pygtrans

发布 *1.2.0*

foyou

2022 年 04 月 19 日

Contents

1 pygtrans 包 **1**

1.1 pygtrans.Translate 模块 1

1.2 pygtrans.ApiKeyTranslate 模块 3

1.3 pygtrans.Null 模块 6

1.4 pygtrans.LanguageResponse 模块 6

1.5 pygtrans.DetectResponse 模块 7

1.6 pygtrans.TranslateResponse 模块 7

2 支持语言 **9**

2.1 源语言 9

2.2 目标语言 13

3 索引和搜索 **17**

4 项目地址 **19**

5 依赖 **21**

6 致谢 **23**

Python 模块索引 **25**

索引 **27**

CHAPTER 1

pygtrans 包

1.1 pygtrans.Translate 模块

通过调用 `google.com | google.cn | ...` web 接口进行翻译如果你有 **google cloud translate apikey**, 请使用*pygtrans.ApiKeyTranslate*

基本功能:

1. 语言检测, 支持批量检测
2. 文本翻译, 支持批量, 支持 html 模式翻译

class pygtrans.Translate.**Translate**(*target: str = 'zh-CN', source: str = 'auto', fmt='html', user_agent: Optional[str] = None, domain: str = 'com', proxies: Optional[Dict] = None*)

基类: `object`

参数

- **target** –str: (可选) 目标语言, 默认: `zh-CN`, 查看完整列表
- **source** –str: (可选) 源语言, 默认: `auto` (自动检测), 查看完整列表
- **fmt** –str: (可选) 文本格式, `text | html`, 默认: `html`
- **user_agent** –str: (可选) 用户代理, 这个参数很重要, 不设置或错误设置非常容易触发 **429 Too Many Requests** 错误, 默认: `GoogleTranslate/6.18.0.06.376053713 (Linux; U; Android 11; GM1900)`, 所以用户可以不用提供. 这

个默认 User-Agent 很稳定, 暂时未发现 429 错误, 如果出现 429, 建议 **模仿默认进行构造**, 或者进行 反馈

- **domain** –str: (可选) 域名 google.com 及其可用平行域名 (如: google.cn), 默认: google.com
- **proxies** –(可选) eg: proxies = { 'http' : 'http://localhost:10809' , 'https' : 'http://localhost:10809' }

基本用法:

```
>>> from pygtrans import Translate
>>> client = Translate(proxies={'http': 'http://localhost:10809', 'https':
↪'http://localhost:10809'})
>>> client.detect('谷歌翻译').language
'zh-CN'
>>> text = client.translate('Hello, Google')
>>> text.translatedText
'你好, 谷歌'
>>> texts = client.translate(['批量测试', '批量翻译'], target='en')
>>> for text in texts:
...     print(text.translatedText)
Batch test
Batch translation
```

detect (*q: str*) → Union[*pygtrans.DetectResponse.DetectResponse*, *pygtrans.Null.Null*]

语言检测

参数 **q** –需要检测的内容, 不支持批量, 如需批量, 请参阅: translate_and_detect().

返回 成功则返回*pygtrans.DetectResponse.DetectResponse* 对象, 失败则返回*pygtrans.Null.Null* 对象

基本用法:

```
>>> from pygtrans import Translate
>>> client = Translate(proxies={'http': 'http://localhost:10809', 'https
↪': 'http://localhost:10809'})
>>> d = client.detect('こんにちは')
>>> assert d.language == 'ja'
```

translate (*q: str*, *target: str = None*, *source: str = None*, *fmt: str = None*) → *pygtrans.TranslateResponse.TranslateResponse*

translate (*q: List[str]*, *target: str = None*, *source: str = None*, *fmt: str = None*) → List[*pygtrans.TranslateResponse.TranslateResponse*]

翻译文本, 支持批量, 支持 html

参数

- **q** –str: 字符串或字符串列表
- **target** –str: (可选) 目标语言, 默认: `self.target`, 查看支持列表
- **source** –str: (可选) 源语言, 默认: `self.source`, 查看支持列表
- **fmt** –str: (可选) 文本格式, `text|html`, 默认: `self.format`

返回 成功则返回: *pygtrans.TranslateResponse.TranslateResponse* 对象, 或*pygtrans.TranslateResponse.TranslateResponse* 对象列表, 这取决于参数: *q* 是字符串还是字符串列表. 失败则返回*pygtrans.Null.Null* 对象

