

# NLP 技术 API 文档



腾讯云

#### 【版权声明】

©2013–2025 腾讯云版权所有

本文档（含所有文字、数据、图片等内容）完整的著作权归腾讯云计算（北京）有限责任公司单独所有，未经腾讯云事先明确书面许可，任何主体不得以任何形式复制、修改、使用、抄袭、传播本文档全部或部分内容。前述行为构成对腾讯云著作权的侵犯，腾讯云将依法采取措施追究法律责任。

#### 【商标声明】



及其它腾讯云服务相关的商标均为腾讯云计算（北京）有限责任公司及其关联公司所有。本文档涉及的第三方主体的商标，依法由权利人所有。未经腾讯云及有关权利人书面许可，任何主体不得以任何方式对前述商标进行使用、复制、修改、传播、抄录等行为，否则将构成对腾讯云及有关权利人商标权的侵犯，腾讯云将依法采取措施追究法律责任。

#### 【服务声明】

本文档意在向您介绍腾讯云全部或部分产品、服务的当时的相关概况，部分产品、服务的内容可能不时有所调整。

您所购买的腾讯云产品、服务的种类、服务标准等应由您与腾讯云之间的商业合同约定，除非双方另有约定，否则，腾讯云对本文档内容不做任何明示或默示的承诺或保证。

#### 【联系我们】

我们致力于为您提供个性化的售前购买咨询服务，及相应的技术售后服务，任何问题请联系 4009100100或95716。

## 文档目录

### API 文档

更新历史

简介

调用方式

请求结构

公共参数

签名方法 v3

签名方法

返回结果

参数类型

文本生成相关接口

对联生成

诗词生成

文本润色

文本补全

句子生成

句子纠错

文本处理相关接口

文本分类V2

情感分析V2

句子相似度V2

词相似度V2

词法分析V2

相似词召回

数据结构

错误码

附录

词性表

实体类型数据

三级分类体系表

# API 文档

## 更新历史

最近更新时间：2023-09-15 02:14:32

### 第 23 次发布

发布时间：2023-09-15 02:14:19

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- TestingTextGeneration

删除数据结构：

- TextGenerationChoices
- TextGenerationMessage
- TextGenerationUsage

### 第 22 次发布

发布时间：2023-09-14 02:08:26

本次发布包含了以下内容：

改善已有的文档。

预下线接口：

- TestingTextGeneration

### 第 21 次发布

发布时间：2023-08-18 01:15:19

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- TestingTextGeneration

新增数据结构：

- TextGenerationChoices
- TextGenerationMessage
- TextGenerationUsage

### 第 20 次发布

发布时间：2023-08-02 01:19:12

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- AutoSummarization
- ChatBot
- CreateDict
- CreateWordItems
- DeleteDict



- DeleteWordItems
- DependencyParsing
- DescribeDict
- DescribeDicts
- DescribeWordItems
- GenerateCouplet
- GeneratePoetry
- KeywordsExtraction
- LexicalAnalysis
- SearchWordItems
- SentenceEmbedding
- SentimentAnalysis
- SimilarWords
- TextClassification
- TextCorrection
- TextCorrectionPro
- TextSimilarity
- TextSimilarityPro
- UpdateDict
- WordEmbedding
- WordSimilarity

**删除数据结构：**

- CCIToken
- ClassificationResult
- DictInfo
- DpToken
- Keyword
- NerToken
- PosToken
- SearchResult
- Similarity
- WordItem

## 第 19 次发布

发布时间：2023-06-26 15:51:14

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [AnalyzeSentiment](#)
- [ClassifyContent](#)
- [ComposeCouplet](#)
- [ComposePoetry](#)

新增数据结构：

- [Category](#)

## 第 18 次发布

发布时间：2023-05-08 01:41:54

本次发布包含了以下内容：

改善已有的文档。

预下线接口：

- AutoSummarization
- ChatBot
- CreateDict
- CreateWordItems
- DeleteDict
- DeleteWordItems
- DependencyParsing
- DescribeDict
- DescribeDicts
- DescribeWordItems
- GenerateCouplet
- GeneratePoetry
- KeywordsExtraction
- LexicalAnalysis
- SearchWordItems
- SentenceEmbedding
- SentimentAnalysis
- SimilarWords
- TextClassification
- TextCorrection
- TextCorrectionPro
- TextSimilarity
- TextSimilarityPro
- UpdateDict
- WordEmbedding
- WordSimilarity

## 第 17 次发布

发布时间：2023-04-25 13:40:49

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- [EvaluateSentenceSimilarity](#)
- [EvaluateWordSimilarity](#)
- [GenerateKeywordSentence](#)
- [ParseWords](#)
- [RetrieveSimilarWords](#)
- [SentenceCorrection](#)
- [TextEmbellish](#)
- [TextWriting](#)

新增数据结构：

- [BasicParticiple](#)
- [CompoundParticiple](#)
- [CorrectionItem](#)
- [Embellish](#)
- [Entity](#)
- [KeywordSentence](#)
- [SentencePair](#)
- [Writing](#)

## 第 16 次发布

发布时间：2022-12-15 10:27:58

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- GenerateCouplet
- GeneratePoetry

## 第 15 次发布

发布时间：2022-03-31 06:11:26

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- DescribeEntity
- DescribeRelation
- DescribeTriple

删除数据结构：

- EntityRelationContent
- EntityRelationObject
- EntityRelationSubject
- TripleContent

## 第 14 次发布

发布时间：2022-03-18 08:15:22

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- TextCorrectionPro

## 第 13 次发布

发布时间：2022-03-15 08:13:08

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- TextSimilarityPro

## 第 12 次发布

发布时间：2021-05-10 16:12:19

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- ClassificationResult
  - 新增成员：ThirdClassName, ThirdClassProbability, FourthClassName, FourthClassProbability, FifthClassName, FifthClassProbability
- NerToken
  - 修改成员：Word, Length, BeginOffset, Type
- PosToken
  - 修改成员：Word, Length, BeginOffset, Pos

## 第 11 次发布

发布时间：2020-08-05 08:15:46

本次发布包含了以下内容：

改善已有的文档。

修改数据结构：

- WordItem
  - 修改成员：CreateTime

## 第 10 次发布

发布时间：2020-07-09 08:14:16

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- SentenceSimilarity

## 第 9 次发布

发布时间：2020-06-22 08:15:14

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- CreateDict
- CreateWordItems
- DeleteDict
- DeleteWordItems
- DescribeDict
- DescribeDicts
- DescribeWordItems
- SearchWordItems
- UpdateDict

