

工作流编排对话型应用 API

对话应用支持会话持久化，可将之前的聊天记录作为上下文进行回答，可适用于聊天/客服 AI 等。

API 服务器:

https://api.dify.ai/v1

基础 URL

Code

```
https://api.dify.ai/v1
```

CopyCopied!

鉴权

Service API 使用 `API-Key` 进行鉴权。**强烈建议开发者把 `API-Key` 放在后端存储，而非分享或者放在客户端存储，以免 `API-Key` 泄露，导致财产损失。** 所有 API 请求都应在 `Authorization` HTTP Header 中包含您的 `API-Key`，如下所示：

Code

```
Authorization: Bearer {API_KEY}
```

CopyCopied!

`POST`/chat-messages

发送对话消息

创建会话消息。

Request Body

	•
Name	•
query	•
Type	•
	•

string

Description

用户输入/提问内容。

Name

inputs

Type

object

Description

允许传入 App 定义的各变量值。inputs 参数包含了多组键值对（Key/Value pairs），每组的键对应一个特定变量，每组的值则是该变量的具体值。如果变量是文件类型，请指定一个包含以下 files 中所述键的对象。默认 {}

Name

response_mode

Type

string

Description

- **streaming** 流式模式（推荐）。基于 SSE（[Server-Sent Events](#)）实现类似打字机输出方式的流式返回。
- **blocking** 阻塞模式，等待执行完毕后返回结果。（请求若流程较长可能会被中断）。由于 Cloudflare 限制，请求会在 100 秒超时无返回后中断。

Name

user

Type

string

Description

用户标识，用于定义终端用户的身份，方便检索、统计。 由开发者定义规则，需保证用户标识在应用内唯一。服务 API 不会共享 WebApp 创建的对话。

Name

conversation_id

Type

string

Description

（选填）会话 ID，需要基于之前的聊天记录继续对话，必须传之前消息的 conversation_id。

Name

files

Type

array[object]

Description

文件列表，适用于传入文件结合文本理解并回答问题，仅当模型支持 Vision 能力时可用。

- `type` (string) 支持类型：
 - `document` 具体类型包含：'TXT', 'MD', 'MARKDOWN', 'PDF', 'HTML', 'XLSX', 'XLS', 'DOCX', 'CSV', 'EML', 'MSG', 'PPTX', 'PPT', 'XML', 'EPUB'
 - `image` 具体类型包含：'JPG', 'JPEG', 'PNG', 'GIF', 'WEBP', 'SVG'
 - `audio` 具体类型包含：'MP3', 'M4A', 'WAV', 'WEBM', 'AMR'
 - `video` 具体类型包含：'MP4', 'MOV', 'MPEG', 'MPGA'
 - `custom` 具体类型包含：其他文件类型
- `transfer_method` (string) 传递方式：
 - `remote_url`: 图片地址。
 - `local_file`: 上传文件。
- `url` 图片地址。（仅当传递方式为 `remote_url` 时）。
- `upload_file_id` 上传文件 ID。（仅当传递方式为 `local_file` 时）。

Name

`auto_generate_name`

Type

`bool`

Description

（选填）自动生成标题，默认 `true`。若设置为 `false`，则可通过调用会话重命名接口并设置 `auto_generate` 为 `true` 实现异步生成标题。

Response

当 `response_mode` 为 `blocking` 时，返回 `ChatCompletionResponse` object。当 `response_mode` 为 `streaming` 时，返回 `ChunkChatCompletionResponse` object 流式序列。

ChatCompletionResponse

返回完整的 App 结果，Content-Type 为 application/json。

- `event` (string) 事件类型，固定为 `message`
- `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
- `id` (string) 唯一 ID
- `message_id` (string) 消息唯一 ID
- `conversation_id` (string) 会话 ID

- `mode` (string) App 模式，固定为 `chat`
- `answer` (string) 完整回复内容
- `metadata` (object) 元数据
 - `usage` (Usage) 模型用量信息
 - `retriever_resources` (array[RetrieverResource]) 引用和归属分段列表
- `created_at` (int) 消息创建时间戳，如：1705395332

ChunkChatCompletionResponse

返回 App 输出的流式块，Content-Type 为 `text/event-stream`。每个流式块均为 `data:` 开头，块之间以 `\n\n` 即两个换行符分隔，如下所示：

```
data: {"event": "message", "task_id":
"900bbd43-dc0b-4383-a372-aa6e6c414227", "id":
"663c5084-a254-4040-8ad3-51f2a3c1a77c", "answer": "Hi",
"created_at": 1705398420}\n\n
```

CopyCopied!

流式块中根据 `event` 不同，结构也不同：

- `event: message` LLM 返回文本块事件，即：完整的文本以分块的方式输出。
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `message_id` (string) 消息唯一 ID
 - `conversation_id` (string) 会话 ID
 - `answer` (string) LLM 返回文本块内容
 - `created_at` (int) 创建时间戳，如：1705395332
- `event: message_file` 文件事件，表示有新文件需要展示
 - `id` (string) 文件唯一 ID
 - `type` (string) 文件类型，目前仅为 `image`
 - `belongs_to` (string) 文件归属，`user` 或 `assistant`，该接口返回仅为 `assistant`
 - `url` (string) 文件访问地址
 - `conversation_id` (string) 会话 ID
- `event: message_end` 消息结束事件，收到此事件则代表流式返回结束。
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `message_id` (string) 消息唯一 ID
 - `conversation_id` (string) 会话 ID
 - `metadata` (object) 元数据
 - `usage` (Usage) 模型用量信息
 - `retriever_resources` (array[RetrieverResource]) 引用和归属分段列表
- `event: tts_message` TTS 音频流事件，即：语音合成输出。内容是 `Mp3` 格式的音频块，使用 `base64` 编码后的字符串，播放的时候直接解码即可。(开启自动播放才有此消息)
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `message_id` (string) 消息唯一 ID
 - `audio` (string) 语音合成之后的音频块使用 `Base64` 编码之后的文本内容，播放的时候直接 `base64` 解码送入播放器即可
 - `created_at` (int) 创建时间戳，如：1705395332
- `event: tts_message_end` TTS 音频流结束事件，收到这个事件表示音频流返回结束。
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口