基本用法:

```
>>> from pygtrans import Translate
>>> client = Translate(proxies={'http': 'http://localhost:10809', 'https
↪': 'http://localhost:10809'})
>>> text = client.translate('Hello, Google')
>>> text.translatedText
'你好，谷歌'
>>> texts = client.translate(['批量测试', '批量翻译'], target='en')
>>> for text in texts:
...     print(text.translatedText)
Batch test
Batch translation
```

tts(*q: str*, *target: Optional[str] = None*) → Union[bytes, *pygtrans.Null.Null*]

语音: 实验性功能

参数

- **q** –只支持短语字符串
- **target** –目标语言

返回 返回二进制数据, 需要自行写入文件, MP3

1.2 pygtrans.ApiKeyTranslate 模块

通过 Google Cloud Translate APIKEY 进行翻译如果你没有 APIKEY, 请使用*pygtrans.Translate.Translate*

基本功能:

1. 获取语言列表
2. 语言检测, 支持批量检测

3. 文本翻译, 支持批量, 支持 html 模式翻译

class pygtrans.ApiKeyTranslate.**ApiKeyTranslate**(*api_key: str*, *target: str = 'zh-CN'*, *source: Optional[str] = None*, *fmt: str = 'html'*, *model: str = 'nmt'*, *proxies: Optional[Dict] = None*)

基类: object

参数

- **api_key** –str: 谷歌云翻译 APIKEY, 查看详情
- **target** –str: (可选) 目标语言, 默认: zh-CN, 参考列表
- **source** –str: (可选) 源语言, 默认: auto (自动检测), 参考列表
- **fmt** –str: (可选) 文本格式, text | html, 默认: html
- **model** –str: (可选) 翻译模型. 可以是 base 使用 Phrase-Based Machine Translation (PBMT) 模型，或者 nmt 使用 Neural Machine Translation (NMT) 模型。如果省略，则使用 nmt。如果模型是 nmt，并且 NMT 模型不支持请求的语言翻译对，则使用 PBMT 模型翻译请求。
- **proxies** –(可选) eg: *proxies = {'http': 'http://localhost:10809','https': 'http://localhost:10809'}*

基本用法:

```
>>> from pygtrans import ApiKeyTranslate
>>> client = ApiKeyTranslate(api_key='<api_key>')
>>> langs = client.languages()  # 此种方式的语言列表, 请使用此方法获取
>>> langs[0]
LanguageResponse(language='sq', name='阿尔巴尼亚语')
>>> text = client.translate('Google Translate')
>>> text.translatedText
'谷歌翻译'
>>> text.detectedSourceLanguage
'en'
>>> texts = client.translate(['🅕🅕🅕🅕🅕🅕', 'こんにちは'])
>>> texts[0].translatedText
'你好'
>>> texts[0].detectedSourceLanguage
'ko'
>>> texts[1].translatedText
'你好'
>>> texts[1].detectedSourceLanguage
'ja'
```

detect(*q: str*) → *pygtrans.DetectResponse.DetectResponse*

detect(*q: List[str]*) → List[*pygtrans.DetectResponse.DetectResponse*]

语言检测, 支持批量

参数 **q** –字符串或字符串列表

返回 成功则返回: `pygtrans.TranslateResponse.DetectResponse` 对象, 或 `pygtrans.TranslateResponse.DetectResponse` 对象列表, 这取决于 *参数: q* 是字符串还是字符串列表. 失败则返回`pygtrans.Null.Null` 对象

基本用法:

```
>>> from pygtrans import ApiKeyTranslate
>>> client = ApiKeyTranslate(api_key='<api_key>')
>>> d1 = client.detect('Hello')
>>> d1.language
'en'
>>> assert isinstance(client.detect(['Hello', 'Google']), list)
```

languages(*target: Optional[str] = None*, *model: Optional[str] = None*) → Union[List[*pygtrans.LanguageResponse.LanguageResponse*], *pygtrans.Null.Null*]

语言支持列表

translate(*q: str*, *target: str = None*, *source: str = None*, *fmt: str = None*, *model: str = None*) → *pygtrans.TranslateResponse.TranslateResponse*

translate(*q: List[str]*, *target: str = None*, *source: str = None*, *fmt: str = None*, *model: str = None*) → List[*pygtrans.TranslateResponse.TranslateResponse*]

