1:文章知识: 支持pdf,url,docx 2.问答知识-文档: 支持pdf,url,docx 3.问答知识-表格: 支持xlsx



			4.商品库-表格: 支持xlsx 5.自定义: 支持pdf,url,docx
custom_separator	List<string>	否	当前知识类型为自定义(konwledge_type=5)的时候传 切片规则 默认\n
sentence_size	int	否	当前知识类型为自定义(konwledge_type=5)的时候传,取值范围: 20-2000,切片大小 默认300

请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-  
application/open/document/upload_document/1734786402936664064' \  
2 --header 'accept: */*' \  
3 --header 'Authorization:  
eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJT05BTCIsInVzZXJfaWQiOiJlLC9rZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEtOTkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc19pZCI6IjEjZ20Tk3NTcwOTMyNzA0MjUzMjQjLCJ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2VZz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHnzYhQyS7Cg' \  
4 --form 'files=@"/Users/liu/Downloads/xxx.pdf"' \  
5 --form 'files=@"/Users/liu/Downloads/xxx.docx"'
```

接口响应参数

参数名称	类型	参数说明
successInfo	list<T>	上传成功的文件
failedInfos	list<R>	上传失败的文件

T:

参数名称	类型	参数说明
documentId	int	知识id
fileName	string	文件名

R:

参数名称	类型	参数说明
fileName	string	文件名
failReason	string	失败原因

示例

```
1 {
2   "data": {
3     "successInfos": [
4       {
5         "documentId": "123456788990",
6         "fileName": "xxx.pdf"
7       },
8       {
9         "documentId": "123456788",
10        "fileName": "xxx.docx"
11      }
12    ],
13    "failedInfos": [
14      {
15        "fileName": "xxx.xlsx",
16        "failReason": "不支持的文档类型"
17      }
18    ]
19  },
20  "code": 200,
21  "message": "请求成功",
22  "timestamp": 1690355734843
23 }
```

编辑知识

用于编辑个人知识库

接口请求

传输方式	https
请求地址	/document/{id}
调用方式	同步调用，等待返回结果

字符编码	UTF-8
接口请求格式	PUT
响应格式	JSON
接口请求类型	PUT

接口路径参数

参数名称	类型	是否必填	参数说明
id	string	是	知识id

接口请求参数

参数名称	类型	是否必填	参数说明
knowledge_type	int	是	知识类型: 1:文章知识: 支持pdf,url,docx 2.问答知识-文档: 支持pdf,url,docx 3.问答知识-表格: 支持xlsx 4.商品库-表格: 支持xlsx 5.自定义: 支持pdf,url,docx
custom_separator	List<string>	否	当前知识类型为自定义(knowledge_type=5)的时候传 切片规则才会生效, 默认\n
sentence_size	int	否	当前知识类型为自定义(knowledge_type=5)的时候传切片字数才会生效,取值范围: 20-2000,切片大小默认300

请求示例

```
1 curl --location --request PUT 'https://open.bigmodel.cn/api/llm-application/open/document/1734816157446680576' \  
2 --header 'accept: */*' \  
3 --header 'Authorization: eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOiJlE2MzU4NiwiOiJlcll9rZXkiOiJlZTFNTAwNC05MDZiLTQ0TEt0TkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc9pZ'
```

```
CI6Ijc20Tk3NTcw0TMYNzA0MjUzMjQiLCJ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
Zz132Gaw0GHG0x-sE0DD_cCfKfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg' \
4 --header 'Content-Type: application/json' \
5 --data '{
6     "knowledge_type": 1
7 }'
```

接口响应参数

响应示例

```
1 {
2     "code": 200,
3     "message": "请求成功",
4     "timestamp": 1689649504996
5 }
6
```

知识列表

某个知识库的知识列表

接口请求

传输方式	https
请求地址	/document
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	QUERY
响应格式	JSON
接口请求类型	GET

接口请求参数

参数名称	类型	是否必填	参数说明
knowledge_id	string	是	知识库id

page	int	否	页 默认 1
size	int	否	每页数量 默认10

请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-application/open/document?
knowledge_id=1734786402936664064' \
2 --header 'accept: */*' \
3 --header 'Authorization:
eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOiJlE2MzU4NiwiZm
Nlc19rZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEt0TkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc19pZ
CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQlLCJ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg'
```