新增数据结构：

- DictInfo
- SearchResult
- WordItem

## 第 8 次发布

发布时间：2020-06-11 08:14:21

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- LexicalAnalysis
  - 新增入参：DictId

## 第 7 次发布

发布时间：2020-05-19 08:13:38

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- TextSimilarity

新增数据结构：

- Similarity

## 第 6 次发布

发布时间：2020-05-08 08:14:08

本次发布包含了以下内容：

改善已有的文档。

修改接口：

- SentimentAnalysis
  - 新增入参：Mode
  - 新增出参：Neutral

## 第 5 次发布

发布时间：2020-02-11 17:03:41

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- SensitiveWordsRecognition
- TextApproval

删除数据结构：

- EvilToken

## 第 4 次发布

发布时间：2020-01-02 21:20:16

本次发布包含了以下内容：

改善已有的文档。

删除接口：

- ContentApproval

## 第 3 次发布

发布时间：2019-11-21 19:33:08

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- ChatBot
- DescribeEntity
- DescribeRelation
- DescribeTriple

新增数据结构：

- EntityRelationContent
- EntityRelationObject
- EntityRelationSubject
- TripleContent

## 第 2 次发布

发布时间：2019-09-19 20:21:32

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- TextApproval

修改接口：

- KeywordsExtraction
  - 新增入参：Num

新增数据结构：

- EvilToken

## 第 1 次发布

发布时间：2019-06-13 19:35:53

本次发布包含了以下内容：

改善已有的文档。

新增接口：

- AutoSummarization
- ContentApproval
- DependencyParsing
- KeywordsExtraction
- LexicalAnalysis
- SensitiveWordsRecognition
- SentenceEmbedding
- SentenceSimilarity
- SentimentAnalysis
- SimilarWords
- TextClassification
- TextCorrection
- WordEmbedding
- WordSimilarity

新增数据结构：

- CCIToken
- ClassificationResult
- DpToken
- Keyword
- NerToken
- PosToken

## 简介

最近更新时间：2025-01-14 01:21:37

欢迎使用 NLP 技术 API **3.0 版本**。全新的 API 接口文档更加规范和全面，统一的参数风格和公共错误码，统一的 SDK/CLI 版本与 API 文档严格一致，给您带来简单快捷的使用体验。支持全地域就近接入让您更快连接腾讯云产品。更多腾讯云 API 3.0 使用介绍请查看：[快速入门](#)

NLP 技术（Natural Language Process，NLP）深度整合了腾讯内部的 NLP 技术，提供多项智能文本处理和文本生成能力，包括词法分析、相似词召回、词相似度、句子相似度、文本润色、句子纠错、文本补全、句子生成等。满足各行业的文本智能需求。

# 调用方式

## 请求结构

最近更新时间：2024-08-28 02:09:23

### 1. 服务地址

API 支持就近地域接入，本产品就近地域接入域名为 `nlp.tencentcloudapi.com`，也支持指定地域域名访问，例如广州地域的域名为 `nlp.ap-guangzhou.tencentcloudapi.com`。

推荐使用就近地域接入域名。根据调用接口时客户端所在位置，会自动解析到最近的某个具体地域的服务器。例如在广州发起请求，会自动解析到广州的服务器，效果和指定 `nlp.ap-guangzhou.tencentcloudapi.com` 是一致的。

**注意：对时延敏感的业务，建议指定带地域的域名。**

**注意：域名是 API 的接入点，并不代表产品或者接口实际提供服务的地域。产品支持的地域列表请在调用方式/公共参数文档中查阅，接口支持的地域请在接口文档输入参数中查阅。**

目前支持的域名列表为：

| 接入地域               | 域名                                       |
|--------------------|------------------------------------------|
| 就近地域接入（推荐，只支持非金融区） | nlp.tencentcloudapi.com                  |
| 华南地区(广州)           | nlp.ap-guangzhou.tencentcloudapi.com     |
| 华东地区(上海)           | nlp.ap-shanghai.tencentcloudapi.com      |
| 华北地区(北京)           | nlp.ap-beijing.tencentcloudapi.com       |
| 西南地区(成都)           | nlp.ap-chengdu.tencentcloudapi.com       |
| 西南地区(重庆)           | nlp.ap-chongqing.tencentcloudapi.com     |
| 港澳台地区(中国香港)        | nlp.ap-hongkong.tencentcloudapi.com      |
| 亚太东南(新加坡)          | nlp.ap-singapore.tencentcloudapi.com     |
| 亚太东南(曼谷)           | nlp.ap-bangkok.tencentcloudapi.com       |
| 亚太南部(孟买)           | nlp.ap-mumbai.tencentcloudapi.com        |
| 亚太东北(首尔)           | nlp.ap-seoul.tencentcloudapi.com         |
| 亚太东北(东京)           | nlp.ap-tokyo.tencentcloudapi.com         |
| 美国东部(弗吉尼亚)         | nlp.na-ashburn.tencentcloudapi.com       |
| 美国西部(硅谷)           | nlp.na-siliconvalley.tencentcloudapi.com |
| 欧洲地区(法兰克福)         | nlp.eu-frankfurt.tencentcloudapi.com     |

### 2. 通信协议

腾讯云 API 的所有接口均通过 HTTPS 进行通信，提供高安全性的通信通道。

### 3. 请求方法

支持的 HTTP 请求方法：

- POST（推荐）
- GET

POST 请求支持的 Content-Type 类型：

- application/json（推荐），必须使用签名方法 v3（TC3-HMAC-SHA256）。
- application/x-www-form-urlencoded，必须使用签名方法 v1（HmacSHA1 或 HmacSHA256）。
- multipart/form-data（仅部分接口支持），必须使用签名方法 v3（TC3-HMAC-SHA256）。



GET 请求的请求包大小不得超过32KB。POST 请求使用签名方法 v1（HmacSHA1、HmacSHA256）时不得超过1MB。POST 请求使用签名方法 v3（TC3-HMAC-SHA256）时支持10MB。

