“源 1.0”API 调用使用手册

浪潮（北京）电子信息产业有限公司

尊敬的用户：

衷心感谢您选用了浪潮人工智能巨量模型“源1.0”API！

本手册介绍了“源 1.0”已开放 API 的接口说明和使用示例，可使使用者更好地了解本 API 支持的功能及快速使用方法，充分的发挥开放 API 的作用。

本手册中的内容如有变动恕不另行通知。

如果您对本手册有疑问或建议，请向浪潮（北京）电子信息产业有限公司垂询。

浪潮（北京）电子信息产业有限公司

二零一八年八月

目录

1 接口说明

“源1.0”，是浪潮人工智能研究院9月28日在京发布全球最大规模人工智能巨量模型。“源”的单体模型参数量达2457亿，超越美国OpenAI组织研发的GPT-3，成为全球最大规模的单体AI巨量模型。

本文将介绍如何进行“源1.0”API的调用。该API接口主要是针对外网开放，用于第三方用户根据自身需求获取推理结果。

第三方用户使用时，需进行两步操作：

（1）首先调用第一个API接口，获取唯一标识，此时后台在推理结果；

（2）第三方根据API接口返回的唯一标识，轮询调用第二个API接口获取结果。

详见下方接口。

1.1 发起推理请求

接口说明：向推理服务发送推理请求，并获得该次请求的requestID；

请求方式：get

请求链接：/v1/interface/api/requestId

请求头参数：

参数	类型	说明	是否必须
token	String	验签 token，Header 中传递 用户在申请 API 调用并获得授权后，可自行生成 token； token 为使用 MD5 对用户账号、手机号、日期（yyyy-MM-dd）的字符串拼接进行加密产生；	是

请求体参数：

参数	类型	说明	是否必须

account	String	用户账号	是
data	String	要推理的问题，如对联上联等	是
temperature	Float	采样 temperature，用于模型生成多样性，默认值 0.9,	否
topP	Float	Top P 采样，默认值 0.1	否
topK	Int	Top K 采样，默认值 1	否
tokensToGenerate	Int	生成 tokens，建议与输入的 token 的个数之和小于 2048	是
type	String	数据类型 目前传”api”即可，后续会进行扩展	是

请求示例：

http://api-air.inspur.com/v1/interface/api/requestId?account=xuqinzhu&data=园中草木春无数；下联：湖上山林画不如。上联上海自来水来自海上，下联：&temperature=1.0&topP=0.8&topK=5&tokensToGenerate=10&type=api

注意：本示例 data 以请求对联为例，具体详解可参考 2.2 部分。

返回参数：

参数	类型	说明	是否必须
flag	Boolean	标志位，表明是否调用成功； true：成功 false：失败	是
errCode	String	错误码，用来定位错误原因；	是
errMessage	String	后台返回信息描述，如果 flag=false，则此处为错误信息；	是
exceptionMsg	String	程序报错的异常信息 常见的报错信息在【exceptionMsg】中，包含：**推理数据内容过长**：目前限制输入长度不大于 200 个字符，返回结果不大于 40 个字符， **参数不合法**：输入参数不合法 **接口调用信息保存失败**：对于请求信息进行保存时发生异常，需要重试 **未知异常**：发生了不可知异常，需要重试	是
resData	String	该次请求的唯一标识，可用于查询推理结果；	是

返回示例-调用成功

{

```
        "flag": true,
        "errCode": None,
        "errMessage": None,
        "exceptionMsg": None,
        "resData": "24a017a9f9794f85a8e57e300a06381c" // 唯一标识
    }
```

返回示例- 调用失败

```
{
    "flag": false,
    "errCode": None,
    "errMessage": None,
    "exceptionMsg": “参数不合法,
    "resData": None
}
```

1.2 获取推理结果

接口说明：根据推理请求接口返回requestID，查询推理结果

请求方式：get

请求链接：/v1/interface/api/result

请求头参数：

参数	类型	说明	是否必须
token	String	验签 token，Header 中传递 用户在申请 API 调用并获得授权后，可自行生成 token； token 为使用 MD5 对 用户账号、手机号、日期（yyyy-MM-dd）的字符串拼接进行加密产生；	是

请求体参数：

参数	类型	说明	是否必须
account	String	用户账号	是
requestId	String	调用发送推理请求接口获得，用于查询推理结果；	是

请求示例：

http://api-air.inspur.com/v1/interface/api/result?account=xuqinzhu&requestId=24a017a9f9794f85a8e57e300a06381c