接口响应参数

参数名称	类型	参数说明
list	list<documentData>	模型列表
total	long	模型总数量

响应示例

```
1 {
2   "data": {
3     "list": [
4       {
5         "id": "12312121212",
6         "custom_separator": "\n",
7         "sentence_size": 300,
8         "length": 0,
9         "word_num": 100,
10        "name": "",
11        "url": "",
12        "embedding_stat": 0,
13        "failInfo": {
14          "embedding_code": 10002, // 见统一基础结构体：向量化失败状态码
15          "embedding_msg": "字数超出限制"
16        }
17      }
18    ],
```

```
19         "total": 1
20     },
21     "code": 200,
22     "message": "请求成功",
23     "timestamp": 1689649504996
24 }
```

知识详情

根据id获取知识详情

接口请求

传输方式	https
请求地址	/document/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON
接口请求类型	GET

接口路径参数

参数名称	类型	是否必填	参数说明
id	string	是	知识id

请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-
  application/open/document/1734816157446680576' \
2 --header 'accept: */*' \
3 --header 'Authorization:
  eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOiJlMzU4NiwiZm
  Nlcll9rZXkiOiJlZTFNTAwNC05MDZiLTQ0TEt0TkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc19pZ
  CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQiLCJ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
  Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg'
```

## 接口响应参数

documentData

### 响应示例

```
1 {
2   "data":{
3     "id":"12121212121212",
4     "custom_separator":"\n",
5     "sentence_size":300,
6     "length":0,
7     "word_num":100,
8     "name":"",
9     "url":"",
10    "embedding_stat":0,
11    "failInfo":{
12      "embedding_code":10002, // 见统一基础结构体：向量化失败状态码
13      "embedding_msg":"字数超出限制"
14    }
15  },
16  "code":200,
17  "message":"请求成功",
18  "timestamp":1689649504996
19 }
```

## 删除知识

### 接口请求

传输方式	https
请求地址	/document/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	PATH
响应格式	JSON
接口请求类型	DELETE

## 接口路径参数

参数名称	类型	是否必填	参数说明
id	string	是	知识id

请求示例

```
1 curl --location --request DELETE 'https://open.bigmodel.cn/api/llm-
  application/open/document/1734816156775591936' \
2 --header 'accept: */*' \
3 --header 'Authorization:
  eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOiE2MzU4NiwiZX
  Nlcl9rZXkiOiJlZTFNTAwNC05MDZiLTQ0TEtOTkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc19pZ
  CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQ1LCJlc2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
  Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg'
```

接口响应参数

参数名称	类型	参数说明
------	----	------

响应示例

```
1 {
2   "code": 200,
3   "message": "请求成功",
4   "timestamp": 1689649504996
5 }
6
```

上传url构建知识

用于创建个人知识,文件下载地址或者web页面

接口请求

传输方式	https
请求地址	/document/upload_url
调用方式	同步调用返回上传结果; 向量化结果需要调查询接口
字符编码	UTF-8





```

Nlcl9rZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEtOTkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc9pZ
CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQlLCJlc2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNzYhQyS7Cg' \
4 --header 'Content-Type: application/json' \
5 --data '{
6   "knowledge_id": "1734786402936664064",
7   "upload_detail": [
8     {
9       "knowledge_type": 1,
10      "url": "https://blog.csdn.net/zmm0413/article/details/108078400"
11    }
12  ]
13 }'
```

## 接口响应参数

### 响应示例