## 4. 字符编码

均使用 UTF-8 编码。

## 公共参数

最近更新时间：2024-11-15 01:52:49

公共参数是用于标识用户和接口签名的参数，如非必要，在每个接口单独的文档中不再对这些参数进行说明，但每次请求均需要携带这些参数，才能正常发起请求。

公共参数的具体内容会因您使用的签名方法版本不同而有所差异。

## 使用签名方法 v3 的公共参数

签名方法 v3（有时也称作 TC3-HMAC-SHA256）相比签名方法 v1（有些文档可能会简称签名方法），更安全，支持更大的请求包，支持 POST JSON 格式，性能有一定提升，推荐使用该签名方法计算签名。完整介绍详见 [签名方法 v3](#)。

注意：出于简化的目的，部分接口文档中的示例使用的是签名方法 v1 GET 请求，而不是更安全的签名方法 v3。

使用签名方法 v3 时，公共参数需要统一放到 HTTP Header 请求头部中，如下表所示：

| 参数名称          | 类型      | 必选 | 描述                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------|---------|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Action        | String  | 是  | HTTP 请求头：X-TC-Action。操作的接口名称。取值参考接口文档输入参数章节关于公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Region        | String  | -  | HTTP 请求头：X-TC-Region。地域参数，用来标识希望操作哪个地域的数据。取值参考接口文档中输入参数章节关于公共参数 Region 的说明。 <b>注意：某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。</b>                                                                                                                                                                                                                                                                                                                                                                                                                          |
| Timestamp     | Integer | 是  | HTTP 请求头：X-TC-Timestamp。当前 UNIX 时间戳，可记录发起 API 请求的时间。例如 1529223702。 <b>注意：如果与服务器时间相差超过5分钟，会引起签名过期错误。</b>                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Version       | String  | 是  | HTTP 请求头：X-TC-Version。操作的 API 的版本。取值参考接口文档中输入公共参数 Version 的说明。例如云服务器的版本 2017-03-12。                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| Authorization | String  | 是  | HTTP 标准身份认证头部字段，例如：<br>TC3-HMAC-SHA256 Credential=AKID***/Date/service/tc3_request, SignedHeaders=content-type;host, Signature=fe5f80f77d5fa3beca038a248ff027d0445342fe2855ddc963176630326f1024<br>其中，<br>- TC3-HMAC-SHA256：签名方法，目前固定取该值；<br>- Credential：签名凭证，AKID*** 是 SecretId；Date 是 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致；service 为具体产品名，通常为域名前缀。例如，域名 cvm.tencentcloudapi.com 意味着产品名是 cvm。本产品取值为 nlp；tc3_request 为固定字符串；<br>- SignedHeaders：参与签名计算的头部信息，content-type 和 host 为必选头部；<br>- Signature：签名摘要，计算过程详见 <a href="#">文档</a> 。 |
| Token         | String  | 否  | HTTP 请求头：X-TC-Token。即 <a href="#">安全凭证服务</a> 所颁发的临时安全凭证中的 Token，使用时需要将 SecretId 和 SecretKey 的值替换为临时安全凭证中的 TmpSecretId 和 TmpSecretKey。使用长期密钥时不能设置此 Token 字段。                                                                                                                                                                                                                                                                                                                                                                                              |
| Language      | String  | 否  | HTTP 请求头：X-TC-Language。指定接口返回的语言，仅部分接口支持此参数。取值：zh-CN，en-US。zh-CN 返回中文，en-US 返回英文。                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |

假设用户想要查询广州地域的云服务器实例列表中的前十个，接口参数设置为偏移量 Offset=0，返回数量 Limit=10，则其请求结构按照请求 URL、请求头部、请求体示例如下：

HTTP GET 请求结构示例：

```
https://cvm.tencentcloudapi.com/?Limit=10&Offset=0

Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2018-10-09/cvm/tc3_request, SignedHeaders=content-type;host, Signature=5da7a33f6993f0614b047e5df4582db9e9bf4672ba50567dba16c6ccf174c474
Content-Type: application/x-www-form-urlencoded
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1539084154
X-TC-Region: ap-guangzhou
```

HTTP POST（application/json）请求结构示例：

```
https://cvm.tencentcloudapi.com/
```

```
Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2018-05-30/cvm/tc3_request, SignedHeaders=content-type;host, Signature=582c400e06b5924a6f2b5d7d672d79c15b13162d9279b0855cfba6789a8edb4c
Content-Type: application/json
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1527672334
X-TC-Region: ap-guangzhou

{"Offset":0,"Limit":10}
```

HTTP POST（multipart/form-data）请求结构示例（仅特定的接口支持）：

```
https://cvm.tencentcloudapi.com/

Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2018-05-30/cvm/tc3_request, SignedHeaders=content-type;host, Signature=582c400e06b5924a6f2b5d7d672d79c15b13162d9279b0855cfba6789a8edb4c
Content-Type: multipart/form-data; boundary=58731222010402
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1527672334
X-TC-Region: ap-guangzhou

--58731222010402
Content-Disposition: form-data; name="Offset"

0
--58731222010402
Content-Disposition: form-data; name="Limit"

10
--58731222010402--
```

## 使用签名方法 v1 的公共参数

使用签名方法 v1（有时会称作 HmacSHA256 和 HmacSHA1），公共参数需要统一放到请求串中，完整介绍详见[文档](#)

| 参数名称            | 类型      | 必选 | 描述                                                                                                                |
|-----------------|---------|----|-------------------------------------------------------------------------------------------------------------------|
| Action          | String  | 是  | 操作的接口名称。取值参考接口文档中输入参数章节关于公共参数 Action 的说明。例如云服务器的查询实例列表接口，取值为 DescribeInstances。                                   |
| Region          | String  | —  | 地域参数，用来标识希望操作哪个地域的数据。接口接受的地域取值参考接口文档中输入参数公共参数 Region 的说明。<br><b>注意：</b> 某些接口不需要传递该参数，接口文档中会对此特别说明，此时即使传递该参数也不会生效。 |
| Timestamp       | Integer | 是  | 当前 UNIX 时间戳，可记录发起 API 请求的时间。例如1529223702，如果与当前时间相差过大，会引起签名过期错误。                                                   |
| Nonce           | Integer | 是  | 随机正整数，与 Timestamp 联合起来，用于防止重放攻击。                                                                                  |
| SecretId        | String  | 是  | 在 <a href="#">云API密钥</a> 上申请的标识身份的 SecretId，一个 SecretId 对应唯一的 SecretKey，而 SecretKey 会用来生成请求签名 Signature。          |
| Signature       | String  | 是  | 请求签名，用来验证此次请求的合法性，需要用户根据实际的输入参数计算得出。具体计算方法参见 <a href="#">文档</a> 。                                                 |
| Version         | String  | 是  | 操作的 API 的版本。取值参考接口文档中入参公共参数 Version 的说明。例如云服务器的版本 2017-03-12。                                                     |
| SignatureMethod | String  | 否  | 签名方式，目前支持 HmacSHA256 和 HmacSHA1。只有指定此参数为 HmacSHA256 时，才使用 HmacSHA256 算法验证签名，其他情况均使用 HmacSHA1 验证签名。                |

