**1. 接口说明**

**注意： 该接口可以正式使用。如您需要申请使用，请点击前往**[产品页面](https://xinghuo.xfyun.cn/sparkapi?scr=price" \t "_blank)**领取免费额度。**

*1. 计费包含接口的输入和输出内容；*  
*2. 1token 约等于1.5个中文汉字 或者 0.8个英文单词；*  
*3. Spark Lite支持[搜索]内置插件；Spark Pro, Spark Max和Spark 4.0Ultra支持[搜索]、[天气]、[日期]、[诗词]、[字词]、[股票]六个内置插件；*  
*4. Spark 4.0Ultra/Max现已支持system、Function Call功能；*

[#](https://www.xfyun.cn/doc/spark/HTTP%E8%B0%83%E7%94%A8%E6%96%87%E6%A1%A3.html#_2-%E8%AF%B7%E6%B1%82%E5%9C%B0%E5%9D%80)**2. 请求地址**

**Tips: 星火大模型API当前有Lite、Pro、Pro-128K、Max、Max-32K和4.0 Ultra六个版本，各版本独立计量tokens。**

<https://spark-api-open.xf-yun.com/v1/chat/completions>

**3. 请求说明**

[#](https://www.xfyun.cn/doc/spark/HTTP%E8%B0%83%E7%94%A8%E6%96%87%E6%A1%A3.html#_3-1-%E8%AF%B7%E6%B1%82%E5%A4%B4)**3.1. 请求头**

请到控制台获取http服务接口认证信息中的APIPassword,假如获取到的值为123456，则请求头如下：

Content-Type: application/json

Authorization: Bearer 123456

利用上方的请求头发起请求示例如下：

curl -i -k -X POST 'https://spark-api-open.xf-yun.com/v1/chat/completions' \

--header 'Authorization: Bearer 123456' \

--header 'Content-Type: application/json' \

--data '{

"model":"generalv3.5",

"messages": [

{

"role": "user",

"content": "来一个只有程序员能听懂的笑话"

}

],

"stream": true

}'

| **参数名称** | **类型** | **是否必传** | **取值范围** | **描述** |
| --- | --- | --- | --- | --- |
| model | string | 是 | lite generalv3 pro-128k generalv3.5 max-32k 4.0Ultra | 指定访问的模型版本: lite指向Lite版本; generalv3指向Pro版本; pro-128k指向Pro-128K版本; generalv3.5指向Max版本; max-32k指向Max-32K版本; 4.0Ultra指向4.0 Ultra版本; |
| user | string | 否 | 自定义 | 用户的唯一id，表示一个用户，user\_123456 |
| messages | array | 是 |  | 输入数组 |
| messages.role | string | 是 | user assistant system tool | 角色，user表示用户，assistant表示大模型，system表示命令，tool代表function call执行结果 |
| messages.content | string | 是 |  | 角色对应的文本内容 |
| temperature | float | 否 | 取值范围[0, 2] 默认值1.0 | 核采样阈值 |
| top\_p | int | 否 | 取值范围(0, 1] 默认值1 | 生成过程中核采样方法概率阈值，例如，取值为0.8时，仅保留概率加起来大于等于0.8的最可能token的最小集合作为候选集。取值越大，生成的随机性越高；取值越低，生成的确定性越高。 |
| top\_k | int | 否 | 取值范围[1, 6] 默认值4 | 从k个中随机选择一个(非等概率) |
| presence\_penalty | float | 否 | 取值范围[-2.0,2.0] 默认0 | 重复词的惩罚值 |
| frequency\_penalty | float | 否 | 取值范围[-2.0,2.0] 默认0 | 频率惩罚值 |
| stream | bool | 否 | true false | 是否流式返回结果。默认是false 表示非流式。 如果使用流式，服务端使用SSE的方式推送结果，客户端自己适配处理结果。 |
| max\_tokens | int | 否 | Pro、Max、Max-32K、4.0 Ultra 取值为[1,8192]，默认为4096; Lite、Pro-128K 取值为[1,4096]，默认为4096。 | 模型回答的tokens的最大长度 |
| response\_format | object | 否 |  | 指定模型的输出格式 |
| response\_format.type | string | 否 | text json\_object | { "type": "json\_object" } 指定模型输出json格式 使用 JSON 模式时，请始终指示模型通过对话中的某些消息（例如通过系统或用户消息）生成 JSON |
| tools | array Optional | 否 |  | 工具参数 |
| tools.function | object | 否 |  | {"type":"function", "function":{"name": "my\_function", "description": "xxx", "parameters": {...}}} parameters要符合json schema 描述 |
| tools.web\_search | object | 否，默认表示开启 | 属性值：{"type": "web\_search", "web\_search": {"enable": true, "show\_ref\_label": true}} enable 取值范围为：true or false show\_ref\_label 取值范围为：true or false | enable 开关表示是否开启搜索功能，关闭时不会联网搜索； show\_ref\_label 开关表示触发检索时是否返回信源信息（仅在enable为true时生效） |
| tool\_calls\_switch | bool | 否，默认表示关闭 | true or false | 设置为true时，触发function call结果中tool\_calls以数组格式返回，默认为 false，则以json格式返回 |
| tool\_choice | string or object Optional | 否 | auto none required {"type": "function", "function": {"name": "my\_function"}} | 设置模型自动选择调用的函数： auto：传了tool时默认为auto，模型自动选择调用的函数 none：模型禁用函数调用 required：模型始终选择一个或多个函数进行调用 {"type": "function", "function": {"name": "my\_function"}} ：模型强制调用指定函数 |