- `message_id` (string) 消息唯一 ID
 - `audio` (string) 结束事件是没有音频的，所以这里是空字符串
 - `created_at` (int) 创建时间戳，如：1705395332
- event: `message_replace` 消息内容替换事件。 开启内容审查和审查输出内容时，若命中了审查条件，则会通过此事件替换消息内容为预设回复。
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `message_id` (string) 消息唯一 ID
 - `conversation_id` (string) 会话 ID
 - `answer` (string) 替换内容（直接替换 LLM 所有回复文本）
 - `created_at` (int) 创建时间戳，如：1705395332
- event: `workflow_started` workflow 开始执行
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `workflow_run_id` (string) workflow 执行 ID
 - `event` (string) 固定为 `workflow_started`
 - `data` (object) 详细内容
 - `id` (string) workflow 执行 ID
 - `workflow_id` (string) 关联 Workflow ID
 - `created_at` (timestamp) 开始时间
- event: `node_started` node 开始执行
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `workflow_run_id` (string) workflow 执行 ID
 - `event` (string) 固定为 `node_started`
 - `data` (object) 详细内容
 - `id` (string) workflow 执行 ID
 - `node_id` (string) 节点 ID
 - `node_type` (string) 节点类型
 - `title` (string) 节点名称
 - `index` (int) 执行序号，用于展示 Tracing Node 顺序
 - `predecessor_node_id` (string) 前置节点 ID，用于画布展示执行路径
 - `inputs` (object) 节点中所有使用到的前置节点变量内容
 - `created_at` (timestamp) 开始时间
- event: `node_finished` node 执行结束，成功失败同一事件中不同状态
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `workflow_run_id` (string) workflow 执行 ID
 - `event` (string) 固定为 `node_finished`
 - `data` (object) 详细内容
 - `id` (string) node 执行 ID
 - `node_id` (string) 节点 ID
 - `index` (int) 执行序号，用于展示 Tracing Node 顺序
 - `predecessor_node_id` (string) optional 前置节点 ID，用于画布展示执行路径
 - `inputs` (object) 节点中所有使用到的前置节点变量内容
 - `process_data` (json) Optional 节点过程数据
 - `outputs` (json) Optional 输出内容
 - `status` (string) 执行状态 `running` / `succeeded` / `failed` / `stopped`

- `error` (string) Optional 错误原因
- `elapsed_time` (float) Optional 耗时(s)
- `execution_metadata` (json) 元数据
 - `total_tokens` (int) optional 总使用 tokens
 - `total_price` (decimal) optional 总费用
 - `currency` (string) optional 货币，如 `USD` / `RMB`
- `created_at` (timestamp) 开始时间
- `event: workflow_finished` workflow 执行结束，成功失败同一事件中不同状态
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `workflow_run_id` (string) workflow 执行 ID
 - `event` (string) 固定为 `workflow_finished`
 - `data` (object) 详细内容
 - `id` (string) workflow 执行 ID
 - `workflow_id` (string) 关联 Workflow ID
 - `status` (string) 执行状态 `running` / `succeeded` / `failed` / `stopped`
 - `outputs` (json) Optional 输出内容
 - `error` (string) Optional 错误原因
 - `elapsed_time` (float) Optional 耗时(s)
 - `total_tokens` (int) Optional 总使用 tokens
 - `total_steps` (int) 总步数（冗余），默认 0
 - `created_at` (timestamp) 开始时间
 - `finished_at` (timestamp) 结束时间
- `event: error` 流式输出过程中出现的异常会以 `stream event` 形式输出，收到异常事件后即结束。
 - `task_id` (string) 任务 ID，用于请求跟踪和下方的停止响应接口
 - `message_id` (string) 消息唯一 ID
 - `status` (int) HTTP 状态码
 - `code` (string) 错误码
 - `message` (string) 错误消息
- `event: ping` 每 10s 一次的 ping 事件，保持连接存活。

Errors

- 404, 对话不存在
- 400, `invalid_param`, 传入参数异常
- 400, `app_unavailable`, App 配置不可用
- 400, `provider_not_initialize`, 无可模型凭据配置
- 400, `provider_quota_exceeded`, 模型调用额度不足
- 400, `model_currently_not_support`, 当前模型不可用
- 400, `completion_request_error`, 文本生成失败
- 500, 服务内部异常

Request

POST

/chat-messages

```
curl -X POST 'https://api.dify.ai/v1/chat-messages' \
```

```
--header 'Authorization: Bearer {api_key}' \
--header 'Content-Type: application/json' \
--data-raw '{
  "inputs": {},
  "query": "What are the specs of the iPhone 13 Pro Max?",
  "response_mode": "streaming",
  "conversation_id": "",
  "user": "abc-123",
  "files": [
    {
      "type": "image",
      "transfer_method": "remote_url",
      "url": "https://cloud.dify.ai/logo/logo-site.png"
    }
  ]
}'
```

CopyCopied!

阻塞模式

Response

```
{
  "event": "message",
  "task_id": "c3800678-a077-43df-a102-53f23ed20b88",
  "id": "9da23599-e713-473b-982c-4328d4f5c78a",
  "message_id": "9da23599-e713-473b-982c-4328d4f5c78a",
  "conversation_id": "45701982-8118-4bc5-8e9b-64562b4555f2",
  "mode": "chat",
  "answer": "iPhone 13 Pro Max specs are listed here:...",
  "metadata": {
    "usage": {
      "prompt_tokens": 1033,
      "prompt_unit_price": "0.001",
      "prompt_price_unit": "0.001",
      "prompt_price": "0.0010330",
      "completion_tokens": 128,
      "completion_unit_price": "0.002",
      "completion_price_unit": "0.001",
      "completion_price": "0.0002560",
      "total_tokens": 1161,
      "total_price": "0.0012890",
      "currency": "USD",
      "latency": 0.7682376249867957
    }
  },
}
```



```

    "retriever_resources": [
      {
        "position": 1,
        "dataset_id": "101b4c97-fc2e-463c-90b1-5261a4cdcafb",
        "dataset_name": "iPhone",
        "document_id": "8dd1ad74-0b5f-4175-b735-7d98bbbb4e00",
        "document_name": "iPhone List",
        "segment_id": "ed599c7f-2766-4294-9d1d-e5235a61270a",
        "score": 0.98457545,
        "content": "\"Model\", \"Release Date\", \"Display  
Size\", \"Resolution\", \"Processor\", \"RAM\", \"Storage\", \"Camera\", \"Ba  
ttery\", \"Operating System\"\\n\\n\"iPhone 13 Pro Max\", \"September 24,  
2021\", \"6.7 inch\", \"1284 x 2778\", \"Hexa-core (2x3.23 GHz Avalanche +  
4x1.82 GHz Blizzard)\", \"6 GB\", \"128, 256, 512 GB, 1TB\", \"12 MP\", \"4352  
mAh\", \"iOS 15\""
      }
    ]
  },
  "created_at": 1705407629
}

```

CopyCopied!