| 参数名称     | 类型     | 必选 | 描述                                                                                                                                      |
|----------|--------|----|-----------------------------------------------------------------------------------------------------------------------------------------|
| Token    | String | 否  | 即 <a href="#">安全凭证服务</a> 所颁发的临时安全凭证中的 Token，使用时需要将 SecretId 和 SecretKey 的值替换为临时安全凭证中的 TmpSecretId 和 TmpSecretKey。使用长期密钥时不能设置此 Token 字段。 |
| Language | String | 否  | 指定接口返回的语言，仅部分接口支持此参数。取值：zh-CN，en-US。zh-CN 返回中文，en-US 返回英文。                                                                              |

假设用户想要查询广州地域的云服务器实例列表，其请求结构按照请求 URL、请求头部、请求体示例如下：

HTTP GET 请求结构示例：

```
https://cvm.tencentcloudapi.com/?Action=DescribeInstances&Version=2017-03-12&SignatureMethod=HmacSHA256&Timestamp=1527672334&Signature=37ac2f4fde00b0ac9bd9eadeb459b1bbee224158d66e7ae5fcadb70b2d181d02&Region=ap-guangzhou&Nonce=23823223&SecretId=AKID*****

Host: cvm.tencentcloudapi.com
```

HTTP POST 请求结构示例：

```
https://cvm.tencentcloudapi.com/

Host: cvm.tencentcloudapi.com
Content-Type: application/x-www-form-urlencoded

Action=DescribeInstances&Version=2017-03-12&SignatureMethod=HmacSHA256&Timestamp=1527672334&Signature=37ac2f4fde00b0ac9bd9eadeb459b1bbee224158d66e7ae5fcadb70b2d181d02&Region=ap-guangzhou&Nonce=23823223&SecretId=AKID*****
***
```

# 签名方法 v3

最近更新时间：2024-12-25 01:51:19

以下文档说明了签名方法 v3 的签名过程，但仅在您编写自己的代码来调用腾讯云 API 时才有用。我们推荐您使用 [腾讯云 API Explorer](#)，[腾讯云 SDK](#) 和 [腾讯云命令行工具（TCCLI）](#) 等开发者工具，从而无需学习如何对 API 请求进行签名。

推荐使用 API Explorer

</> 点击调试

您可以通过 API Explorer 的【签名串生成】模块查看每个接口签名的生成过程。

腾讯云 API 会对每个请求进行身份验证，用户需要使用安全凭证，经过特定的步骤对请求进行签名（Signature），每个请求都需要在公共参数中指定该签名结果并以指定的方式和格式发送请求。

## 为什么要进行签名

签名通过以下方式帮助保护请求：

### 1. 验证请求者的身份

签名确保请求是由持有有效访问密钥的人发送的。请参阅控制台 [云 API 密钥](#) 页面获取密钥相关信息。

### 2. 保护传输中的数据

为了防止请求在传输过程中被篡改，腾讯云 API 会使用请求参数来计算请求的哈希值，并将生成的哈希值加密后作为请求的一部分，发送到腾讯云 API 服务器。服务器会使用收到的请求参数以同样的过程计算哈希值，并验证请求中的哈希值。如果请求被篡改，将导致哈希值不一致，腾讯云 API 将拒绝本次请求。

签名方法 v3（TC3-HMAC-SHA256）功能上覆盖了以前的签名方法 v1，而且更安全，支持更大的请求，支持 JSON 格式，POST 请求支持传空数组和空字符串，性能有一定提升，推荐使用该签名方法计算签名。

首次接触，建议使用 [API Explorer](#) 中的“签名串生成”功能，选择签名版本为“API 3.0 签名 v3”，可以对生成签名过程进行验证，也可直接生成 SDK 代码。推荐使用腾讯云 API 配套的 8 种常见的编程语言 SDK，已经封装了签名和请求过程，均已开源，支持 [Python](#)、[Java](#)、[PHP](#)、[Go](#)、[NodeJS](#)、[.NET](#)、[C++](#)、[Ruby](#)。

## 申请安全凭证

本文使用的安全凭证为密钥，密钥包括 SecretId 和 SecretKey。每个用户最多可以拥有两对密钥。

- SecretId：用于标识 API 调用者身份，可以简单类比为用户名。
- SecretKey：用于验证 API 调用者的身份，可以简单类比为密码。
- 用户必须严格保管安全凭证，避免泄露，否则将危及财产安全。如已泄露，请立刻禁用该安全凭证。

申请安全凭证的具体步骤如下：

- 登录 [腾讯云管理中心控制台](#)。
- 前往 [云API密钥](#) 的控制台页面。
- 在 [云API密钥](#) 页面，单击【新建密钥】创建一对密钥。

## 签名版本 v3 签名过程

云 API 支持 GET 和 POST 请求。对于 GET 方法，只支持 Content-Type: application/x-www-form-urlencoded 协议格式。对于 POST 方法，目前支持 Content-Type: application/json 以及 Content-Type: multipart/form-data 两种协议格式，json 格式绝大多数接口均支持，multipart 格式只有特定接口支持，此时该接口不能使用 json 格式调用，参考具体业务接口文档说明。推荐使用 POST 请求，因为两者的结果并无差异，但 GET 请求只支持 32 KB 以内的请求包。