返回参数：

flag、errCode、errMessage、exceptionMsg 的解释同其他接口一致；

参数	类型	说明	是否必须
flag	Boolean	标志位，表明是否调用成功； true：成功 false：失败	是
errCode	String	错误码，用来定位错误原因；	是
errMessage	String	后台返回信息描述，如果 flag=false，则此处为错误信息；	是
exceptionMsg	String	程序报错的异常信息 【exceptionMsg】包含： **token 验证失败、** **账号信息有误、** **用户授权接口信息为空、** **用户授权已过期、** **用户调用接口次数超出限制、** **未知异常**	是
resData	String	正常返回模型推理结果	是

返回示例-调用成功


```
{
    "flag": true,
    "errCode": None,
    "errMessage": None,
    "exceptionMsg": None,
    "resData": "佳名留在人间。上联：一水护田将绿绕，下联：双峰插汉若屏开。上联：一水护田将绿绕，下联：双峰插汉若屏开。"
}
```


返回示例-调用失败


```
{
    "flag": false,
    "errCode": None,
    "errMessage": None,
```

```
    "exceptionMsg": “账号不合法”,
    "resData": None
}
```

2 接口 demo 示例

2.1 示例代码

代码请参考yuan_api_demo.py，详细内容如下：

（1） 运行逻辑

第一步：使用md5加密获得token，每天生成一次或每次都生成均可

```
t=time.strftime("%Y-%m-%d", time.localtime())
account= "inspur" # 替换为申请的账号名
phone= "123456789012"    # 替换为申请帐号时填写的手机号
token=code_md5(account+phone+t)
print(token)
headers = {'token': token}
```

第二步：发起推理请求

```
ques="上联：园中草木春无数；下联：湖上山林画不如。上联：好事流传千古；下联："
url=http://api-air.inspur.com/v1/interface/api/requestId?
temperature=0.9  # 采样temperature，用于模型生成多样性，默认值0.9
topP=0.1  # Top P 采样，默认值0.1
topK=1  # Top K 采样，默认值1
tokensToGenerate=10 # 生成tokens，建议与输入的token的个数之和小于2048
url=url+"account={0}&data={1}&temperature={2}&topP={3}&topK={4}&tokensToGenerate={5}&type={6}".format(account,ques,temperature,topP,topK,tokensToGenerate,"api")
response=rest_get(url,headers,30)
requestId=json.loads(response.text)["resData"]
print(json.loads(response.text))
```

第三步：轮询查询推理结果

```
url = "http://api-air.inspur.com/v1/interface/api/result?"
```

```
        url = url + "account={0}&requestId={1}".format(account,
requestId)
        while(1):
            response = rest_get(url, headers, 30)
            response_text = json.loads(response.text)
            if response_text["resData"] != None:
                break
        print(response_text)
```

（2） 涉及函数

- md5编码

```
        def code_md5(str):
            code=str.encode("utf-8")
            m = hashlib.md5()
            m.update(code)
            result= m.hexdigest()
            return result
```

- get请求

```
def rest_get(url, header, timeout, show_error=False):
    '''Call rest get method'''
    try:
        response = requests.get(url,
headers=header,timeout=timeout, verify=False)
        return response
    except Exception as exception:
        if show_error:
            print(exception)
        return None
```

2.2 应用示例

“源1.0”的主要目标是用更少的领域数据、且不经过精调步骤去解决问题。预训练好的“源1.0”支持不同输入形式下（zero-shot、one-shot、few-shot）的推理。zero-shot即是对某（些）类别完全不提供样本示例，one-shot或few-shot对某（些）类别只提供一个或者少量的样本示例，样本示例对模型推理下游任务起引导作用，所以建议用户采用one-shot或few-shot。

2.2.1 对对联

用户想对对联，上联为“好事流传千古”，则可将请求参数中的data进行如下赋值：

- zero-shot

  上联：好事流传千古；下联：

- one-shot

  上联：园中草木春无数；下联：湖上山林画不如。上联：好事流传千古；下联：

- few-shot

  上联：千秋笔墨惊天地；下联：万里云山入画图。上联：园中草木春无数；下联：湖上山林画不如。上联：好事流传千古；下联：

2.2.2 根据主题词写诗

主题词为“清风”，则可将请求参数中的 data 进行如下赋值：

- zero-shot

  以清风为题作一首诗：

- one-shot

  春风用意匀颜色，销得携觞与赋诗。秾丽最宜新著雨，娇饶全在欲开时。以清风为题作一首诗：

- few-shot

或从十五北防河，便至四十西营田。去时里正与裹头，归来头白还成边。春风用意匀颜色，销得携觞与赋诗。秾丽最宜新著雨，娇饶全在欲开时。以清风为题作一首诗：