流式模式

Response

```

data: {"event": "workflow_started", "task_id":
"5ad4cb98-f0c7-4085-b384-88c403be6290", "workflow_run_id":
"5ad498-f0c7-4085-b384-88cbe6290", "data": {"id":
"5ad498-f0c7-4085-b384-88cbe6290", "workflow_id": "dfjasklfjdslag",
"created_at": 1679586595}}
data: {"event": "node_started", "task_id":
"5ad4cb98-f0c7-4085-b384-88c403be6290", "workflow_run_id":
"5ad498-f0c7-4085-b384-88cbe6290", "data": {"id":
"5ad498-f0c7-4085-b384-88cbe6290", "node_id": "dfjasklfjdslag",
"node_type": "start", "title": "Start", "index": 0, "predecessor_node_id":
"fdljewklfklgejlglsd", "inputs": {}, "created_at": 1679586595}}
data: {"event": "node_finished", "task_id":
"5ad4cb98-f0c7-4085-b384-88c403be6290", "workflow_run_id":
"5ad498-f0c7-4085-b384-88cbe6290", "data": {"id":
"5ad498-f0c7-4085-b384-88cbe6290", "node_id": "dfjasklfjdslag",
"node_type": "start", "title": "Start", "index": 0, "predecessor_node_id":
"fdljewklfklgejlglsd", "inputs": {}, "outputs": {}, "status": "succeeded",
"elapsed_time": 0.324, "execution_metadata": {"total_tokens": 63127864,
"total_price": 2.378, "currency": "USD"}, "created_at": 1679586595}}

```

```
data: {"event": "workflow_finished", "task_id":  
"5ad4cb98-f0c7-4085-b384-88c403be6290", "workflow_run_id":  
"5ad498-f0c7-4085-b384-88cbe6290", "data": {"id":  
"5ad498-f0c7-4085-b384-88cbe6290", "workflow_id": "dfjasklfjdslag",  
"outputs": {}, "status": "succeeded", "elapsed_time": 0.324, "total_tokens"  
63127864, "total_steps": "1", "created_at": 1679586595, "finished_at":  
1679976595}}  
data: {"event": "message", "message_id":  
"5ad4cb98-f0c7-4085-b384-88c403be6290", "conversation_id":  
"45701982-8118-4bc5-8e9b-64562b4555f2", "answer": " I", "created_at":  
1679586595}  
data: {"event": "message", "message_id":  
"5ad4cb98-f0c7-4085-b384-88c403be6290", "conversation_id":  
"45701982-8118-4bc5-8e9b-64562b4555f2", "answer": "'m", "created_at":  
1679586595}  
data: {"event": "message", "message_id":  
"5ad4cb98-f0c7-4085-b384-88c403be6290", "conversation_id":  
"45701982-8118-4bc5-8e9b-64562b4555f2", "answer": " glad", "created_at":  
1679586595}  
data: {"event": "message", "message_id":  
"5ad4cb98-f0c7-4085-b384-88c403be6290", "conversation_id":  
"45701982-8118-4bc5-8e9b-64562b4555f2", "answer": " to", "created_at":  
1679586595}  
data: {"event": "message", "message_id" :  
"5ad4cb98-f0c7-4085-b384-88c403be6290", "conversation_id":  
"45701982-8118-4bc5-8e9b-64562b4555f2", "answer": " meet", "created_at":  
1679586595}  
data: {"event": "message", "message_id" :  
"5ad4cb98-f0c7-4085-b384-88c403be6290", "conversation_id":  
"45701982-8118-4bc5-8e9b-64562b4555f2", "answer": " you", "created_at":  
1679586595}  
data: {"event": "message_end", "id":  
"5e52ce04-874b-4d27-9045-b3bc80def685", "conversation_id":  
"45701982-8118-4bc5-8e9b-64562b4555f2", "metadata": {"usage":  
{"prompt_tokens": 1033, "prompt_unit_price": "0.001", "prompt_price_unit":  
"0.001", "prompt_price": "0.0010330", "completion_tokens": 135,  
"completion_unit_price": "0.002", "completion_price_unit": "0.001",  
"completion_price": "0.0002700", "total_tokens": 1168, "total_price":  
"0.0013030", "currency": "USD", "latency": 1.381760165997548},  
"retriever_resources": [{"position": 1, "dataset_id":  
"101b4c97-fc2e-463c-90b1-5261a4cdcafb", "dataset_name": "iPhone",  
"document_id": "8dd1ad74-0b5f-4175-b735-7d98bbbb4e00", "document_name":  
"iPhone List", "segment_id": "ed599c7f-2766-4294-9d1d-e5235a61270a",  
"score": 0.98457545, "content": "\"Model\", \"Release Date\", \"Display
```

```
Size\","Resolution\","Processor\","RAM\","Storage\","Camera\","Ba
ttery\","Operating System"\n"iPhone 13 Pro Max\","September 24,
2021\","6.7 inch\","1284 x 2778\","Hexa-core (2x3.23 GHz Avalanche +
4x1.82 GHz Blizzard)\","6 GB\","128, 256, 512 GB, 1TB\","12 MP\","4352
mAh\","iOS 15\""}}}}
```

[illegible]

```
data: {"event": "tts_message_end", "conversation_id":  
"23dd85f3-1a41-4ea0-b7a9-062734ccfaf9", "message_id":  
"a8bdc41c-13b2-4c18-bfd9-054b9803038c", "created_at": 1721205487,  
"task id": "3bf8a0bb-e73b-4690-9e66-4e429bad8ee7", "audio": ""}
```

CopyCopied!

POST/files/upload

上传文件

上传文件并在发送消息时使用，可实现图文多模态理解。支持您的应用程序所支持的所有格式。上传的文件仅供当前终端用户使用。

Request Body

该接口需使用 `multipart/form-data` 进行请求。

Name	•
file	•
Type	•
file	•
Description	•
要上传的文件。	•
	•
	•
	•
Name	•

user

Type

string

Description

用户标识，用于定义终端用户的身份，必须和发送消息接口传入 `user` 保持一致。

Response

成功上传后，服务器会返回文件的 ID 和相关信息。

- `id` (uuid) ID
- `name` (string) 文件名
- `size` (int) 文件大小 (byte)
- `extension` (string) 文件后缀
- `mime_type` (string) 文件 mime-type
- `created_by` (uuid) 上传人 ID
- `created_at` (timestamp) 上传时间

Errors

- 400, `no_file_uploaded`, 必须提供文件
- 400, `too_many_files`, 目前只接受一个文件
- 400, `unsupported_preview`, 该文件不支持预览
- 400, `unsupported_estimate`, 该文件不支持估算
- 413, `file_too_large`, 文件太大
- 415, `unsupported_file_type`, 不支持的扩展名, 当前只接受文档类文件
- 503, `s3_connection_failed`, 无法连接到 S3 服务
- 503, `s3_permission_denied`, 无权限上传文件到 S3
- 503, `s3_file_too_large`, 文件超出 S3 大小限制

Request

POST

/files/upload

```
curl -X POST 'https://api.dify.ai/v1/files/upload' \
--header 'Authorization: Bearer {api_key}' \
--form 'file=@localfile;type=image/[png|jpeg|jpg|webp|gif]' \
--form 'user=abc-123'
```

CopyCopied!