下面以云服务器查询广州实例列表作为例子，分步骤介绍签名的计算过程。我们选择该接口是因为：

- 云服务器默认已开通，该接口很常用；
- 该接口是只读的，不会改变现有资源的状态；
- 接口覆盖的参数种类较全，可以演示包含数据结构的数组如何使用。

在示例中，不论公共参数或者接口的参数，我们尽量选择容易犯错的情况。在实际调用接口时，请根据实际情况来，每个接口的参数并不相同，不要照抄这个例子的参数和值。此外，这里只展示了部分公共参数和接口输入参数，用户可以根据实际需要添加其他参数，例如 Language 和 Token 公共参数（在 HTTP 头部设置，添加 X-TC-前缀）。

假设用户的 SecretId 和 SecretKey 分别是：AKID\*\*\*\*\* 和 \*\*\*\*\*。用户想查看广州区云服务器名为“未命名”的主机状态，只返回一条数据。则请求可能为：

```
curl -X POST https://cvm.tencentcloudapi.com \
-H "Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f" \
-H "Content-Type: application/json; charset=utf-8" \
-H "Host: cvm.tencentcloudapi.com" \
-H "X-TC-Action: DescribeInstances" \
-H "X-TC-Timestamp: 1551113065" \
-H "X-TC-Version: 2017-03-12" \
-H "X-TC-Region: ap-guangzhou" \
-d '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}'
```

下面详细解释签名计算过程。

## 1. 拼接规范请求串

按如下伪代码格式拼接规范请求串（CanonicalRequest）：

```
CanonicalRequest =
HTTPRequestMethod + '\n' +
CanonicalURI + '\n' +
CanonicalQueryString + '\n' +
CanonicalHeaders + '\n' +
SignedHeaders + '\n' +
HashedRequestPayload
```

| 字段名称                 | 解释                                                                                                                                                                                                                                                                                                                                                                                                                |
|----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| HTTPRequestMethod    | HTTP 请求方法（GET、POST）。此示例取值为 POST。                                                                                                                                                                                                                                                                                                                                                                                  |
| CanonicalURI         | URI 参数，API 3.0 固定为正斜杠 (/)。                                                                                                                                                                                                                                                                                                                                                                                        |
| CanonicalQueryString | 发起 HTTP 请求 URL 中的查询字符串，对于 POST 请求，固定为空字符串""，对于 GET 请求，则为 URL 中间号 (?) 后面的字符串内容，例如：Limit=10&Offset=0。<br>注意：CanonicalQueryString 需要参考 <a href="#">RFC3986</a> 进行 URLEncode 编码（特殊字符编码后需大写字母），字符集 UTF-8。推荐使用编程语言标准库进行编码。                                                                                                                                                                                              |
| CanonicalHeaders     | 参与签名的头部信息，至少包含 host 和 content-type 两个头部，也可加入其他头部参与签名以提高自身请求的唯一性和安全性，此示例额外增加了接口名头部。<br>拼接规则：<br>1. 头部 key 和 value 统一转成小写，并去掉首尾空格，按照 key:value\n 格式拼接；<br>2. 多个头部，按照头部 key（小写）的 ASCII 升序进行拼接。<br><br>此示例计算结果是 content-type:application/json; charset=utf-8\nhost:cvm.tencentcloudapi.com\nx-tc-action:describeinstances\n。<br>注意：content-type 必须和实际发送的相符合，有些编程语言网络库即使未指定也会自动添加 charset 值，如果签名时和发送时不一致，服务器会返回签名校验失败。 |
| SignedHeaders        | 参与签名的头部信息，说明此次请求有哪些头部参与了签名，和 CanonicalHeaders 包含的头部内容是一一对应的。content-type 和 host 为必选头部。<br>拼接规则：<br>1. 头部 key 统一转成小写；<br>2. 多个头部 key（小写）按照 ASCII 升序进行拼接，并且以分号 (;) 分隔。<br><br>此示例为 content-type;host;x-tc-action                                                                                                                                                                                                    |
| HashedRequestPayload | 请求正文（payload，即 body，此示例为 {"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}）的哈希值，计算伪代码为 Lowercase(HexEncode(Hash.SHA256(RequestPayload)))，即对 HTTP 请求正文做 SHA256 哈希，然后十六进制编                                                                                                                                                                                                     |

| 字段名称 | 解释                                                                                                                         |
|------|----------------------------------------------------------------------------------------------------------------------------|
|      | 码，最后编码串转换成小写字母。对于 GET 请求，RequestPayload 固定为空字符串。此示例计算结果是 35e9c5b0e3ae67532d3c9f17ead6c90222632e5b1ff7f6e89887f1398934f064。 |

根据以上规则，示例中得到的规范请求串如下：

```
POST
/

content-type:application/json; charset=utf-8
host:cvm.tencentcloudapi.com
x-tc-action:describeinstances

content-type;host;x-tc-action
35e9c5b0e3ae67532d3c9f17ead6c90222632e5b1ff7f6e89887f1398934f064
```

## 2. 拼接待签名字符串

按如下格式拼接待签名字符串：

```
StringToSign =
Algorithm + "\n" +
RequestTimestamp + "\n" +
CredentialScope + "\n" +
HashedCanonicalRequest
```

| 字段名称                   | 解释                                                                                                                                                                                           |
|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Algorithm              | 签名算法，目前固定为 TC3-HMAC-SHA256。                                                                                                                                                                  |
| RequestTimestamp       | 请求时间戳，即请求头部的公共参数 X-TC-Timestamp 取值，取当前时间 UNIX 时间戳，精确到秒。此示例取值为 1551113065。                                                                                                                    |
| CredentialScope        | 凭证范围，格式为 Date/service/tc3_request，包含日期、所请求的服务和终止字符串（tc3_request）。Date 为 UTC 标准时间的日期，取值需要和公共参数 X-TC-Timestamp 换算的 UTC 标准时间日期一致；service 为产品名，必须与调用的产品域名一致。此示例计算结果是 2019-02-25/cvm/tc3_request。 |
| HashedCanonicalRequest | 前述步骤拼接所得规范请求串的哈希值，计算伪代码为 Lowercase(HexEncode(Hash.SHA256(CanonicalRequest)))。此示例计算结果是 7019a55be8395899b900fb5564e4200d984910f34794a27cb3fb7d10ff6a1e84。                                      |