文本翻译, 支持批量

参数

- **q** –str: 字符串或字符串列表
- **target** –str: (可选) 目标语言, 默认: `self.target`, 查看支持列表
- **source** –str: (可选) 源语言, 默认: `self.source`, 查看支持列表
- **fmt** –str: (可选) 文本格式, `text`|`html`, 默认: `self.format`
- **model** –str: (可选) 翻译模型, `nmt`|`pbmt`, 默认: `self.model`

返回 成功则返回: `pygtrans.TranslateResponse.TranslateResponse` 对象, 或`pygtrans.TranslateResponse.TranslateResponse` 对象列表, 这取决于 *参数: q* 是字符串还是字符串列表. 失败则返回`pygtrans.Null.Null` 对象

基本用法:

```
>>> from pygtrans import ApiKeyTranslate
>>> client = ApiKeyTranslate(api_key='<api_key>')
>>> text = client.translate('Google Translate')
>>> text.translatedText
'谷歌翻译'
>>> text.detectedSourceLanguage
'en'
>>> texts = client.translate(['🅕🅕🅕🅕🅕🅕', 'こんにちは'])
>>> texts[0].translatedText, texts[1].translatedText
('你好', '你好')
```

pygtrans.ApiKeyTranslate.**split_list**(*obj_list: List, sub_size: int = 128*) → List[list]

split list

参数

- **obj_list** –list object
- **sub_size** –sub list size

返回 List[list]

pygtrans.ApiKeyTranslate.**split_list_by_content_size**(*obj_list: List[str], content_size: int = 102400*) → List[List[str]]

...

1.3 pygtrans.Null 模块

表示一个失败的结果

class pygtrans.Null.**Null**(*response: requests.models.Response*)

基类：object

参数 **response** –请求失败的 requests.Response 对象

1.4 pygtrans.LanguageResponse 模块

LanguageResponse

class pygtrans.LanguageResponse.**LanguageResponse**(*language: str, name: Optional[str] = None*)

基类：object

1.5 pygtrans.DetectResponse 模块

DetectResponse

class pygtrans.DetectResponse.**DetectResponse**(*language: str*, *isReliable: bool = True*, *confidence: float = 1.0*)

基类：object

1.6 pygtrans.TranslateResponse 模块

TranslateResponse

class pygtrans.TranslateResponse.**TranslateResponse**(*translatedText: str*, *detectedSourceLanguage: Optional[str] = None*, *model: Optional[str] = None*)

基类：object

CHAPTER 2

支持语言

2.1 源语言

1. 检测语言: `auto`
2. 阿尔巴尼亚语: `sq`
3. 阿拉伯语: `ar`
4. 阿姆哈拉语: `am`
5. 阿塞拜疆语: `az`
6. 爱尔兰语: `ga`
7. 爱沙尼亚语: `et`
8. 奥利亚语: `or`
9. 巴斯克语: `eu`
10. 白俄罗斯语: `be`
11. 保加利亚语: `bg`
12. 冰岛语: `is`
13. 波兰语: `pl`
14. 波斯尼亚语: `bs`
15. 波斯语: `fa`

16. 布尔语 (南非荷兰语): `af`
17. 鞑靼语: `tt`
18. 丹麦语: `da`
19. 德语: `de`
20. 俄语: `ru`
21. 法语: `fr`
22. 菲律宾语: `tl`
23. 芬兰语: `fi`
24. 弗里西语: `fy`
25. 高棉语: `km`
26. 格鲁吉亚语: `ka`
27. 古吉拉特语: `gu`
28. 哈萨克语: `kk`
29. 海地克里奥尔语: `ht`
30. 韩语: `ko`
31. 豪萨语: `ha`
32. 荷兰语: `nl`
33. 吉尔吉斯语: `ky`
34. 加利西亚语: `gl`
35. 加泰罗尼亚语: `ca`
36. 捷克语: `cs`
37. 卡纳达语: `kn`
38. 科西嘉语: `co`
39. 克罗地亚语: `hr`
40. 库尔德语: `ku`
41. 拉丁语: `la`
42. 拉脱维亚语: `lv`
43. 老挝语: `lo`
44. 立陶宛语: `lt`
45. 卢森堡语: `lb`