Response

```
{
  "id": "72fa9618-8f89-4a37-9b33-7e1178a24a67",
  "name": "example.png",
  "size": 1024,
  "extension": "png",
  "mime_type": "image/png",
  "created_by": 123,
  "created_at": 1577836800,
}
```

CopyCopied!

[POST](#)/chat-messages/:task_id/stop

停止响应

仅支持流式模式。

Path

- `task_id` (string) 任务 ID，可在流式返回 `Chunk` 中获取

Request Body

- `user` (string) Required 用户标识，用于定义终端用户的身份，必须和发送消息接口传入 `user` 保持一致。API 无法访问 WebApp 创建的会话。

Response

- `result` (string) 固定返回 `success`

Request

[POST](#)

/chat-messages/:task_id/stop

`curl -X POST`

```
'https://api.dify.ai/v1/chat-messages/:task_id/stop' \
-H 'Authorization: Bearer {api_key}' \
-H 'Content-Type: application/json' \
--data-raw '{ "user": "abc-123" }'
```

CopyCopied!

Response

```
{
  "result": "success"
}
```

CopyCopied!

[POST](#) /messages/:message_id/feedbacks

消息反馈（点赞）

消息终端用户反馈、点赞，方便应用开发者优化输出预期。

Path Params

Name	•
message_id	•
Type	•
string	•
Description	•
消息 ID	•
	•

Request Body

Name	•
rating	•
Type	•
string	•
Description	•
点赞 like, 点踩 dislike, 撤销点赞 null	•
	•
Name	• <div></div>
	• <div></div>

user	<div><div></div></div>
Type	<div><div></div></div>
string	<div><div></div></div>
Description	<div><div><div></div><div></div></div></div>
用户标识，由开发者定义规则，需保证用户标识在应用内唯一。	<div><div><div></div><div></div><div></div></div></div>
Name	<div><div></div></div>
content	<div><div></div></div>
Type	<div><div></div></div>
string	<div><div></div></div>
Description	<div><div><div></div><div></div></div></div>
消息反馈的具体信息。	<div><div><div></div><div></div></div></div>

Response

- result (string) 固定返回 success

Request

```
POST
/messages/:message_id/feedbacks

curl -X POST
'https://api.dify.ai/v1/messages/:message_id/feedbacks \
--header 'Authorization: Bearer {api_key}' \
--header 'Content-Type: application/json' \
--data-raw '{
  "rating": "like",
  "user": "abc-123",
  "content": "message feedback information"
}'

CopyCopied!
```

Response

```
{  
  "result": "success"  
}
```

CopyCopied!

GET/app/feedbacks

获取 APP 的消息点赞和反馈

获取应用的终端用户反馈、点赞。

Query

Name	•
page	•
Type	•
string	•
Description	•
(选填) 分页，默认值：1	•
Name	•
limit	•
Type	•
string	•
Description	•
(选填) 每页数量，默认值：20	•

-
- **Response**
 - **data** (List) 返回该 APP 的点赞、反馈列表。

Request

```
GET
/app/feedbacks
curl -X GET
'https://api.dify.ai/v1/app/feedbacks?page=1&limit=20'
CopyCopied!
```

Response

```
{
  "data": [
    {
      "id": "8c0fbed8-e2f9-49ff-9f0e-15a35bdd0e25",
      "app_id": "f252d396-fe48-450e-94ec-e184218e7346",
      "conversation_id": "2397604b-9deb-430e-b285-4726e51fd62d",
      "message_id": "709c0b0f-0a96-4a4e-91a4-ec0889937b11",
      "rating": "like",
      "content": "message feedback information-3",
      "from_source": "user",
      "from_end_user_id": "74286412-9a1a-42c1-929c-01edb1d381d5",
      "from_account_id": null,
      "created_at": "2025-04-24T09:24:38",
      "updated_at": "2025-04-24T09:24:38"
    }
  ]
}
```

```
CopyCopied!
GET/messages/{message_id}/suggested
```

获取下一轮建议问题列表

获取下一轮建议问题列表。

Path Params

Name	

message_id	•
Type	•
string	•
Description	•
Message ID	•
	•
Query	•
Name	•
user	•
Type	•
string	•
Description	•
用户标识，由开发者定义规则，需保证用户标识在应用内唯一。	•
	•

Request

```
GET
/messages/{message_id}/suggested

curl --location --request GET
'https://api.dify.ai/v1/messages/{message_id}/suggested?user=abc-123' \
--header 'Authorization: Bearer ENTER-YOUR-SECRET-KEY' \
--header 'Content-Type: application/json'
```

CopyCopied!

Response

```
{
  "result": "success",
  "data": [
```

CopyCopied!

GET/messages

获取会话历史消息

滚动加载形式返回历史聊天记录，第一页返回最新 **limit** 条，即：倒序返回。

Query

Name	•
conversation_id	•
Type	•
string	•
Description	•
会话 ID	•
	•
	•
Name	•
user	•
Type	•
string	•
Description	•

用户标识，由开发者定义规则，需保证用户标识在应用内唯一。

	<div>•</div> <div>•</div> <div>•</div>
Name	
<code>first_id</code>	<div>•</div>
Type	<div>•</div>
<code>string</code>	<div>•</div>
Description	<div>•</div> <div>•</div>
当前页第一条聊天记录 ID，默认 <code>null</code>	<div>•</div> <div>•</div> <div>•</div>
Name	<div>•</div>
<code>limit</code>	<div>•</div>
Type	<div>•</div>
<code>int</code>	<div>•</div>
Description	<div>•</div> <div>•</div>
一次请求返回多少条聊天记录，默认 20 条。	<div>•</div> <div>•</div>

Response

- `data` (array[object]) 消息列表
 - `id` (string) 消息 ID
 - `conversation_id` (string) 会话 ID
 - `inputs` (object) 用户输入参数。
 - `query` (string) 用户输入 / 提问内容。
 - `message_files` (array[object]) 消息文件
 - `id` (string) ID
 - `type` (string) 文件类型，image 图片
 - `url` (string) 预览图片地址
 - `belongs_to` (string) 文件归属方，user 或 assistant