注意：

1. Date 必须从时间戳 X-TC-Timestamp 计算得到，且时区为 UTC+0。如果加入系统本地时区信息，例如东八区，将导致白天和晚上调用成功，但是凌晨时调用必定失败。假设时间戳为 1551113065，在东八区的时间是 2019-02-26 00:44:25，但是计算得到的 Date 取 UTC+0 的日期应为 2019-02-25，而不是 2019-02-26。
2. Timestamp 必须是当前系统时间，且需确保系统时间和标准时间是同步的，如果相差超过五分钟则必定失败。如果长时间不和标准时间同步，可能运行一段时间后，请求失败，返回签名过期错误。

根据以上规则，示例中得到的待签名字符串如下：

```
TC3-HMAC-SHA256
1551113065
2019-02-25/cvm/tc3_request
7019a55be8395899b900fb5564e4200d984910f34794a27cb3fb7d10ff6a1e84
```

### 3. 计算签名

1) 计算派生签名密钥，伪代码如下：

```
SecretKey = "*****"  
SecretDate = HMAC_SHA256("TC3" + SecretKey, Date)  
SecretService = HMAC_SHA256(SecretDate, Service)  
SecretSigning = HMAC_SHA256(SecretService, "tc3_request")
```

派生出的密钥 SecretDate、SecretService 和 SecretSigning 是二进制的数，可能包含不可打印字符，将其转为十六进制字符串打印的输出分别为：  
da98fb70dcf6b112dc21038d1eeeb3a95c74b4dcb12c1131f864f6066bd02be0，  
8d70cbefb03939f929db64d32dc2ba89b1095620119fe3e050e2b18c5bd2752f，  
b596b923aad85185e2d1f6659d2a062e0a86731226e021e61bfe06f7ed05f5af。

请注意，不同的编程语言，HMAC 库函数中参数顺序可能不一样，请以实际情况为准。此处的伪代码密钥参数 key 在前，消息参数 data 在后。通常标准库函数会提供二进制格式的返回值，也可能会提供打印友好的十六进制格式的返回值，此处使用的是二进制格式。

| 字段名称      | 解释                                           |
|-----------|----------------------------------------------|
| SecretKey | 原始的 SecretKey，即 *****。                       |
| Date      | 即 Credential 中的 Date 字段信息。此示例取值为 2019-02-25。 |
| Service   | 即 Credential 中的 Service 字段信息。此示例取值为 cvm。     |

2) 计算签名，伪代码如下：

```
Signature = HexEncode(HMAC_SHA256(SecretSigning, StringToSign))
```

此示例计算结果是 10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f。

### 4. 拼接 Authorization

按如下格式拼接 Authorization：

```
Authorization =  
Algorithm + ' ' +  
'Credential=' + SecretId + '/' + CredentialScope + ', ' +  
'SignedHeaders=' + SignedHeaders + ', ' +  
'Signature=' + Signature
```

| 字段名称            | 解释                                                                             |
|-----------------|--------------------------------------------------------------------------------|
| Algorithm       | 签名方法，固定为 TC3-HMAC-SHA256。                                                      |
| SecretId        | 密钥对中的 SecretId，即 AKID*****。                                                    |
| CredentialScope | 见上文，凭证范围。此示例计算结果是 2019-02-25/cvm/tc3_request。                                  |
| SignedHeaders   | 见上文，参与签名的头部信息。此示例取值为 content-type;host;x-tc-action。                            |
| Signature       | 签名值。此示例计算结果是 10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f。 |

根据以上规则，示例中得到的值为：

```
TC3-HMAC-SHA256 Credential=AKID*****/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f
```

最终完整的调用信息如下：



```
POST https://cvm.tencentcloudapi.com/
Authorization: TC3-HMAC-SHA256 Credential=AKID*****/2019-02-25/cvm/tc3_request, SignedHeaders=content-type;host;x-tc-action, Signature=10b1a37a7301a02ca19a647ad722d5e43b4b3cff309d421d85b46093f6ab6c4f
Content-Type: application/json; charset=utf-8
Host: cvm.tencentcloudapi.com
X-TC-Action: DescribeInstances
X-TC-Version: 2017-03-12
X-TC-Timestamp: 1551113065
X-TC-Region: ap-guangzhou

{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}
```

#### ⚠ 注意：

请求发送时的 HTTP 头部（Header）和请求体（Payload）必须和签名计算过程中的内容完全一致，否则会返回签名不一致错误。可以通过打印实际请求内容，网络抓包等方式对比排查。

## 签名演示

在实际调用 API 3.0 时，推荐使用配套的腾讯云 SDK 3.0，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 [SDK 中心](#)。当前支持的编程语言有：

- [Python](#)
- [Java](#)
- [PHP](#)
- [Go](#)
- [NodeJS](#)
- [.NET](#)
- [C++](#)
- [Ruby](#)

下面提供了不同产品的生成签名 demo，您可以找到对应的产品参考签名的生成：

- [Signature Demo](#)

为了更清楚地解释签名过程，下面以实际编程语言为例，将上述的签名过程完整实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK。

