

云译网 接口文档

目录

1. 用户注册接口	3
2. 用户登录接口	4
3. 用户注销接口	5
4. 密码重置接口	6
5. 用户头像上传接口	7
6. 获取用户信息接口	8
7. 用户修改信息接口	9
8. OCR 识别接口	11
9. OCR 识别结果收藏接口	12
10. AI 对话接口	13
11. AI 对话历史记录保存接口	14
12. 查看对话历史记录接口	16
13. 上传个性化设置接口	17
14. 用户反馈上传接口	18
15. 在线客服聊天接口	19
16. 即时翻译（首页）	20
17. 个性化翻译	21
18. 润色翻译	22
19. 图片翻译	23
20. 文档翻译	24
21. 翻译校对	25
22. 上传翻译记录	26
23. 查询翻译记录	28
24. 获取历史翻译记录（首页）	29
25. 获取本周翻译详情（首页）	31

1. 用户注册接口

接口名称: 用户注册接口
接口描述: 用于新用户注册
请求方式: POST
URL: /users/register
请求参数:

参数名	类型	是否必填	说明
username	String	是	用户名
password	String	是	密码
password2	String	是	确认密码
sms_code	String	是	短信验证码
mobile	String	是	手机号
allow	String	是	是否同意用户协议

请求示例

```
{
  "username": "exampleUser",
  "password": "password123",
  "password2": "password123",
  "sms_code": "123456",
  "mobile": "12345678901",
  "allow": "true"}
```

响应参数

参数名	类型	是否必须	描述
status	int	是	状态码（200 表示成功）
message	String	是	注册成功或失败的信息
User_id	int	否	用户 ID（注册成功返回）

参数名	类型	是否必须	描述
username	String	否	用户名（注册成功返回）
mobile	String	否	手机号（注册成功返回）

响应示例

```
{
  "status": 200,
  "message": "注册成功",
  "id": 123,
  "username": "exampleUser",
  "mobile": "12345678901"}
```

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **500 Internal Server Error:** 服务器内部错误。

日志记录

- 记录请求参数、响应结果以及错误信息。

2. 用户登录接口

接口名称: 用户登录接口

接口描述: 用于用户登录

请求方式: POST

URL: /users/login

请求参数

参数名	类型	是否必填	说明
username	String	是	用户名
password	String	是	密码

请求示例

```
{
  "username": "exampleUser",
  "password": "password123"}
```

响应参数

参数名	类型	是否必须	描述
status	int	是	状态码（200 表示成功）
message	String	是	登录成功或失败的信息
token	String	否	登录成功后的会话令牌

响应示例

```
{
  "status": 200,
  "message": "登录成功",
  "token": "abcdefg123456"}
```

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **401 Unauthorized:** 用户名或密码错误。
- **500 Internal Server Error:** 服务器内部错误。

日志记录

- 记录请求参数、响应结果以及错误信息。

3. 用户注销接口

接口名称: 用户注销接口

接口描述: 用于用户退出登录

请求方式: POST

URL: /users/logout

请求参数

- 无

响应参数

参数名	类型	是否必须	描述
status	int	是	状态码（200 表示成功）
message	String	是	注销成功信息

响应示例

```
{
  "status": 200,
  "message": "注销成功"}
```

错误处理

- **500 Internal Server Error:** 服务器内部错误。

日志记录

- 记录请求参数、响应结果以及错误信息。

4. 密码重置接口

接口名称: 密码重置接口

接口描述: 用于用户忘记密码时的密码重置操作

请求方式: POST

URL: /users/reset-password

请求参数

参数名	类型	是否必填	说明
mobile	String	是	手机号
new_password	String	是	新密码
sms_code	String	是	短信验证码
user_id	int	是	用户的唯一标识

请求示例

```
{
  "userID": "12345",
  "mobile": "12345678901",
  "new_password": "newPassword123",
  "sms_code": "123456"}
```

返回参数

参数名	类型	是否必须	描述
-----	----	------	----

参数名	类型	是否必须	描述
status	int	是	状态码（200 表示成功）
message	String	是	密码重置结果信息

响应示例

{ "status": 200, "message": "密码重置成功"}

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **500 Internal Server Error:** 服务器内部错误。