46. 卢旺达语: `rw`
47. 罗马尼亚语: `ro`
48. 马尔加什语: `mg`
49. 马耳他语: `mt`
50. 马拉地语: `mr`
51. 马拉雅拉姆语: `ml`
52. 马来语: `ms`
53. 马其顿语: `mk`
54. 毛利语: `mi`
55. 蒙古语: `mn`
56. 孟加拉语: `bn`
57. 缅甸语: `my`
58. 苗语: `hmn`
59. 南非科萨语: `xh`
60. 南非祖鲁语: `zu`
61. 尼泊尔语: `ne`
62. 挪威语: `no`
63. 旁遮普语: `pa`
64. 葡萄牙语: `pt`
65. 普什图语: `ps`
66. 齐切瓦语: `ny`
67. 日语: `ja`
68. 瑞典语: `sv`
69. 萨摩亚语: `sm`
70. 塞尔维亚语: `sr`
71. 塞索托语: `st`
72. 僧伽罗语: `si`
73. 世界语: `eo`
74. 斯洛伐克语: `sk`
75. 斯洛文尼亚语: `sl`

76. 斯瓦希里语: `sw`
77. 苏格兰盖尔语: `gd`
78. 宿务语: `ceb`
79. 索马里语: `so`
80. 塔吉克语: `tg`
81. 泰卢固语: `te`
82. 泰米尔语: `ta`
83. 泰语: `th`
84. 土耳其语: `tr`
85. 土库曼语: `tk`
86. 威尔士语: `cy`
87. 维吾尔语: `ug`
88. 乌尔都语: `ur`
89. 乌克兰语: `uk`
90. 乌兹别克语: `uz`
91. 西班牙语: `es`
92. 希伯来语: `iw`
93. 希腊语: `el`
94. 夏威夷语: `haw`
95. 信德语: `sd`
96. 匈牙利语: `hu`
97. 修纳语: `sn`
98. 亚美尼亚语: `hy`
99. 伊博语: `ig`
100. 意大利语: `it`
101. 意第绪语: `yi`
102. 印地语: `hi`
103. 印尼巽他语: `su`
104. 印尼语: `id`
105. 印尼爪哇语: `jw`

106. 英语: `en`
107. 约鲁巴语: `yo`
108. 越南语: `vi`
109. 中文: `zh-CN`

2.2 目标语言

1. 阿尔巴尼亚语: `sq`
2. 阿拉伯语: `ar`
3. 阿姆哈拉语: `am`
4. 阿塞拜疆语: `az`
5. 爱尔兰语: `ga`
6. 爱沙尼亚语: `et`
7. 奥利亚语: `or`
8. 巴斯克语: `eu`
9. 白俄罗斯语: `be`
10. 保加利亚语: `bg`
11. 冰岛语: `is`
12. 波兰语: `pl`
13. 波斯尼亚语: `bs`
14. 波斯语: `fa`
15. 布尔语 (南非荷兰语): `af`
16. 鞑靼语: `tt`
17. 丹麦语: `da`
18. 德语: `de`
19. 俄语: `ru`
20. 法语: `fr`
21. 菲律宾语: `tl`
22. 芬兰语: `fi`
23. 弗里西语: `fy`
24. 高棉语: `km`

25. 格鲁吉亚语: `ka`
26. 古吉拉特语: `gu`
27. 哈萨克语: `kk`
28. 海地克里奥尔语: `ht`
29. 韩语: `ko`
30. 豪萨语: `ha`
31. 荷兰语: `nl`
32. 吉尔吉斯语: `ky`
33. 加利西亚语: `gl`
34. 加泰罗尼亚语: `ca`
35. 捷克语: `cs`
36. 卡纳达语: `kn`
37. 科西嘉语: `co`
38. 克罗地亚语: `hr`
39. 库尔德语: `ku`
40. 拉丁语: `la`
41. 拉脱维亚语: `lv`
42. 老挝语: `lo`
43. 立陶宛语: `lt`
44. 卢森堡语: `lb`
45. 卢旺达语: `rw`
46. 罗马尼亚语: `ro`
47. 马尔加什语: `mg`
48. 马耳他语: `mt`
49. 马拉地语: `mr`
50. 马拉雅拉姆语: `ml`
51. 马来语: `ms`
52. 马其顿语: `mk`
53. 毛利语: `mi`
54. 蒙古语: `mn`