### Java

```
import java.nio.charset.Charset;
import java.nio.charset.StandardCharsets;
import java.security.MessageDigest;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.TimeZone;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPITC3Demo {
    private final static Charset UTF8 = StandardCharsets.UTF_8;
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    private final static String SECRET_ID = System.getenv("TENCENTCLOUD_SECRET_ID");
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
}
```

```
private final static String SECRET_KEY = System.getenv("TENCENTCLOUD_SECRET_KEY");
private final static String CT_JSON = "application/json; charset=utf-8";

public static byte[] hmac256(byte[] key, String msg) throws Exception {
    Mac mac = Mac.getInstance("HmacSHA256");
    SecretKeySpec secretKeySpec = new SecretKeySpec(key, mac.getAlgorithm());
    mac.init(secretKeySpec);
    return mac.doFinal(msg.getBytes(UTF8));
}

public static String sha256Hex(String s) throws Exception {
    MessageDigest md = MessageDigest.getInstance("SHA-256");
    byte[] d = md.digest(s.getBytes(UTF8));
    return DatatypeConverter.printHexBinary(d).toLowerCase();
}

public static void main(String[] args) throws Exception {
    String service = "cvm";
    String host = "cvm.tencentcloudapi.com";
    String region = "ap-guangzhou";
    String action = "DescribeInstances";
    String version = "2017-03-12";
    String algorithm = "TC3-HMAC-SHA256";
    String timestamp = "1551113065";
    //String timestamp = String.valueOf(System.currentTimeMillis() / 1000);
    SimpleDateFormat sdf = new SimpleDateFormat("yyyy-MM-dd");
    // 注意时区, 否则容易出错
    sdf.setTimeZone(TimeZone.getTimeZone("UTC"));
    String date = sdf.format(new Date(Long.valueOf(timestamp + "000")));

    // ***** 步骤 1: 拼接规范请求串 *****
    String httpRequestMethod = "POST";
    String canonicalUri = "/";
    String canonicalQueryString = "";
    String canonicalHeaders = "content-type:application/json; charset=utf-8\n"
        + "host:" + host + "\n" + "x-tc-action:" + action.toLowerCase() + "\n";
    String signedHeaders = "content-type;host;x-tc-action";

    String payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}] }";
    String hashedRequestPayload = sha256Hex(payload);
    String canonicalRequest = httpRequestMethod + "\n" + canonicalUri + "\n" + canonicalQueryString + "\n"
        + canonicalHeaders + "\n" + signedHeaders + "\n" + hashedRequestPayload;
    System.out.println(canonicalRequest);

    // ***** 步骤 2: 拼接待签名字符串 *****
    String credentialScope = date + "/" + service + "/" + "tc3_request";
    String hashedCanonicalRequest = sha256Hex(canonicalRequest);
    String stringToSign = algorithm + "\n" + timestamp + "\n" + credentialScope + "\n" + hashedCanonicalRequest;
    System.out.println(stringToSign);

    // ***** 步骤 3: 计算签名 *****
    byte[] secretDate = hmac256(("TC3" + SECRET_KEY).getBytes(UTF8), date);
    byte[] secretService = hmac256(secretDate, service);
    byte[] secretSigning = hmac256(secretService, "tc3_request");
    String signature = DatatypeConverter.printHexBinary(hmac256(secretSigning, stringToSign)).toLowerCase();
    System.out.println(signature);
}
```

```
// ***** 步骤 4: 拼接 Authorization *****

String authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;

System.out.println(authorization);

TreeMap<String, String> headers = new TreeMap<String, String>();
headers.put("Authorization", authorization);
headers.put("Content-Type", CT_JSON);
headers.put("Host", host);
headers.put("X-TC-Action", action);
headers.put("X-TC-Timestamp", timestamp);
headers.put("X-TC-Version", version);
headers.put("X-TC-Region", region);

StringBuilder sb = new StringBuilder();
sb.append("curl -X POST https://").append(host)
.append(" -H \"Authorization: ").append(authorization).append("\")")
.append(" -H \"Content-Type: application/json; charset=utf-8\"")
.append(" -H \"Host: ").append(host).append("\")")
.append(" -H \"X-TC-Action: ").append(action).append("\")")
.append(" -H \"X-TC-Timestamp: ").append(timestamp).append("\")")
.append(" -H \"X-TC-Version: ").append(version).append("\")")
.append(" -H \"X-TC-Region: ").append(region).append("\")")
.append(" -d ").append(payload).append("");
System.out.println(sb.toString());
}
}
```

## Python

```
# -*- coding: utf-8 -*-
import hashlib, hmac, json, os, sys, time
from datetime import datetime

# 密钥参数
# 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
secret_id = os.environ.get("TENCENTCLOUD_SECRET_ID")
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
secret_key = os.environ.get("TENCENTCLOUD_SECRET_KEY")

service = "cvm"
host = "cvm.tencentcloudapi.com"
endpoint = "https://" + host
region = "ap-guangzhou"
action = "DescribeInstances"
version = "2017-03-12"
algorithm = "TC3-HMAC-SHA256"
#timestamp = int(time.time())
timestamp = 1551113065
date = datetime.utcfromtimestamp(timestamp).strftime("%Y-%m-%d")
params = {"Limit": 1, "Filters": [{"Values": [u"未命名"], "Name": "instance-name"}]}

# ***** 步骤 1: 拼接规范请求串 *****

http_request_method = "POST"
canonical_uri = "/"
canonical_querystring = ""
```

```
ct = "application/json; charset=utf-8"
payload = json.dumps(params)
canonical_headers = "content-type:%s\nhost:%s\nx-tc-action:%s\n" % (ct, host, action.lower())
signed_headers = "content-type;host;x-tc-action"
hashed_request_payload = hashlib.sha256(payload.encode("utf-8")).hexdigest()
canonical_request = (http_request_method + "\n" +
canonical_uri + "\n" +
canonical_querystring + "\n" +
canonical_headers + "\n" +
signed_headers + "\n" +
hashed_request_payload)
print(canonical_request)

# ***** 步骤 2: 拼接待签名字符串 *****
credential_scope = date + "/" + service + "/" + "tc3_request"
hashed_canonical_request = hashlib.sha256(canonical_request.encode("utf-8")).hexdigest()
string_to_sign = (algorithm + "\n" +
str(timestamp) + "\n" +
credential_scope + "\n" +
hashed_canonical_request)
print(string_to_sign)

# ***** 步骤 3: 计算签名 *****
# 计算签名摘要函数
def sign(key, msg):
return hmac.new(key, msg.encode("utf-8"), hashlib.sha256).digest()
secret_date = sign(("TC3" + secret_key).encode("utf-8"), date)
secret_service = sign(secret_date, service)
secret_signing = sign(secret_service, "tc3_request")
signature = hmac.new(secret_signing, string_to_sign.encode("utf-8"), hashlib.sha256).hexdigest()
print(signature)

# ***** 步骤 4: 拼接 Authorization *****
authorization = (algorithm + " " +
"Credential=" + secret_id + "/" + credential_scope + ", " +
"SignedHeaders=" + signed_headers + ", " +
"Signature=" + signature)
print(authorization)

print('curl -X POST ' + endpoint
+ ' -H "Authorization: ' + authorization + '"
+ ' -H "Content-Type: application/json; charset=utf-8"
+ ' -H "Host: ' + host + '"
+ ' -H "X-TC-Action: ' + action + '"
+ ' -H "X-TC-Timestamp: ' + str(timestamp) + '"
+ ' -H "X-TC-Version: ' + version + '"
+ ' -H "X-TC-Region: ' + region + '"
+ " -d '" + payload + "'")
```