```

1 {
2   "data": {
3     "successInfos": [
4       {
5         "documentId": "122121212",
6         "url": "xxx.com"
7       },
8       {
9         "documentId": "12121212121",
10        "url": "xxx.com"
11      }
12    ],
13    "failedInfos": [
14      {
15        "url": "xxx.com",
16        "failReason": "不支持的文档类型"
17      }
18    ]
19  },
20  "code": 200,
21  "message": "请求成功",
22  "timestamp": 1689649504996
23 }
24
```

## 重新向量化

用于重新向量化文档（重试等操作）, 异步等待最终返回结果, 多用于url知识场景

接口请求

传输方式	https
请求地址	/document/embedding/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON
接口请求类型	POST

接口路径参数

参数名称	类型	是否必填	参数说明
id	string	是	知识id

请求示例

```
1 curl --location --request POST 'https://open.bigmodel.cn/api/llm-
  application/open/document/embedding/1734888894421270528' \
2 --header 'accept: */*' \
3 --header 'Authorization:
eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfaWQiOiE2MzU4NiwiZX
Nlcl9rZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEt0TkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc19pZ
CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQiLCJ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
Zz132GawOGHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHnzYhQyS7Cg'
```

接口响应参数

参数名称	类型	参数说明
------	----	------

响应示例

```
1 {
2   "code": 200,
3   "message": "请求成功",
4   "timestamp": 1689649504996
5 }
6
```

## 应用



创建一个专属应用，应用可以引用自建知识库来提高问答准确度

## 创建应用

用于创建个人应用

## 接口请求

传输方式	https
请求地址	/application
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON
接口请求类型	POST

## 接口请求参数

参数名称	类型	是否必填	参数说明
name	string	是	应用名称, 1-20字
desc	string	是	应用描述, 0-100字
prompt	string	否	模版,格式为“ <code>{{知识}}</code> xxx <code>{{用户}}</code> ”,必须包含知识和用户, 且占位符 <code>{{知识}}</code> 有且只有一个, 占位符 <code>{{用户}}</code> 有且只有一个
top_p	string	否	默认0.7, 用温度取样的另一种方法, 称为核取样

			<p>取值范围是：(0.0, 1.0) 开区间，不能等于 0 或 1，默认值为 0.7</p> <p>模型考虑具有 top_p 概率质量tokens的结果</p> <p>例如：0.1 意味着模型解码器只考虑从前 10% 的概率的候选集中取tokens</p> <p>建议您根据应用场景调整 top_p 或 temperature 参数，但不要同时调整两个参数</p>
temperature	string	否	<p>默认0.95，采样温度，控制输出的随机性，必须为正数</p> <p>取值范围是：(0.0,1.0]，不能等于 0,默认值为 0.95</p> <p>值越大，会使输出更随机，更具创造性；值越小，输出会更加稳定或确定</p> <p>建议您根据应用场景调整 top_p 或 temperature 参数，但不要同时调整两个参数</p>
knowledge_ids	list<long>	是	知识库id列表
model	enum	否	取值范围：chatglm_pro,chatglm_std,chatglm_lite,chatglm_lite_32k; 默认chatglm_pro
icon_color	enum	否	取值范围：blue, red, orange, purple, sky
icon_type	enum	否	<p>取值范围：</p> <p>'robot_assist',</p> <p>'robot_navigate',</p> <p>'robot_guardian',</p> <p>'robot_service',</p> <p>'robot_messenger',</p> <p>'robot_chat',</p> <p>'robot_engineer'</p>

## 请求示例

```

1 curl --location 'https://open.bigmodel.cn/api/llm-application/open/application' \
2 --header 'accept: */*' \
3 --header 'Authorization: eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfaWQiOiJlLCJl9rZXkiOiJlZTFNTAwNC05MDZiLTQ0TE0TbkMiO00TE0NTU5NGU1MDAiLCJjdXN0b21lc1pZCI6Ijc2OTk3NTcwOTMyNzA0MjUzMjQ1LCJlc2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2VZz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHnzYhQyS7Cg' \

```