[#](https://www.xfyun.cn/doc/spark/HTTP%E8%B0%83%E7%94%A8%E6%96%87%E6%A1%A3.html#_4-%E5%93%8D%E5%BA%94%E5%8F%82%E6%95%B0)**4. 响应参数**

**请求错误时的响应格式：**

API

{

"error": {

"message": "invalid user",

"type": "api\_error",

"param": null,

"code": null

}

}

SDK

openai.AuthenticationError: Error code: 401 - {'error': {'message': 'invalid user', 'type': 'api\_error', 'param': None, 'code': None}}

**非流式请求成功时的响应格式：**

{

"code": 0,

"message": "Success",

"sid": "cha000b0003@dx1905cd86d6bb86d552",

"choices": [

{

"message": {

"role": "assistant",

"content": "你好，我是由科大讯飞构建的星火认知智能模型。\n如果你有任何问题或者需要帮助的地方，请随时告诉我！我会尽力为你提供解答和支持。请问有什么可以帮到你的吗？"

},

"index": 0

}

],

"usage": {

"prompt\_tokens": 6,

"completion\_tokens": 42,

"total\_tokens": 48

}

}

**响应的参数说明：**

| **参数名称** | **类型** | **描述** |
| --- | --- | --- |
| code | int | 错误码：0表示成功，非0表示错误 |
| message | string | 错误码的描述信息 |
| sid | string | 本次请求的唯一id |
| choices | array | 大模型结果的数组 |
| choices.message | object | 大模型结果 |
| choices.message.role | string | 大模型的角色 |
| choices.message.content | string | 大模型输出的内容 |
| choices.index | int | 大模型的结果序号，在多候选中使用 |
| usage | object | 本次请求消耗的token数量 |
| usage.prompt\_tokens | int | 用户输入信息，消耗的token数量 |
| usage.completion\_tokens | int | 大模型输出信息，消耗的token数量 |
| usage.total\_tokens | int | 用户输入+大模型输出，总的token数量 |

流式请求成功时的响应格式：

data:{"code":0,"message":"Success","sid":"cha000b000c@dx1905cf38fc8b86d552","id":"cha000b000c@dx1905cf38fc8b86d552","created":1719546385,"choices":[{"delta":{"role":"assistant","content":"你好"},"index":0}]}

data:{"code":0,"message":"Success","sid":"cha000b000c@dx1905cf38fc8b86d552","id":"cha000b000c@dx1905cf38fc8b86d552","created":1719546385,"choices":[{"delta":{"role":"assistant","content":"，很高兴"},"index":0}]}

data:{"code":0,"message":"Success","sid":"cha000b000c@dx1905cf38fc8b86d552","id":"cha000b000c@dx1905cf38fc8b86d552","created":1719546385,"choices":[{"delta":{"role":"assistant","content":"为你解答问题"},"index":0}]}

data:{"code":0,"message":"Success","sid":"cha000b000c@dx1905cf38fc8b86d552","id":"cha000b000c@dx1905cf38fc8b86d552","created":1719546385,"choices":[{"delta":{"role":"assistant","content":"。\n"},"index":0}]}

data:{"code":0,"message":"Success","sid":"cha000b000c@dx1905cf38fc8b86d552","id":"cha000b000c@dx1905cf38fc8b86d552","created":1719546387,"choices":[{"delta":{"role":"assistant","content":"我是讯飞星火认知大模型，由科大讯飞构建的认知智能系统。"},"index":0}]}

data:{"code":0,"message":"Success","sid":"cha000b000c@dx1905cf38fc8b86d552","id":"cha000b000c@dx1905cf38fc8b86d552","created":1719546388,"choices":[{"delta":{"role":"assistant","content":"我具备与人类进行自然交流的能力，可以高效地满足各领域的认知智能需求。"},"index":0}]}

data:{"code":0,"message":"Success","sid":"cha000b000c@dx1905cf38fc8b86d552","id":"cha000b000c@dx1905cf38fc8b86d552","created":1719546389,"choices":[{"delta":{"role":"assistant","content":"无论你有什么问题或者需要帮助的地方，我都将尽我所能提供支持和解决方案。请随时告诉我你的需求！"},"index":0}]}

data:{"code":0,"message":"Success","sid":"cha000b000c@dx1905cf38fc8b86d552","id":"cha000b000c@dx1905cf38fc8b86d552","created":1719546389,"choices":[{"delta":{"role":"assistant","content":""},"index":0}],"usage":{"prompt\_tokens":6,"completion\_tokens":68,"total\_tokens":74}}

data:[DONE]