日志记录

- 记录请求参数、响应结果以及错误信息。

5. 用户头像上传接口

接口名称: 用户头像上传接口

接口描述: 用于上传用户头像

请求方式: POST

URL: /users/avatar/upload

请求参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识
avatar	file	是	用户头像文件

请求示例

使用文件上传工具（如 `form-data` 方式）上传文件。

返回参数

参数名	类型	是否必须	描述
status	int	是	状态码（200 表示成功）

参数名	类型	是否必须	描述
message	String	是	头像上传结果信息
avatar_url	String	否	上传成功后的头像 URL（可选）

响应示例

```
{
  "status": 200,
  "message": "头像上传成功",
  "avatar_url": "https://example.com/avatar/123.jpg"}
```

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **500 Internal Server Error:** 服务器内部错误。

日志记录

- 记录请求参数、响应结果以及错误信息。

6. 获取用户信息接口

接口名称: 获取用户信息接口

接口描述: 用于获取当前已登录用户的信息

请求方式: GET

URL: /users/info

请求参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识
fields	String	否	需要返回的字段列表，用逗号分隔。如果未指定，则返回全部字段

参数示例:

- 请求全部字段: /users/info
- 请求部分字段: /users/info?fields=username,mobile,register_date

返回参数

参数名	类型	是否必须	描述
username	String	否	用户名
mobile	String	否	手机号
register_date	String	否	注册日期
login_history	Array	否	登录历史
location	String	否	籍贯
avatar_url	String	否	用户头像 URL
email	String	否	绑定邮箱

响应示例

```
{
  "username": "exampleUser",
  "mobile": "12345678901",
  "register_date": "2024-01-01",
  "login_history": ["2024-01-01 10:00", "2024-01-02 12:00"],
  "location": "China",
  "avatar_url": "https://example.com/avatar/123.jpg",
  "email": "user@example.com"}
```

错误处理

- **500 Internal Server Error:** 服务器内部错误。

日志记录

- 记录请求参数、响应结果以及错误信息。

7. 用户修改信息接口

接口名称: 用户修改信息接口

接口描述: 用于修改当前用户的部分信息

请求方式: POST

URL: /users/update

请求参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识
username	String	否	新用户名
mobile	String	否	新手机号
location	String	否	籍贯
email	String	否	绑定邮箱

请求示例

```
{
  "username": "newExampleUser",
  "mobile": "09876543210",
  "location": "USA",
  "email": "newuser@example.com"}
```

返回参数

参数名	类型	是否必须	描述
status	int	是	状态码（200 表示成功）
message	String	是	信息更新结果消息

响应示例

```
{
  "status": 200,
  "message": "信息更新成功"}
```

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **500 Internal Server Error:** 服务器内部错误。

日志记录

- 记录请求参数、响应结果以及错误信息。

8. OCR 识别接口

接口名称: OCR 识别接口

接口描述: 用于进行 OCR 识别操作，将图片或文档中的文本内容识别并返回，可将结果显示在翻译页面的输入框中。

请求方式: POST

URL: /ocr/recognize（由服务转发到实际 OCR API）

请求参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识
file	file	是	待识别的图片、PDF 或文档文件
file_type	String	否	文件类型，默认值为 image，支持 image、pdf、doc 等

请求示例

使用文件上传工具（如 `form-data` 方式）上传文件，并指定文件类型（可选）。

返回参数

参数名	类型	是否必须	描述
status	int	是	状态码（200 表示成功）
message	String	是	识别结果的提示信息
text_content	String	是	OCR 识别出的文本内容
record_id	int	否	识别结果的历史记录 ID（可选）

响应示例

```
{
  "status": 200,
  "message": "识别成功",
  "text_content": "这是 OCR 识别出的文本内容。",
  "record_id": 12345}
```

错误处理

- 400 Bad Request:** 请求参数不完整或格式错误。
- 415 Unsupported Media Type:** 不支持的文件格式。
- 500 Internal Server Error:** 服务器内部错误，如 OCR 服务不可用。