```
4 --header 'Content-Type: application/json' \  
5 --data '{  
6     "name": "app",  
7     "desc": "test",  
8     "knowledge_ids": [0],  
9     "model": "chatglm_pro",  
10    "prompt": "有如下的相关信息：\n\n\"\"\n\n\"{知识}\n\n\"\"\n\n\n\n使用上述相关的信息，  
回答下方问题，你所需要的全部内容都在相关信息中：\n\n\"\"\n\n\"{用户}\n\n\"\"\n\n\"\"  
11 }'
```

接口响应参数

参数名称	类型	参数说明
id	string	生成的应用唯一id

响应示例

```
1 {  
2     "data": "111111111111111111"  
3     "code": 200,  
4     "message": "请求成功",  
5     "timestamp": 1689649504996  
6 }  
7
```

编辑应用

用于编辑个人应用

接口请求

传输方式	https
请求地址	/application/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON

接口请求类型	PUT
--------	-----

## 路径参数

参数名称	类型	是否必填	参数说明
id	string	是	应用id

## 接口请求参数

参数名称	类型	是否必填	参数说明
name	string	否	应用名称
desc	string	否	应用描述
prompt	string	否	模版,格式为“ <code>{{知识}}xxx{{用户}}</code> ”,必须包含知识和用户,且占位符 <code>{{知识}}</code> 有且只有一个,占位符 <code>{{用户}}</code> 有且只有一个
top_p	string	否	默认0.7,用温度取样的另一种方法,称为核取样 取值范围是: <code>(0.0, 1.0)</code> 开区间,不能等于0或1,默认值为0.7 模型考虑具有 <code>top_p</code> 概率质量tokens的结果 例如:0.1意味着模型解码器只考虑从前10%的概率的候选集中取tokens 建议您根据应用场景调整 <code>top_p</code> 或 <code>temperature</code> 参数,但不要同时调整两个参数
temperature	string	否	默认0.95,采样温度,控制输出的随机性,必须为正数 取值范围是: <code>[0.0,1.0]</code> ,不能等于0,默认值为0.95 值越大,会使输出更随机,更具创造性;值越小,输出会更加稳定或确定 建议您根据应用场景调整 <code>top_p</code> 或 <code>temperature</code> 参数,但不要同时调整两个参数
knowledge_ids	list<long>	否	知识库id列表
model	enum	否	取值范围:chatglm_pro,chatglm_std,chatglm_lite,chatglm_lite_32k
icon_color	enum	否	取值范围:blue, red, orange, purple, sky

icon_type	enum	否	取值范围:  'robot_assist',  'robot_navigate',  'robot_guardian',  'robot_service',  'robot_messenger',  'robot_chat',  'robot_engineer'
-----------	------	---	---

请求示例

```
1 curl --location --request PUT 'https://open.bigmodel.cn/api/llm-  
application/open/application/1712773365310599168' \  
2 --header 'accept: */*' \  
3 --header 'Authorization:  
eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOiJlE2MzU4NiwiZmVzZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEtOTkxMi00OTE0NTU5NGU1MDAiLCJjdXN0b21lc9pZ  
CI6Ijc20Tk3NTcwOTMyNzA0MjUzZmQjQjEjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V  
Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHnzYhQyS7Cg' \  
4 --header 'Content-Type: application/json' \  
5 --data '{  
6     "name": "小程序"  
7 }'
```

接口响应参数

响应示例

```
1 {  
2     "code": 200,  
3     "message": "请求成功",  
4     "timestamp": 1689649504996  
5 }  
6
```

应用列表

接口请求





## 响应示例