- `answer` (string) 回答消息内容
- `created_at` (timestamp) 创建时间
- `feedback` (object) 反馈信息
 - `rating` (string) 点赞 like / 点踩 dislike
- `retriever_resources` (array[RetrieverResource]) 引用和归属分段列表
- `has_more` (bool) 是否存在下一页
- `limit` (int) 返回条数，若传入超过系统限制，返回系统限制数量

Request Example

Request

```
GET
/messages
curl -X GET
'https://api.dify.ai/v1/messages?user=abc-123&conversation_id=
' \
--header 'Authorization: Bearer {api_key}'
CopyCopied!
```

Response Example

Response

```
{
  "limit": 20,
  "has_more": false,
  "data": [
    {
      "id": "a076a87f-31e5-48dc-b452-0061adbbc922",
      "conversation_id": "cd78daf6-f9e4-4463-9ff2-54257230a0ce",
      "inputs": {
        "name": "dify"
      },
      "query": "iphone 13 pro",
      "answer": "The iPhone 13 Pro, released on September 24, 2021, features a 6.1-inch display with a resolution of 1170 x 2532. It is equipped with a Hexa-core (2x3.23 GHz Avalanche + 4x1.82 GHz Blizzard) processor, 6 GB of RAM, and offers storage options of 128 GB, 256 GB, 512 GB, and 1 TB. The camera is 12 MP, the battery capacity is 3095 mAh, and it runs on iOS 15.",
      "message_files": [],
      "feedback": null,
      "retriever_resources": [
        {
          "position": 1,
          "dataset_id": "101b4c97-fc2e-463c-90b1-5261a4cdcafb",
```

```

        "dataset_name": "iPhone",
        "document_id": "8dd1ad74-0b5f-4175-b735-7d98bbbb4e00",
        "document_name": "iPhone List",
        "segment_id": "ed599c7f-2766-4294-9d1d-e5235a61270a",
        "score": 0.98457545,
        "content": "\"Model\\\", \"Release Date\\\", \"Display
Size\\\", \"Resolution\\\", \"Processor\\\", \"RAM\\\", \"Storage\\\", \"Camera\\\", \"Ba
ttery\\\", \"Operating System\\\"\\n\\\"iPhone 13 Pro Max\\\", \"September 24,
2021\\\", \"6.7 inch\\\", \"1284 x 2778\\\", \"Hexa-core (2x3.23 GHz Avalanche +
4x1.82 GHz Blizzard)\\\", \"6 GB\\\", \"128, 256, 512 GB, 1TB\\\", \"12 MP\\\", \"4352
mAh\\\", \"iOS 15\\\"\"
    }
],
    "created_at": 1705569239
}
]
}
CopyCopied!

```

Response Example (智能助手)

Response

```

{
  "limit": 20,
  "has_more": false,
  "data": [
    {
      "id": "d35e006c-7c4d-458f-9142-be4930abdf94",
      "conversation_id": "957c068b-f258-4f89-ba10-6e8a0361c457",
      "inputs": {},
      "query": "draw a cat",
      "answer": "I have generated an image of a cat for you. Please check
your messages to view the image.",
      "message_files": [
        {
          "id": "976990d2-5294-47e6-8f14-7356ba9d2d76",
          "type": "image",
          "url":
"http://127.0.0.1:5001/files/tools/976990d2-5294-47e6-8f14-7356ba9d2d76.
png?timestamp=1705988524&nonce=55df3f9f7311a9acd91bf074cd524092&sign=z4
3nMS01L2HBvoqADLkRxr7Biz0fkjeDstnJiCK1zh8=",
          "belongs_to": "assistant"
        }
      ],
    }
  ],
}

```

```
"feedback": null,
"retriever_resources": [],
"created_at": 1705988187
}
]
```

CopyCopied!

[GET](#)/conversations

获取会话列表

获取当前用户的会话列表，默认返回最近的 20 条。

Query

Name	<div><div></div><div></div><div></div></div>
user	<div><div></div><div></div><div></div></div>
Type	<div><div></div><div></div><div></div></div>
string	<div><div></div><div></div><div></div></div>
Description	<div><div></div><div></div><div></div></div>
用户标识，由开发者定义规则，需保证用户标识在应用内唯一。	
Name	<div><div></div><div></div><div></div></div>
last_id	<div><div></div><div></div><div></div></div>
Type	<div><div></div><div></div><div></div></div>
string	<div><div></div><div></div><div></div></div>
Description	<div><div></div><div></div><div></div></div>
(选填) 当前页最后面一条记录的 ID，默认 null	

Name

`limit`

Type

`int`

Description

(选填)一次请求返回多少条记录,默认 20 条,最大 100 条,最小 1 条。

Name

`sort_by`

Type

`string`

Description

(选填) 排序字段, 默认 `-updated_at`(按更新时间倒序排列)

- 可选值: `created_at`, `-created_at`, `updated_at`, `-updated_at`
- 字段前面的符号代表顺序或倒序, `-`代表倒序

Response

- `data` (array[object]) 会话列表
 - `id` (string) 会话 ID
 - `name` (string) 会话名称, 默认由大语言模型生成。
 - `inputs` (object) 用户输入参数。
 - `status` (string) 会话状态
 - `introduction` (string) 开场白
 - `created_at` (timestamp) 创建时间
 - `updated_at` (timestamp) 更新时间

- `has_more` (bool)
- `limit` (int) 返回条数，若传入超过系统限制，返回系统限制数量

Request

```
GET
/conversations

curl -X GET
'https://api.dify.ai/v1/conversations?user=abc-123&last_id=&limit=20'
--header 'Authorization: Bearer {api_key}'

CopyCopied!
```

Response

```
{
  "limit": 20,
  "has_more": false,
  "data": [
    {
      "id": "10799fb8-64f7-4296-bbf7-b42bfbe0ae54",
      "name": "New chat",
      "inputs": {
        "book": "book",
        "myName": "Lucy"
      },
      "status": "normal",
      "created_at": 1679667915,
      "updated_at": 1679667915
    },
    {
      "id": "hSIhXBhNe8X1d8Et"
      // ...
    }
  ]
}
```

```
CopyCopied!
```

DELETE/conversations/:conversation_id

删除会话

删除会话。

Path

- `conversation_id` (string) 会话 ID

Request Body

Name	•
<code>user</code>	•
Type	•
<code>string</code>	•
Description	•
用户标识，由开发者定义规则，需保证用户标识在应用内唯一。	•
	•