日志记录

- 记录请求参数、响应结果以及错误信息。

9. OCR 识别结果收藏接口

接口名称: OCR 识别结果收藏接口

接口描述: 用于将 OCR 识别结果与原文件收藏保存成历史记录，便于用户查看历史识别内容。

请求方式: POST

URL: /ocr/recognize/record

请求参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识
text_content	String	是	识别出的文本内容
file	file	是	原始图片、PDF 或文档
timestamp	String	否	收藏时间（系统自动生成，默认当前时间）

请求示例

<pre>{ "text_content": "识别出的文本内容", "file": "example.pdf", "timestamp": "2024-11-07T10:00:00Z"}</pre>
--

返回参数

参数名	类型	是否必须	描述
status	int	是	状态码（200 表示成功）
message	String	是	收藏结果的提示信息
record_id	int	否	成功收藏的记录 ID（可选）

响应示例

```
{
  "status": 200,
  "message": "收藏成功",
  "record_id": 67890}
```

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **409 Conflict:** 重复收藏的请求。
- **500 Internal Server Error:** 服务器内部错误，如数据库连接问题。

日志记录

- 记录请求参数、响应结果以及错误信息。

10. AI 对话接口

接口名称: AI 对话接口

接口描述: 用于进行 AI 对话，用户输入内容后，通过 Google 的 Gimme 大模型 API 返回相应的对话内容。

请求方式: POST

URL: Gimme API

请求参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识
input_text	str	是	用户输入的对话内容
session_id	str	否	会话 ID，若需要维持同一会话历史
timestamp	str	否	对话时间（系统自动生成，默认当前时间）

请求示例

```
{
  "input_text": "你好，AI",
  "session_id": "12345",
  "timestamp": "2024-11-07T10:00:00Z"}
```

返回参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识
input_text	str	是	用户输入的对话内容
session_id	str	否	会话 ID，若需要维持同一会话历史
timestamp	str	否	对话时间（系统自动生成，默认当前时间）

响应示例

```
{
  "status": 200,
  "message": "你好，我是 AI 助手。",
  "response_id": 12345,
  "session_id": "67890"}
```

错误处理

- 400 Bad Request:** 请求参数不完整或格式错误。
- 409 Conflict:** 重复收藏的请求。
- 500 Internal Server Error:** 服务器内部错误，如数据库连接问题。

日志记录

记录请求参数、响应结果以及错误信息。

11. AI 对话历史记录保存接口

接口名称: AI 对话历史记录保存接口

接口描述: 将用户与 AI 的对话内容保存到历史记录中。

请求方式: POST

URL: /ai/dialogue/record

请求参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识

参数名	类型	是否必填	说明
session_id	str	是	会话 ID
input_text	str	是	用户输入的内容
ai_response	str	是	AI 回复的内容
timestamp	str	否	对话时间（系统自动生成，默认当前时间）

请求示例

```
{
  "session_id": "12345",
  "input_text": "你好，AI",
  "ai_response": "你好，我是 AI 助手。",
  "timestamp": "2024-11-07T10:00:00Z"}
```

返回参数

参数名	类型	是否必填	说明
status	int	是	状态码（200 表示成功）
message	str	是	保存成功信息或错误信息
record_id	int	否	保存成功的记录 ID

响应示例

```
{
  "status": 200,
  "message": "保存成功",
  "record_id": 67890}
```

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **409 Conflict:** 重复收藏的请求。
- **500 Internal Server Error:** 服务器内部错误，如数据库连接问题。

日志记录

记录请求参数、响应结果以及错误信息。

12. 查看对话历史记录接口

接口名称: 查看对话历史记录接口
接口描述: 用于查看用户与 AI 的历史对话内容。
请求方式: GET
URL: /ai/dialogue/history
请求参数

参数名	类型	是否必填	说明
user_id	int	是	用户的唯一标识
session_id	str	否	会话 ID（不传则获取所有历史记录）
limit	int	否	每次获取的历史记录条数（默认 10 条）
offset	int	否	分页偏移量，用于分页获取历史记录

返回参数

参数名	类型	是否必填	说明
status	int	是	状态码（200 表示成功）
history	list	是	历史对话记录列表
session_id	str	否	会话 ID