```
1 {
2   "data": {
3     "list": [
4       {
5         "id": "121212121212",
6         "name": "高考填报指南",
7         "desc": "高考填报指南",
8         "model": "chatglm130b",
9         "slice_count": 2,
10        "knowledge_ids": [1,2],
11        "prompt": "以下是知识库内容：\n{{知识}}\n\n请根据知识库中内容回答，如果",
12        "top_p": "0.2",
13        "temperature": "0.3"
14      },
15    ],
16    "total": 1
17  },
18  "code": 200,
19  "message": "请求成功",
20  "timestamp": 1689665168772
21 }
```

## 应用查看

### 接口请求

传输方式	https
请求地址	/application/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON
接口请求类型	GET

### 接口路径参数

参数名称	类型	是否必填	参数说明
id	string	是	应用id

请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/lm-
  application/open/application/1711282458583887873' \
2 --header 'accept: */*' \
3 --header 'Authorization:
  eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJT05BTCIsInVzZXJfawQiOiJlMzU4NiwiOiJlcl9rZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEtOTkxMi000TE0NTU5NGU1MDA1LCJjdXN0b21lc19pZ
  CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQ1LCJlc2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
  Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiG4SSHNz1YhQyS7Cg'
```

接口响应参数

参考 applicationData

响应示例

```
1 {
2   "data": {
3     "id": "121212121212",
4     "name": "高考填报指南",
5     "desc": "高考填报指南",
6     "model": "chatglm130b",
7     "slice_count": 2,
8     "knowledge_ids": [1, 2],
9     "prompt": "以下是知识库内容: \n{{知识}}\n\n请根据知识库中内容回答, 如果不知道答
10    "top_p": "0.2",
11    "temperature": "0.3"
12  },
13  "code": 200,
14  "message": "请求成功",
15  "timestamp": 1689665168772
16 }
```

删除应用

用于删除应用

接口请求

传输方式	https
请求地址	/application/{id}
调用方式	同步调用，等待返回结果
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON
接口请求类型	DELETE

接口请求参数

参数名称	类型	是否必填	参数说明
------	----	------	------

请求示例

```
1 curl --location --request DELETE 'https://open.bigmodel.cn/api/llm-
  application/open/application/1699710846501031936' \
2 --header 'accept: */*' \
3 --header 'Authorization:
  eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfawQiOiE2MzU4NiwiZX
  Nlcl9rZXkiOiJlZTFhNTAwNC05MDZiLTQ0TEt0TkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc19pZ
  CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQjLCJ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
  Zz13Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHNz1YhQyS7Cg'
```

接口响应参数

参数名称	类型	参数说明
------	----	------

响应示例

```
1 {
2   "code": 200,
3   "message": "请求成功",
4   "timestamp": 1689649504996
5 }
6
```

# 调用应用



使用创建的应用调用大模型

## http同步接口

传输方式	https
请求地址	/model-api/{id}/invoke
调用方式	同步返回结果
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON
接口请求类型	POST

## 路径参数

参数名称	类型	是否必填	参数说明
id	string	是	应用id

## 请求参数

参数名称	类型	是否必填	参数说明
prompt	list	是	<div><div>1.</div><div>调用对话模型时，将当前对话信息列表作为提示输入给模型; 按照 <code>{"role": "user", "content": "你好"}</code> 的键值对形式进行传参; 总长度超过模型最长输入限制后会 自动截断，需按时间由旧到新排序</div></div> <div><div>2.</div><div>list必须是奇数个</div></div>
returnType	String	否	<div>用于控制每次返回内容的类型，空或者没有此字段时默认按照 json_string返回</div> <div>- <code>json_string</code> 返回标准的 JSON 字符串</div>

			- <code>text</code> 返回原始的文本内容
knowledge_ids	list<long>	否	知识库id列表: 不传,则使用应用关联的所有知识库
document_ids	list<long>	否	知识id列表: 不传,则使用应用关联的所有知识库

请求示例

```
1 curl --location 'https://open.bigmodel.cn/api/llm-application/open/model-  
api/1712773365310599168/invoke' \  
2 --header 'accept: */*' \  
3 --header 'Authorization:  
eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJTT05BTCIsInVzZXJfaWQiOiJlcmVzZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEtOTkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc19pZ  
CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQ1LCJlc2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V  
Zz32Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHnz1YhQyS7Cg' \  
4 --header 'Content-Type: application/json' \  
5 --data '{  
6   "prompt": [  
7     {  
8       "role": "user",  
9       "content": "今天吃什么"  
10    }  
11  ]  
12 }'
```