Response

- `result` (string) 固定返回 success

Request

```
DELETE
/conversations/:conversation_id
curl -X DELETE 'https://api.dify.ai/v1/conversations/:conversation_id' \
--header 'Authorization: Bearer {api_key}' \
--header 'Content-Type: application/json' \
--data-raw '{
"user": "abc-123"
}'
CopyCopied!
```

Response

```
204 No Content
CopyCopied!
POST/conversations/:conversation_id/name
```

会话重命名

对话进行重命名，会话名称用于显示在支持多会话的客户端上。

Path

- conversation_id (string) 会话 ID

Request Body

	•
Name	•
name	•
Type	•
string	•
Description	•
	•
(选填) 名称，若 auto_generate 为 true 时，该参数可不传。	•
	•
	•
Name	•
	•
auto_generate	•
Type	•
bool	•
Description	•
	•
(选填) 自动生成标题，默认 false。	•
	•
	•
Name	•
user	•
Type	•
string	•
	•

Description

用户标识，由开发者定义规则，需保证用户标识在应用内唯一。

Response

- `id` (string) 会话 ID
- `name` (string) 会话名称
- `inputs` (object) 用户输入参数
- `status` (string) 会话状态
- `introduction` (string) 开场白
- `created_at` (timestamp) 创建时间
- `updated_at` (timestamp) 更新时间

Request

POST

/conversations/:conversation_id/name

```
curl -X POST
```

```
'https://api.dify.ai/v1/conversations/:conversation_id/name' \
--header 'Authorization: Bearer {api_key}' \
--header 'Content-Type: application/json' \
--data-raw '{
  "name": "",
  "auto_generate": true,
  "user": "abc-123"
}'
```

CopyCopied!

Response

```
{
  "id": "34d511d5-56de-4f16-a997-57b379508443",
  "name": "hello",
  "inputs": {},
  "status": "normal",
  "introduction": "",
  "created_at": 1732731141,
  "updated_at": 1732734510
}
```

CopyCopied!

GET/conversations/:conversation_id/variables

获取对话变量

从特定对话中检索变量。此端点对于提取对话过程中捕获的结构化数据非常有用。

路径参数

	•
Name	•
conversation_id	•
Type	•
string	•
Description	•
要从中检索变量的对话 ID。	•
	•

查询参数

	•
Name	•
user	•
Type	•
string	•
Description	•
用户标识符，由开发人员定义的规则，在应用程序内必须唯一。	•
	•
	•
Name	•
last_id	•
Type	•

string

Description

(选填) 当前页最后面一条记录的 ID, 默认 null

Name

limit

Type

int

Description

(选填)一次请求返回多少条记录, 默认 20 条, 最大 100 条, 最小 1 条。

响应

- `limit` (int) 每页项目数
- `has_more` (bool) 是否有更多项目
- `data` (array[object]) 变量列表
 - `id` (string) 变量 ID
 - `name` (string) 变量名称
 - `value_type` (string) 变量类型 (字符串、数字、布尔等)
 - `value` (string) 变量值
 - `description` (string) 变量描述
 - `created_at` (int) 创建时间戳
 - `updated_at` (int) 最后更新时间戳

错误

- 404, `conversation_not_exists`, 对话不存在

Request

GET

/conversations/:conversation_id/variables

```
curl -X GET
'https://api.dify.ai/v1/conversations/{conversation_id}/variables?user=abc-123' \
--header 'Authorization: Bearer {api_key}'
CopyCopied!
```

Request with variable name filter

```
curl -X GET
'${props.appDetail.api_base_url}/conversations/{conversation_id}/variables?user=abc-123&variable_name=customer_name' \
--header 'Authorization: Bearer {api_key}'
CopyCopied!
```

Response

```
{
  "limit": 100,
  "has_more": false,
  "data": [
    {
      "id": "variable-uuid-1",
      "name": "customer_name",
      "value_type": "string",
      "value": "John Doe",
      "description": "客户名称（从对话中提取）",
      "created_at": 1650000000000,
      "updated_at": 1650000000000
    },
    {
      "id": "variable-uuid-2",
      "name": "order_details",
      "value_type": "json",
      "value": "{\"product\":\"Widget\",\"quantity\":5,\"price\":19.99}",
      "description": "客户的订单详情",
      "created_at": 1650000000000,
      "updated_at": 1650000000000
    }
  ]
}
```

CopyCopied!

[POST](#)/audio-to-text

[语音转文字](#)

Request Body

该接口需使用 `multipart/form-data` 进行请求。

Name	<div><div></div><div></div><div></div></div>
file	<div><div></div><div></div><div></div></div>
Type	<div><div></div><div></div><div></div></div>
file	<div><div></div><div></div><div></div></div>
Description	<div><div></div><div></div><div></div></div>
语音文件。支持格式: <code>['mp3', 'mp4', 'mpeg', 'mpga', 'm4a', 'wav', 'webm']</code> 文件大小限制: 15MB	
Name	<div><div></div><div></div><div></div></div>
user	<div><div></div><div></div><div></div></div>
Type	<div><div></div><div></div><div></div></div>
string	<div><div></div><div></div><div></div></div>
Description	<div><div></div><div></div><div></div></div>
用户标识, 由开发者定义规则, 需保证用户标识在应用内唯一。	

Response

- `text` (string) 输出文字

Request

```
POST
/audio-to-text
curl -X POST 'https://api.dify.ai/v1/audio-to-text' \
--header 'Authorization: Bearer {api_key}' \
```



```
--form
'file=@localfile;type=audio/[mp3|mp4|mpeg|mpga|m4a|wav|webm]
CopyCopied!
```

Response

```
{
  "text": "hello"
}
CopyCopied!
```

[POST](#)/text-to-audio

文字转语音

文字转语音。

Request Body

Name	<div><div></div><div></div></div>
message_id	<div><div></div><div></div></div>
Type	<div><div></div><div></div></div>
str	<div><div></div><div></div></div>
Description	<div><div></div><div></div></div>
<p>Dify 生成的文本消息，那么直接传递生成的 message-id 即可，后台会通过 message_id 查找相应内容直接合成为语音信息。如果同时传 message_id 和 text，优先使用 message_id。</p>	
Name	<div><div></div><div></div></div>
text	<div><div></div><div></div></div>
Type	<div><div></div><div></div></div>
str	<div><div></div><div></div></div>

Description

语音生成内容。如果没有传 message-id 的话，则会使用这个字段的内容

Name

user

Type

string

Description

用户标识，由开发者定义规则，需保证用户标识在应用内唯一。

Request

```
POST
/text-to-audio

curl -o text-to-audio.mp3 -X POST
'https://api.dify.ai/v1/text-to-audio' \
--header 'Authorization: Bearer {api_key}' \
--header 'Content-Type: application/json' \
--data-raw '{
  "message_id": "5ad4cb98-f0c7-4085-b384-88c403be6290",
  "text": "你好 Dify",
  "user": "abc-123"
}'
```

CopyCopied!

headers

```
{
  "Content-Type": "audio/wav"
}
```

CopyCopied!