**响应的参数说明：**

| **参数名称** | **类型** | **描述** |
| --- | --- | --- |
| code | int | 错误码：0表示成功，非0表示错误 |
| message | string | 错误码的描述信息 |
| sid | string | 本次请求的唯一id |
| choices | array | 大模型结果的数组 |
| choices.delta | object | 大模型结果 |
| choices.delta.role | string | 大模型的角色 |
| choices.delta.content | string | 大模型输出的内容 |
| choices.index | int | 大模型的结果序号，在多候选中使用 |
| usage | object | 本次请求消耗的token数量 |
| usage.prompt\_tokens | int | 用户输入信息，消耗的token数量 |
| usage.completion\_tokens | int | 大模型输出信息，消耗的token数量 |
| usage.total\_tokens | int | 用户输入+大模型输出，总的token数量 |

[#](https://www.xfyun.cn/doc/spark/HTTP%E8%B0%83%E7%94%A8%E6%96%87%E6%A1%A3.html#_5-http%E9%9D%9E%E6%B5%81%E5%BC%8F%E8%AF%B7%E6%B1%82%E7%A4%BA%E4%BE%8B)**5. HTTP非流式请求示例**

import requests

url = "https://spark-api-open.xf-yun.com/v1/chat/completions"

data = {

"model": "generalv3.5", # 指定请求的模型

"messages": [

{

"role": "user",

"content": "你是谁"

}

],

"tools": [

{

"type": "function",

"function": {

"name": "get\_current\_weather",

"description": "返回实时天气",

"parameters": {

"type": "object",

"properties": {

"location": {

"type": "string",

"description": "河北省承德市双桥区",

},

"format": {

"type": "string",

"enum": ["celsius", "fahrenheit"],

"description": "使用本地区常用的温度单位计量",

},

},

"required": ["location", "format"],

}

}

}

]

}

header = {

"Authorization": "Bearer 123456" # 注意此处替换自己的APIPassword

}

response = requests.post(url, headers=header, json=data)

print(response.text)

**6. HTTP流式请求示例**

import requests

url = "https://spark-api-open.xf-yun.com/v1/chat/completions"

data = {

"model": "generalv3.5", # 指定请求的模型

"messages": [

{

"role": "user",

"content": "你是谁"

}

],

"stream": True

}

header = {

"Authorization": "Bearer 123456" # 注意此处替换自己的APIPassword

}

response = requests.post(url, headers=header, json=data, stream=True)

# 流式响应解析示例

response.encoding = "utf-8"

for line in response.iter\_lines(decode\_unicode="utf-8"):

print(line)

**7. 使用OpenAI SDK请求示例**

# 安装openai SDK

pip install openai

# 导入SDK，发起请求

from openai import OpenAI

client = OpenAI(

# 控制台获取key和secret拼接，假使控制台获取的APIPassword是123456

api\_key="123456",

base\_url = 'https://spark-api-open.xf-yun.com/v1' # 指向讯飞星火的请求地址

)

completion = client.chat.completions.create(

model='generalv3.5', # 指定请求的版本

messages=[

{

"role": "user",

"content": '说一个程序员才懂的笑话'

}

]

)

print(completion.choices[0].message)

**8. 错误码说明**

| **错误码** | **错误信息** |
| --- | --- |
| 0 | 成功 |
| 10007 | 用户流量受限：服务正在处理用户当前的问题，需等待处理完成后再发送新的请求。（必须要等大模型完全回复之后，才能发送下一个问题） |
| 10013 | 输入内容审核不通过，涉嫌违规，请重新调整输入内容 |
| 10014 | 输出内容涉及敏感信息，审核不通过，后续结果无法展示给用户 |
| 10019 | 表示本次会话内容有涉及违规信息的倾向；建议开发者收到此错误码后给用户一个输入涉及违规的提示 |
| 10907 | token数量超过上限。对话历史+问题的字数太多，需要精简输入 |
| 11200 | 授权错误：该appId没有相关功能的授权 或者 业务量超过限制 |
| 11201 | 授权错误：日流控超限。超过当日最大访问量的限制 |
| 11202 | 授权错误：秒级流控超限。秒级并发超过授权路数限制 |
| 11203 | 授权错误：并发流控超限。并发路数超过授权路数限制 |