接口响应参数

响应示例

```
1 {  
2   "data": {  
3     "requestId": "1697192365237-497892",  
4     "content": "回答的内容"  
5   },  
6   "code": 200,  
7   "message": "请求成功",  
8   "timestamp": 1697192371071  
9 }
```

httpsse接口

接口请求

传输方式	https
请求地址	/model-api/{id}/sse-invoke
调用方式	SSE
字符编码	UTF-8
接口请求格式	JSON
响应格式	标准 Event Stream
接口请求类型	POST
请求头	accept:text/event-stream

路径参数

参数名称	类型	是否必填	参数说明
id	string	是	应用id

接口请求参数

参数名称	类型	是否必填	参数说明
prompt	list	是	<div>1. 调用对话模型时，将当前对话信息列表作为提示输入给模型; 按照 <code>{"role": "user", "content": "你好"}</code> 的键值对形式进行传参; 总长度超过模型最长输入限制后会自动截断，需按时间由旧到新排序</div> <div>2. list必须是奇数个</div>
incremental	boolean	否	SSE接口调用时，用于控制每次返回内容方式是增量还是全量，不提供此参数时默认为全量返回 <ul style="list-style-type: none"><li><code>true</code> 为增量返回</li><li><code>false</code> 为全量返回</li></ul>
knowledge_ids	list<long>	否	知识库id列表: 不传,则使用应用关联的所有知识库
document_ids	list<long>	否	知识id列表: 不传,则使用应用关联的所有知识库

单轮轮对话prompt示例

每次调用只需传本次提问prompt

```
1 [
2   {
3     "role": "user",
4     "content": "今天吃什么"
5   }
6 ]
```

## 多轮对话prompt示例

1. 每次调用需要带上历史对话,最多支持15轮
2. list必须是奇数个
3. 多轮对话只会对第一轮问题注入知识, 后续对话大模型都是基于第一轮制问题匹配的知识进行推理的

```
1 [
2   {
3     "role": "user",
4     "content": "你好"
5   },
6   {
7     "role": "assistant",
8     "content": " 你好！很高兴为您提供帮助。请问您有什么问题需要我解答呢？"
9   },
10  {
11    "role": "user",
12    "content": "今天吃什么"
13  },
14  {
15    "role": "assistant",
16    "content": " 根据相关信息，我无法为您提供具体的建议。但您可以根据自己的口味、营养
17  },
18  {
19    "role": "user",
20    "content": "我喜欢吃甜的"
21  }
22 ]
```

## 请求示例



```

1 curl --location 'https://open.bigmodel.cn/api/llm-application/open/model-
  api/1712773365310599168/sse-invoke' \
2 --header 'accept: */*' \
3 --header 'Authorization:
  eyJhbGciOiJIUzUxMiJ9.eyJ1c2VyX3R5cGUiOiJQRVJT05BTCIsInVzZXJfawQiOiJlE2MzU4NiwiZ
  Nlcl9rZXkiOiJlZTFlNTAwNC05MDZiLTQ0TEtOTkxMi000TE0NTU5NGU1MDAiLCJjdXN0b21lc
  l9pZ
  CI6Ijc20Tk3NTcwOTMyNzA0MjUzMjQjLCJ1c2VybmFtZSI6IjE1MjMwNTY1OTkwIn0.rcufXsv0vR2V
  Zz132Gaw0GHG0x-sE0DD_cCfkfG13jhVAgKq-ePxdSCvb8tX5ctIU9BuiGj4SSHnzYhQyS7Cg' \
4 --header 'Content-Type: application/json' \
5 --data '{
6   "prompt": [
7     {
8       "role": "user",
9       "content": "今天吃什么"
10    }
11  ]
12 }'