GET/info

获取应用基本信息

用于获取应用的基本信息

Response

- `name` (string) 应用名称
- `description` (string) 应用描述
- `tags` (array[string]) 应用标签

Request

```
GET
/info

curl -X GET 'https://api.dify.ai/v1/info' \
-H 'Authorization: Bearer {api_key}'

CopyCopied!
```

Response

```
{
  "name": "My App",
  "description": "This is my app.",
  "tags": [
    "tag1",
    "tag2"
  ],
  "mode": "advanced-chat",
  "author_name": "Dify"
}

CopyCopied!
```


[GET /parameters](#)

获取应用参数

用于进入页面一开始，获取功能开关、输入参数名称、类型及默认值等使用。

Response

- `opening_statement` (string) 开场白
- `suggested_questions` (array[string]) 开场推荐问题列表
- `suggested_questions_after_answer` (object) 启用回答后给出推荐问题。
 - `enabled` (bool) 是否开启
- `speech_to_text` (object) 语音转文本
 - `enabled` (bool) 是否开启
- `text_to_speech` (object) 文本转语音

- `enabled` (bool) 是否开启
 - `voice` (string) 语音类型
 - `language` (string) 语言
 - `autoPlay` (string) 自动播放
 - `enabled` 开启
 - `disabled` 关闭
- `retriever_resource` (object) 引用和归属
 - `enabled` (bool) 是否开启
- `annotation_reply` (object) 标记回复
 - `enabled` (bool) 是否开启
- `user_input_form` (array[object]) 用户输入表单配置
 - `text-input` (object) 文本输入控件
 - `label` (string) 控件展示标签名
 - `variable` (string) 控件 ID
 - `required` (bool) 是否必填
 - `default` (string) 默认值
 - `paragraph` (object) 段落文本输入控件
 - `label` (string) 控件展示标签名
 - `variable` (string) 控件 ID
 - `required` (bool) 是否必填
 - `default` (string) 默认值
 - `select` (object) 下拉控件
 - `label` (string) 控件展示标签名
 - `variable` (string) 控件 ID
 - `required` (bool) 是否必填
 - `default` (string) 默认值
 - `options` (array[string]) 选项值
- `file_upload` (object) 文件上传配置
 - `image` (object) 图片设置 当前仅支持图片类型: `png`, `jpg`, `jpeg`, `webp`, `gif`
 - `enabled` (bool) 是否开启
 - `number_limits` (int) 图片数量限制, 默认 3
 - `transfer_methods` (array[string]) 传递方式列表, `remote_url`, `local_file`, 必选一个
- `system_parameters` (object) 系统参数
 - `file_size_limit` (int) Document upload size limit (MB)
 - `image_file_size_limit` (int) Image file upload size limit (MB)
 - `audio_file_size_limit` (int) Audio file upload size limit (MB)
 - `video_file_size_limit` (int) Video file upload size limit (MB)

Request

GET

/parameters

```
curl -X GET 'https://api.dify.ai/v1/parameters'\
--header 'Authorization: Bearer {api_key}'
```

CopyCopied!

Response

```
{
  "introduction": "nice to meet you",
  "user_input_form": [
    {
      "text-input": {
        "label": "a",
        "variable": "a",
        "required": true,
        "max_length": 48,
        "default": ""
      }
    },
    {
      // ...
    }
  ],
  "file_upload": {
    "image": {
      "enabled": true,
      "number_limits": 3,
      "transfer_methods": [
        "remote_url",
        "local_file"
      ]
    }
  },
  "system_parameters": {
    "file_size_limit": 15,
    "image_file_size_limit": 10,
    "audio_file_size_limit": 50,
    "video_file_size_limit": 100
  }
}
```

CopyCopied!

GET/meta

获取应用 Meta 信息

用于获取工具 icon

Response

- `tool_icons(object[string])` 工具图标
 - 工具名称 (string)
 - `icon (object[string])`
 - (object) 图标
 - `background (string)` hex 格式的背景色
 - `content(string)` emoji
 - (string) 图标 URL

Request

POST

/meta

```
curl -X GET 'https://api.dify.ai/v1/meta' \
-H 'Authorization: Bearer {api_key}'
```

CopyCopied!

Response

```
{
  "tool_icons": {
    "dalle2": {
      "https://cloud.dify.ai/console/api/workspaces/current/tool-provider/bui
ltin/dalle/icon",
      "api_tool": {
        "background": "#252525",
        "content": " "
      }
    }
  }
}
```

CopyCopied!

GET/site

获取应用 WebApp 设置

用于获取应用的 WebApp 设置

Response

- `title (string)` WebApp 名称
- `chat_color_theme (string)` 聊天颜色主题, hex 格式
- `chat_color_theme_inverted (bool)` 聊天颜色主题是否反转
- `icon_type (string)` 图标类型, `emoji`-表情, `image`-图片
- `icon (string)` 图标, 如果是 `emoji` 类型, 则是 emoji 表情符号, 如果是 `image` 类型, 则是图片 URL
- `icon_background (string)` hex 格式的背景色
- `icon_url (string)` 图标 URL

- `description` (string) 描述
- `copyright` (string) 版权信息
- `privacy_policy` (string) 隐私政策链接
- `custom_disclaimer` (string) 自定义免责声明
- `default_language` (string) 默认语言
- `show_workflow_steps` (bool) 是否显示工作流详情
- `use_icon_as_answer_icon` (bool) 是否使用 WebApp 图标替换聊天中的 🤖

Request

```
POST
/meta

curl -X GET 'https://api.dify.ai/v1/site' \
-H 'Authorization: Bearer {api_key}'

CopyCopied!
```

Response

```
{
  "title": "My App",
  "chat_color_theme": "#ff4a4a",
  "chat_color_theme_inverted": false,
  "icon_type": "emoji",
  "icon": " ",
  "icon_background": "#FFEAD5",
  "icon_url": null,
  "description": "This is my app.",
  "copyright": "all rights reserved",
  "privacy_policy": "",
  "custom_disclaimer": "All generated by AI",
  "default_language": "en-US",
  "show_workflow_steps": false,
  "use_icon_as_answer_icon": false,
}

CopyCopied!
```

GET/apps/annotations

获取标注列表

Query

Name	

page

Type

string

Description

页码

Name

limit

Type

string

Description

每页数量

•

•

•

•

•

•

•

•

•

•

•

•

•

•

•

•

Request

GET

/apps/annotations

curl --location --request GET

'https://api.dify.ai/v1/apps/annotations?page=1&limit=20' \

--header 'Authorization: Bearer {api_key}'

CopyCopied!