4. 响应示例

```
{
  "status": 200,
  "history": [
    {
      "input_text": "你好，AI",
      "ai_response": "你好，我是 AI 助手。",
      "timestamp": "2024-11-07T10:00:00Z"
    }
  ],
  "session_id": "12345"}
```

5. 错误处理

- 400 Bad Request:** 请求参数不完整或格式错误。
- 409 Conflict:** 重复收藏的请求。
- 500 Internal Server Error:** 服务器内部错误，如数据库连接问题。

6. 日志记录

记录请求参数、响应结果以及错误信息。

13. 上传个性化设置接口

接口名称: 上传个性化设置接口
接口描述: 用于上传用户在个性化定制页面选择的个性化设置，将设置内容保存到数据库中。
请求方式: POST
URL: /settings/upload
请求参数

参数名	类型	是否必须	说明
user_id	int	是	用户的唯一标识
style	str	是	文本风格选项
field	str	是	专业领域选项
public_term	str	否	公共术语库选择项
custom_term	str	否	自定义术语库
memory_import	str	否	导入的翻译记忆库
memory_edit	str	否	编辑的翻译记忆库
target_language	str	是	目标语言，可能的值：英文普通、美式英语、英式英语、日语、俄语
additional_notes	str	否	额外备注，用户输入的备注内容

请求示例

```
{
  "style": "casual",
  "field": "technology",
  "public_term": "general",
  "custom_term": "my_terms",
  "memory_import": "imported_memory",
  "memory_edit": "edited_memory",
  "target_language": "American English",
  "additional_notes": "Please consider these settings for my future visits."}
```

返回参数

参数名	类型	是否必须	说明
status	int	是	状态码（200 表示成功）
message	str	是	操作结果的提示信息

响应示例

```
{
  "status": 200,
  "message": "个性化设置已成功保存"}
```

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **500 Internal Server Error:** 服务器内部错误，如数据库连接问题。

日志记录

记录请求参数、响应结果以及错误信息。

14. 用户反馈上传接口

接口名称: 用户反馈上传接口

接口描述: 用于用户上传反馈信息，反馈内容包括页面问题、功能建议等。

请求方式: POST

URL: /feedback/upload

请求参数

参数名	类型	是否必须	说明
user_id	int	是	用户的唯一标识
feedback_type	str	是	反馈类型，可能的值有：页面问题、功能建议、其他
feedback_content	str	是	反馈内容，用户填写的详细描述

请求示例

```
{
  "feedback_type": "page_issue",
  "feedback_content": "The login page is not loading properly on my browser."}
```

返回参数

参数名	类型	是否必须	说明
status	int	是	状态码（200 表示成功）
message	str	是	提示信息，操作结果描述

响应示例

```
{
  "status": 200,
  "message": "感谢您的反馈，我们会尽快处理。"}
```

错误处理

- **400 Bad Request:** 请求参数不完整或格式错误。
- **500 Internal Server Error:** 服务器内部错误，如数据库连接问题。

日志记录

记录请求参数、响应结果以及错误信息。

15. 在线客服聊天接口

接口名称: 在线客服聊天接口
接口描述: 用于用户与在线客服进行实时聊天。
请求方式: POST
URL: /customer/chat
请求参数

参数名	类型	是否必须	说明
user_id	int	是	用户 ID，用于标识聊天用户
message	str	是	用户发送的消息
timestamp	str	是	消息发送的时间戳，用于记录消息顺序

请求示例

```
{
  "status": 200,
  "reply": "Hello, how can I assist you today?",
  "timestamp": "2024-11-07T10:00:00Z"}
```

返回参数

参数名	类型	是否必须	说明
status	int	是	状态码（200 表示成功）
reply	str	是	客服回复的消息
timestamp	str	是	消息发送或接收的时间戳

响应示例

```
{
  "user_id": 12345,
  "message": "I have a question about the product."}
```

错误处理

- 400 Bad Request: 请求参数不完整或格式错误。
- 500 Internal Server Error: 服务器内部错误，如数据库连接问题。

日志记录

记录请求参数、响应结果以及错误信息。

16. 即时翻译（首页）

接口名称: translate/instant

请求方法: POST

URL: /api/translate/instant

请求参数:

参数名	类型	是否必填	说明
userID	int	是	用户 ID
text	String	是	待翻译的文本
source	String	否	源语言，默认为自动检测
target	String	是	目标语言

请求示例:

```
{
  "userID": "12345",
  "text": "Hello, how are you?",
  "source": "en",
  "target": "zh"}
```

响应参数:

参数名	类型	描述
translation	String	翻译结果
synonyms	Array	同义词列表
examples	Array	例句列表

响应示例:

```
{
  "translation": "你好，你好吗？",
  "synonyms": ["你好", "您好"],
  "examples": ["Hello, how are you? 你好，你好吗？"]}
```

错误处理:

400 Bad Request: 请求参数不完整或格式错误。
500 Internal Server Error: 服务器内部错误，如无法连接校对服务。

日志记录:

记录请求参数、响应结果以及错误信息。

17. 个性化翻译

接口名称: translate/personalized

请求方法: POST

URL: /api/translate/personalized

请求参数:

参数名	类型	必填	描述
userID	int	是	用户 ID
text	String	是	需要翻译的文本
source	String	否	源语言，默认为自动检测
target	String	是	目标语言

请求示例:

```
{
  "userID": "12345",
  "text": "Hello, how are you?",
  "source": "en",
  "target": "zh",
}
```

响应参数:

参数名	类型	描述
translation	String	翻译结果

响应示例:

```
{
  "translation": "您好，请问您最近如何？"
}
```

错误处理:

400 Bad Request: 请求参数不完整或格式错误。
500 Internal Server Error: 服务器内部错误，如无法连接校对服务。

日志记录:

记录请求参数、响应结果以及错误信息

18. 润色翻译

接口名称: translate/polish

请求方法: POST

URL: /api/translate/polish

请求参数:

参数名	类型	必填	描述
userID	int	是	用户 ID
text	String	是	需要翻译的文本
source	String	否	源语言，默认为自动检测
target	String	是	目标语言
polish_style	String	否	润色风格

请求示例:

```
{
  "text": "Hello, how are you?",
  "source": "en",
  "target": "zh"
  "polish_style": "商务"}
```

响应参数:

参数名	类型	描述
translation	String	翻译结果

响应示例:

```
{
  "translation": "您好，请问您最近如何？"}
```

错误处理:

400 Bad Request: 请求参数不完整或格式错误。
500 Internal Server Error: 服务器内部错误，如无法连接校对服务。

日志记录:

记录请求参数、响应结果以及错误信息

19. 图片翻译

接口名称: translate/image

请求方法: POST

URL: /api/translate/image

请求参数:

参数名	类型	必填	描述
userID	int	是	用户 ID
image	File	是	需要翻译的图片文件
source	String	否	源语言，默认为自动检测
target	String	是	目标语言

请求示例:

```
{
  "image": "image.jpg",
  "source": "en",
  "target": "zh"}
```

响应参数:

参数名	类型	描述
translation	String	翻译结果

响应示例:

```
{
  "translation": "你好，世界"}
```

错误处理:

400 Bad Request: 请求参数不完整或格式错误。

500 Internal Server Error: 服务器内部错误，如无法连接校对服务。

日志记录:

记录请求参数、响应结果以及错误信息。

20. 文档翻译

接口名称: translate/document

请求方法: POST

URL: /api/translate/document

请求参数:

参数名	类型	必填	描述
userID	int	是	用户 ID
document	File	是	需要翻译的文档文件
source	String	否	源语言，默认为自动检测
target	String	是	目标语言

请求示例:

```
{
  "document": "document.pdf",
  "source": "en",
  "target": "zh"}
```

响应参数:

参数名	类型	描述
translation	String	翻译结果

响应示例:

```
{
  "translation": "这是一篇文档的翻译内容"}
```


错误处理:

400 Bad Request: 请求参数不完整或格式错误。

500 Internal Server Error: 服务器内部错误，如无法连接校对服务。

日志记录:

记录请求参数、响应结果以及错误信息。

21. 翻译校对

接口名称: translate/proofread

请求方法: POST

URL: /api/translate/proofread

请求参数:

参数名	类型	必填	描述
userID	int	是	用户 ID
originalText	String	是	原文内容
translatedText	String	是	用户翻译的内容
source	String	否	源语言，默认为自动检测
target	String	是	目标语言

请求示例:

<pre>{ "originalText": "Hello, how are you?", "translatedText": "你好，你怎么样？", "source": "en", "target": "zh"}</pre>

响应参数:

参数名	类型	描述
grammarCheck	Array	语法检查结果
contentConsistency	Array	内容一致性检查结果
terminologyValidation	Array	术语校验结果
languageStyle	Array	语言风格一致性检查结果
sentenceStructure	Array	句式优化建议

响应示例:

```
{
  "grammarCheck": [无误],
  "contentConsistency": [基本一致],
  "terminologyValidation": [XXX 术语翻译错误],
  "languageStyle": ["建议使用更自然的表达方式"],
  "sentenceStructure": ["建议将句子拆分为两个独立句子"]}
```

错误处理:
400 Bad Request: 请求参数不完整或格式错误。
500 Internal Server Error: 服务器内部错误，如无法连接校对服务。

日志记录:
记录请求参数、响应结果以及错误信息。

22. 上传翻译记录

接口名称：翻译记录存储
接口描述：该接口用于存储和查询不同类型的翻译记录，包括即时翻译、个性化翻译、润色翻译、图片翻译、文档翻译和翻译校对。
接口路径：POST /api/v1/translation/store
请求方法：POST
请求参数：

参数名	类型	必填	描述	示例值
user_id	string	是	用户的唯一标识符	12345
translation_type	string	是	翻 译 类 型 （ instant, personalized, polish, image, document, proofread ）	instant
original_text	string	否	原文	The quick brown fox jumps over the lazy dog.
original_image	file	否	翻译图片	
original_doc	file	否	翻译文件	
translated_text	string	是	翻译后的文本	快速的棕色狐狸跳过懒狗。
source	String	是	源语言，默认为自动检测	英语（en）
target	String	是	目标语言	中文(zh)
additional_data	object	否	额外数据，包含同义词、例句等	{“synonyms”:[“swift”], “examples”:[“The quick

				brown fox···”]}
polish_style	string	否	润色风格，如“Elegant”、“Concise”等	Elegant
proofread_data	object	否	校对数据，包含语法检查、内容一致性等	{ “ grammar_check ” : “ OK ” , “ content_consistency ” : “ Good ” }

请求示例：

```
POST /api/v1/translation/store
Content-Type: application/json

{
  "user_id": "12345",
  "translation_type": "instant",
  "source": "en",
  "target": "zh",
  "original_text": "The quick brown fox jumps over the lazy dog.",
  "translated_text": "快速的棕色狐狸跳过懒狗。",
  "additional_data": {
    "synonyms": ["swift"],
    "examples": ["The quick brown fox..."]
  }
}
```

23. 查询翻译记录

接口名称：翻译记录查询接口

接口描述：该接口用于查询不同类型的翻译记录，包括即时翻译、个性化翻译、润色翻译、图片翻译、文档翻译和翻译校对。

接口路径：查询翻译记录: GET /api/v1/translation/query

请求方法：查询翻译记录: GET

请求参数：

参数名	类型	是否必填	描述	示例值
user_id	string	是	用户的唯一标识符	12345
translation_type	string	否	翻译类型（instant, personalized, polish, image, document, proofread）	instant
page	int	否	页码，默认值为 1	1
page_size	int	否	每页记录数，默认值为 10	10

请求示例：

GET /api/v1/translation/query?user_id=12345&translation_type=instant&page=1&page_size=10

返回参数：

参数名	类型	描述
user_id	Str	用户的唯一标识符
total_records	int	总的记录数
page	int	当前页码
page_size	int	每页的记录数
translations	array	翻译记录列表（数组，每个元素是一个对象）
translation_id	str	翻译记录的唯一标识符
translation_type	str	翻译类型
source	String	源语言，默认为自动检测
target	String	目标语言
original_text	str	原文段
translated_text	str	翻译后的文本
additional_data	object	额外数据，包含同义词、例句等
polish_style	str	润色风格
proofread_data	object	校对数据，包含语法检查、内容一致性等