55. 孟加拉语: `bn`
56. 缅甸语: `my`
57. 苗语: `hmn`
58. 南非科萨语: `xh`
59. 南非祖鲁语: `zu`
60. 尼泊尔语: `ne`
61. 挪威语: `no`
62. 旁遮普语: `pa`
63. 葡萄牙语: `pt`
64. 普什图语: `ps`
65. 齐切瓦语: `ny`
66. 日语: `ja`
67. 瑞典语: `sv`
68. 萨摩亚语: `sm`
69. 塞尔维亚语: `sr`
70. 塞索托语: `st`
71. 僧伽罗语: `si`
72. 世界语: `eo`
73. 斯洛伐克语: `sk`
74. 斯洛文尼亚语: `sl`
75. 斯瓦希里语: `sw`
76. 苏格兰盖尔语: `gd`
77. 宿务语: `ceb`
78. 索马里语: `so`
79. 塔吉克语: `tg`
80. 泰卢固语: `te`
81. 泰米尔语: `ta`
82. 泰语: `th`
83. 土耳其语: `tr`
84. 土库曼语: `tk`

85. 威尔士语: `cy`
86. 维吾尔语: `ug`
87. 乌尔都语: `ur`
88. 乌克兰语: `uk`
89. 乌兹别克语: `uz`
90. 西班牙语: `es`
91. 希伯来语: `iw`
92. 希腊语: `el`
93. 夏威夷语: `haw`
94. 信德语: `sd`
95. 匈牙利语: `hu`
96. 修纳语: `sn`
97. 亚美尼亚语: `hy`
98. 伊博语: `ig`
99. 意大利语: `it`
100. 意第绪语: `yi`
101. 印地语: `hi`
102. 印尼巽他语: `su`
103. 印尼语: `id`
104. 印尼爪哇语: `jw`
105. 英语: `en`
106. 约鲁巴语: `yo`
107. 越南语: `vi`
108. 中文（繁体）: `zh-TW`
109. 中文（简体）: `zh-CN`

CHAPTER 3

索引和搜索

- genindex
- modindex
- search

CHAPTER 4

项目地址

欢迎大家 访问

CHAPTER 5

依赖

- requests

CHAPTER 6

致谢

- sphinx-doc
- 最应该感谢的 - Google Translate
- 当然还有 Python
- 还有 readthedocs

Python 模块索引

p

pygtrans.ApiKeyTranslate, 3
pygtrans.DetectResponse, 7
pygtrans.LanguageResponse, 6
pygtrans.Null, 6
pygtrans.Translate, 1
pygtrans.TranslateResponse, 7

索引

A

ApiKeyTranslate (*pygtrans.ApiKeyTranslate* 中的类), 4

D

detect() (*pygtrans.ApiKeyTranslate.ApiKeyTranslate* 方法), 4

detect() (*pygtrans.Translate.Translate* 方法), 2

DetectResponse (*pygtrans.DetectResponse* 中的类), 7

L

LanguageResponse (*pygtrans.LanguageResponse* 中的类), 6

languages() (*pygtrans.ApiKeyTranslate.ApiKeyTranslate* 方法), 5

N

Null (*pygtrans.Null* 中的类), 6

P

pygtrans.ApiKeyTranslate
    模块, 3

pygtrans.DetectResponse
    模块, 7

pygtrans.LanguageResponse
    模块, 6

pygtrans.Null
    模块, 6

pygtrans.Translate
    模块, 1

pygtrans.TranslateResponse
    模块, 7

S

split_list() (在 *pygtrans.ApiKeyTranslate* 模块中), 6

split_list_by_content_size() (在 *pygtrans.ApiKeyTranslate* 模块中), 6

T

Translate (*pygtrans.Translate* 中的类), 1

translate() (*pygtrans.ApiKeyTranslate.ApiKeyTranslate* 方法), 5

translate() (*pygtrans.Translate.Translate* 方法), 2

TranslateResponse (*pygtrans.TranslateResponse* 中的类), 7

tts() (*pygtrans.Translate.Translate* 方法), 3

◆

模块
    pygtrans.ApiKeyTranslate, 3
    pygtrans.DetectResponse, 7
    pygtrans.LanguageResponse, 6
    pygtrans.Null, 6
    pygtrans.Translate, 1
    pygtrans.TranslateResponse, 7