Response

{

"data": [

{

"id": "69d48372-ad81-4c75-9c46-2ce197b4d402",

"question": "What is your name?",

"answer": "I am Dify.",

"hit_count": 0,


```
    "created_at": 1735625869
  }
],
"has_more": false,
"limit": 20,
"total": 1,
"page": 1
}
```

CopyCopied!

[POST](#)/apps/annotations

创建标注

Query

	<div><div></div><div></div></div>
Name	<div><div></div><div></div></div>
question	<div><div></div><div></div></div>
Type	<div><div></div><div></div></div>
string	<div><div></div><div></div></div>
Description	<div><div></div><div></div></div>
问题	<div><div></div><div></div></div>
	<div><div></div><div></div></div>
Name	<div><div></div><div></div></div>
answer	<div><div></div><div></div></div>
Type	<div><div></div><div></div></div>
string	<div><div></div><div></div></div>
Description	<div><div></div><div></div></div>

答案内容

Request

POST

/apps/annotations

curl --location --request POST 'https://api.dify.ai/v1/apps/annotations' \

--header 'Authorization: Bearer {api_key}' \

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

--data-raw '{"question": "What is your name?","answer": "I am Dify."}'

CopyCopied!

Response

{

"id": "69d48372-ad81-4c75-9c46-2ce197b4d402",

"question": "What is your name?",

"answer": "I am Dify.",

"hit_count": 0,

"created_at": 1735625869

}

CopyCopied!

PUT/apps/annotations/{annotation_id}

更新标注

Query

Name	
annotation_id	
Type	
string	
Description	

标注 ID

•

•

•

Name

•

question

•

Type

•

string

•

Description

•

•

问题

•

•

•

Name

•

answer

•

Type

•

string

•

Description

•

•

答案内容

•

•

Request

PUT

/apps/annotations/{annotation_id}

curl --location --request PUT

'https://api.dify.ai/v1/apps/annotations/{annotation_id}' \

--header 'Authorization: Bearer {api_key}' \

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

--data-raw '{"question": "What is your name?","answer": "I am

Dify."}'

CopyCopied!

Response

```
{
  "id": "69d48372-ad81-4c75-9c46-2ce197b4d402",
  "question": "What is your name?",
  "answer": "I am Dify.",
  "hit_count": 0,
  "created_at": 1735625869
}
```

CopyCopied!

DELETE/apps/annotations/{annotation_id}

删除标注

Query

	•
Name	•
annotation_id	•
Type	•
string	•
Description	•
标注 ID	•
	•

Request

```
PUT
/apps/annotations/{annotation_id}

curl --location --request DELETE 'https://api.dify.ai/v1/apps/annotations/{annotation_id}' \
--header 'Authorization: Bearer {api_key}' \
--header 'Content-Type: application/json'
```

Response

204 No Content

CopyCopied!

POST/apps/annotation-reply/{action}

标注回复初始设置

Query

Name	<div><div></div><div></div></div>
action	<div><div></div><div></div></div>
Type	<div><div></div><div></div></div>
string	<div><div></div><div></div></div>
Description	<div><div></div><div></div></div>
动作，只能是 'enable' 或 'disable'	<div><div></div><div></div></div>
Name	<div><div></div><div></div></div>
embedding_provider_name	<div><div></div><div></div></div>
Type	<div><div></div><div></div></div>
string	<div><div></div><div></div></div>
Description	<div><div></div><div></div></div>
指定的嵌入模型提供商，必须先 within 系统内设定好接入的模型，对应的是 provider 字段	<div><div></div><div></div></div>
Name	<div><div></div><div></div></div>

embedding_model_name	<div><div></div></div>
Type	<div><div></div></div>
string	<div><div></div></div>
Description	<div><div></div></div>
指定的嵌入模型，对应的是 model 字段	<div><div></div></div>
Name	<div><div></div></div>
score_threshold	<div><div></div></div>
Type	<div><div></div></div>
number	<div><div></div></div>
Description	<div><div></div></div>
相似度阈值，当相似度大于该阈值时，系统会自动回复，否则不回复	<div><div></div></div>

嵌入模型的提供商和模型名称可以通过以下接口获取：v1/workspaces/current/models/model-types/text-embedding，具体见：通过 API 维护知识库。使用的 Authorization 是 Dataset 的 API Token。

Request

```
POST
/apps/annotation-reply/{action}

curl --location --request POST 'https://api.dify.ai/v1/apps/annotation-reply/{action}' \
--header 'Authorization: Bearer {api_key}' \
--header 'Content-Type: application/json' \
--data-raw '{"score_threshold": 0.9, "embedding_provider_name": "zhipu", "embedding_model_name": "embedding_3"}'
CopyCopied!
```

Response

```
{
  "job_id": "b15c8f68-1cf4-4877-bf21-ed7cf2011802",
```

```
"job_status": "waiting"
}
```

CopyCopied!

该接口是异步执行，所以会返回一个 job_id，通过查询 job 状态接口可以获取到最终的执行结果。

GET/apps/annotation-reply/{action}/status/{job_id}

查询标注回复初始设置任务状态

Query

Name	<div><div></div><div></div></div>
action	<div><div></div><div></div></div>
Type	<div><div></div><div></div></div>
string	<div><div></div><div></div></div>
Description	<div><div></div><div></div></div>
动作，只能是 'enable' 或 'disable'，并且必须和标注回复初始设置接口的动作一致	
Name	<div><div></div><div></div></div>
job_id	<div><div></div><div></div></div>
Type	<div><div></div><div></div></div>
string	<div><div></div><div></div></div>
Description	<div><div></div><div></div></div>
任务 ID，从标注回复初始设置接口返回的 job_id	

Request

GET

/apps/annotations

```
curl --location --request GET
```

```
'https://api.dify.ai/v1/apps/annotation-reply/{action}/status/{job_id}' \
```

```
--header 'Authorization: Bearer {api_key}'
```

CopyCopied!

Response

```
{
```

```
  "job_id": "b15c8f68-1cf4-4877-bf21-ed7cf2011802",
```

```
  "job_status": "waiting",
```

```
  "error_msg": ""
```

```
}
```