```

## 接口响应参数

SSE 响应是字符串流格式，先看下具体响应示例

```

1 id: "fb981fde-0080-4933-b87b-4a29eaba8d17"
2 event: "add"
3 data: "作为一个"
4
5 id: "fb981fde-0080-4933-b87b-4a29eaba8d17"
6 event: "add"
7 data: "大型语言模型"
8
9 id: "fb981fde-0080-4933-b87b-4a29eaba8d17"
10 event: "add"
11 data: "我可以"
12
13 ... ..
14
15 Id: "fb981fde-0080-4933-b87b-4a29eaba8d17"
16 event: "finish"
17 data: "我可以"

```

- 1 id: 消息唯一标识
- 2 event: 消息类型，add 增量，finish 结束，errorhandle 错误，interrupted 中断
- 3 data: 具体内容

# 切片位置信息

用于获取调用应用对话所匹配的切片位置信息

目前只支持 pdf 和 excel 展示切片位置信息

## 接口请求

传输方式	https
请求地址	/v2/{request_id}/slice_info
调用方式	同步调用
字符编码	UTF-8
接口请求格式	JSON
响应格式	JSON
接口请求类型	POST

## 路径参数

参数名称	类型	是否必填	参数说明
request_id	string	是	使用调用应用对话接口返回的唯一id <a href="#">1. 同步接口: requestId</a> <a href="#">2. sse接口: id</a>

## 接口请求参数

## 接口返回参数

参数名称	类型	参数说明
document_slices	list<DcoumentSlice s>	知识切片信息
has_old_docum ent	boolean	是否存在没有切片位置的历史文档 true:可以找出hide_positions为true的文档重新进行向量化

		false:无需关注
--	--	------------

DcocumentSlices:

参数名称	类型	参数说明
document	Document	知识信息
slice_info	List<Slice>	切片信息
hide_positions	boolean	是否存在历史文档切片没有位置信息; true: 需要重新进行向量化才能获取位置信息,见 重新向量化接口; false:无需关注

Document:

参数名称	类型	参数说明
id	string	知识唯一id
name	string	知识名称
url	string	知识url
d_type	int	知识类型, 见统一结构体

Slice:

参数名称	类型	参数说明
document_id	string	知识唯一id
position	Position	pdf切片位置信息
line	int	sheet行号
sheet_name	string	sheet名称
text	string	切片内容

注: pdf知识 使用position位置信息; excel使用 line和sheet\_name位置信息

Position:

参数名称	类型	参数说明
x0	decimal	左侧到行左侧的距离
x1	decimal	字符顶部到顶部距离
top	decimal	字符顶部到顶部距离
bottom	decimal	字符底部到顶部距离
page	int	所在页
height	decimal	所在页高
width	decimal	所在页宽度

示例

```
1 {
2   "data":{
3     "document_slices":[
4       {
5         "document":{
6           "id":"171128203689965971",
7           "name":"xx.pdf",
8           "url":"https://cdn.bigmodel.cn/knowledge_test/xx.pdf",
9           "dtype":3
10        },
11        "slice_info":[
12          {
13            "document_id":"171128203689965971",
14            "position":{
15              "x0":90,
16              "x1":267.46999999999997,
17              "top":293.293,
18              "bottom":303.24300000000005,
19              "page":6,
20              "height":841.9,
21              "width":595.3
22            },
23            "text":"你好你好你好你好你好你好你好你好你好"
24          }
25        ],
26        "hide_positions":false
27      },
28      {
29        "document":{
```

```
30     "id": "171128203689965971",
31     "name": "xx.docx",
32     "url": "https://cdn.bigmodel.cn/knowledge_test/xx.docx",
33     "dtype": 6
34 },
35 "slice_info": [
36     {
37         "document_id": "171128203689965971",
38         "line": 1,
39         "sheet_name": "sheet1",
40         "text": "你好你好你好你好你好你好你好你好你好"
41     }
42 ],
43 "hide_positions": false
44 },
45 ],
46 "has_old_document": false
47 },
48 "code": 200,
49 "message": "请求成功",
50 "timestamp": 1697438393502
51 }
```