返回示例：

```
{
  "user_id": "12345",
  "total_records": 100,
  "page": 1,
  "page_size": 10,
  "translations": [
    {
      "translation_id": "1",
      "translation_type": "instant",
      "original_text": "The quick brown fox jumps over the lazy dog.",
      "translated_text": "快速的棕色狐狸跳过懒狗。",
      "additional_data": {
        "synonyms": ["swift"],
        "examples": ["The quick brown fox..."]
      }
    },
    {
      "translation_id": "2",
      "translation_type": "proofread",
      "original_text": "The quick brown fox jumps over the lazy dog.",
      "translated_text": "快速的棕色狐狸跳过懒狗。",
      "proofread_data": {
        "grammar_check": "OK",
        "content_consistency": "Good",
        "terminology_check": "Good",
        "language_style_consistency": "Good",
        "sentence_structure_optimization": "Good"
      }
    }
  ]
}
```

24. 获取历史翻译记录（首页）

接口名称：查询本周数据接口

接口描述：该接口用于查询本周数据的概览，包括翻译次数、查询例句次数、AI 对话次数和查询同义词次数的统计信息，以及翻译次数的排行榜前 3 名。

接口路径：GET /api/v1/weekly-data

请求方法：GET

请求参数

参数名	类型	是否必填	描述	示例值
userID	int	是	用户 ID	userID
start_date	string	否	查询的起始日期 (格 式 : YYYY-MM-DD)	2024-11-04
end_date	string	否	查询的结束日期 (格 式 : YYYY-MM-DD)	2024-11-10

示例请求

GET /api/v1/weekly-data?start_date=2024-11-04&end_date=2024-11-10

响应参数:

参数名	类型	描述
total_translations	int	本周内所有翻译操作的总次数
translation_ranking	Array	本周翻译次数最多的前 3 个语言对的排行榜
language_pair	string	翻译的语言对, 例如 “English to Chinese”
count	int	该语言对的翻译次数
total_sentence_queries	int	本周内查询例句的总次数
total_ai_conversations	int	本周内 AI 对话的总次数
total_synonym_queries	int	本周内查询同义词的总次数

响应示例

<pre>{ "weekly_summary": { "total_translations": 12345, "translation_ranking": [{ "language_pair": "English to Chinese", "count": 3456 }, { "language_pair": "Spanish to Chinese", "count": 2345 }] } }</pre>

```
    {
      "language_pair": "French to Chinese",
      "count": 1234
    }
  ],

  "total_sentence_queries": 5678,
  "total_ai_conversations": 4567,
  "total_synonym_queries": 3456
}}
```

错误码

错误码	描述
400	无效的请求参数
500	服务器内部错误

25. 获取本周翻译详情（首页）

接口介绍：查询用户本周翻译记录接口
接口描述：该接口用于查询用户本周的翻译记录，包括翻译的语言、前三个最常翻译的语种，以及用户最常翻译的词汇。
接口路径：GET /api/v1/user/weekly-translation-history
请求方法：GET

请求参数

参数名	类型	是否必填	描述	示例值
user_id	string	是	用户的唯一标识符	12345
start_date	string	否	查询的起始日期（格式：YYYY-MM-DD）	2024-11-04
end_date	string	否	查询的结束日期（格式：YYYY-MM-DD）	2024-11-10

请求示例：
GET
/api/v1/user/weekly-translation-history?user_id=12345&start_date=2024-11-04&end_date=2024-11-10

响应参数

参数名	类型	描述
weekly_translation_history	array	本周的翻译记录（对象）。
languages	Array、str	本周内用户翻译的语言对列表
top_3_languages	Array、str	本周内用户最常翻译的前三个语种
top_words	Array、str	本周内用户最常翻译的词汇

响应示例

```
{
  "user_id": "12345",
  "weekly_translation_history": {
    "languages": [
      "English to Chinese",
      "Spanish to Chinese",
      "French to Chinese"
    ],
    "top_3_languages": [
      "English",
      "Spanish",
      "French"
    ],
    "top_words": [
      "example",
      "word",
      "translation"
    ]
  }
}
```

错误码

错误码	描述
400	无效的请求参数
500	服务器内部错误
404	用户 ID 不存在