## Golang

```
package main

import (
"crypto/hmac"
```

```
"crypto/sha256"
"encoding/hex"
"fmt"
"os"
"strings"
"time"
)

func sha256hex(s string) string {
    b := sha256.Sum256([]byte(s))
    return hex.EncodeToString(b[:])
}

func hmacsha256(s, key string) string {
    hashed := hmac.New(sha256.New, []byte(key))
    hashed.Write([]byte(s))
    return string(hashed.Sum(nil))
}

func main() {
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    secretId := os.Getenv("TENCENTCLOUD_SECRET_ID")
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
    secretKey := os.Getenv("TENCENTCLOUD_SECRET_KEY")
    host := "cvm.tencentcloudapi.com"
    algorithm := "TC3-HMAC-SHA256"
    service := "cvm"
    version := "2017-03-12"
    action := "DescribeInstances"
    region := "ap-guangzhou"
    //var timestamp int64 = time.Now().Unix()
    var timestamp int64 = 1551113065

    // step 1: build canonical request string
    httpRequestMethod := "POST"
    canonicalURI := "/"
    canonicalQueryString := ""
    canonicalHeaders := fmt.Sprintf("content-type:%s\nhost:%s\nx-tc-action:%s\n",
        "application/json; charset=utf-8", host, strings.ToLower(action))
    signedHeaders := "content-type;host;x-tc-action"
    payload := `{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}`
    hashedRequestPayload := sha256hex(payload)
    canonicalRequest := fmt.Sprintf("%s\n%s\n%s\n%s\n%s\n%s",
        httpRequestMethod,
        canonicalURI,
        canonicalQueryString,
        canonicalHeaders,
        signedHeaders,
        hashedRequestPayload)
    fmt.Println(canonicalRequest)

    // step 2: build string to sign
    date := time.Unix(timestamp, 0).UTC().Format("2006-01-02")
    credentialScope := fmt.Sprintf("%s/%s/tc3_request", date, service)
    hashedCanonicalRequest := sha256hex(canonicalRequest)
    string2sign := fmt.Sprintf("%s\n%d\n%s\n%s",
        algorithm,
```

```
timestamp,
credentialScope,
hashedCanonicalRequest)
fmt.Println(string2sign)

// step 3: sign string
secretDate := hmacsha256(date, "TC3"+secretKey)
secretService := hmacsha256(service, secretDate)
secretSigning := hmacsha256("tc3_request", secretService)
signature := hex.EncodeToString([]byte(hmacsha256(string2sign, secretSigning)))
fmt.Println(signature)

// step 4: build authorization
authorization := fmt.Sprintf("%s Credential=%s/%s, SignedHeaders=%s, Signature=%s",
algorithm,
secretId,
credentialScope,
signedHeaders,
signature)
fmt.Println(authorization)

curl := fmt.Sprintf(`curl -X POST https://%s\
-H "Authorization: %s"\
-H "Content-Type: application/json; charset=utf-8"\
-H "Host: %s" -H "X-TC-Action: %s"\
-H "X-TC-Timestamp: %d"\
-H "X-TC-Version: %s"\
-H "X-TC-Region: %s"\
-d '%s'`, host, authorization, host, action, timestamp, version, region, payload)
fmt.Println(curl)
}
```

## PHP

```
<?php
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$host = "cvm.tencentcloudapi.com";
$service = "cvm";
$version = "2017-03-12";
$action = "DescribeInstances";
$region = "ap-guangzhou";
// $timestamp = time();
$timestamp = 1551113065;
$algorithm = "TC3-HMAC-SHA256";

// step 1: build canonical request string
$httpRequestMethod = "POST";
$canonicalUri = "/";
$canonicalQueryString = "";
$canonicalHeaders = implode("\n", [
"content-type:application/json; charset=utf-8",
"host:".$host,
"x-tc-action:".strtolower($action),
```

```
""
});
$signedHeaders = implode(";", [
    "content-type",
    "host",
    "x-tc-action",
]);
$payload = '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]';
$hashedRequestPayload = hash("SHA256", $payload);
$canonicalRequest = $httpRequestMethod."\n"
.$canonicalUri."\n"
.$canonicalQueryString."\n"
.$canonicalHeaders."\n"
.$signedHeaders."\n"
.$hashedRequestPayload;
echo $canonicalRequest.PHP_EOL;

// step 2: build string to sign
$date = gmdate("Y-m-d", $timestamp);
$credentialScope = $date."/".$service."/tc3_request";
$hashedCanonicalRequest = hash("SHA256", $canonicalRequest);
$stringToSign = $algorithm."\n"
.$timestamp."\n"
.$credentialScope."\n"
.$hashedCanonicalRequest;
echo $stringToSign.PHP_EOL;

// step 3: sign string
$secretDate = hash_hmac("SHA256", $date, "TC3".$secretKey, true);
$secretService = hash_hmac("SHA256", $service, $secretDate, true);
$secretSigning = hash_hmac("SHA256", "tc3_request", $secretService, true);
$signature = hash_hmac("SHA256", $stringToSign, $secretSigning);
echo $signature.PHP_EOL;

// step 4: build authorization
$authorization = $algorithm
." Credential=". $secretId."/". $credentialScope
.", SignedHeaders=". $signedHeaders.", Signature=". $signature;
echo $authorization.PHP_EOL;

$curl = "curl -X POST https://".$host
.' -H "Authorization: '.$authorization.'"
.' -H "Content-Type: application/json; charset=utf-8"
.' -H "Host: '.$host.'"
.' -H "X-TC-Action: '.$action.'"
.' -H "X-TC-Timestamp: '.$timestamp.'"
.' -H "X-TC-Version: '.$version.'"
.' -H "X-TC-Region: '.$region.'"
.' -d "'.$payload.'"";
echo $curl.PHP_EOL;
```

## Ruby

```
# -*- coding: UTF-8 -*-
# require ruby>=2.3.0
require 'digest'
```

```
require 'json'
require 'time'
require 'openssl'

# 密钥参数
# 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
secret_id = ENV["TENCENTCLOUD_SECRET_ID"]
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
secret_key = ENV["TENCENTCLOUD_SECRET_KEY"]

service = 'cvm'
host = 'cvm.tencentcloudapi.com'
endpoint = 'https://' + host
region = 'ap-guangzhou'
action = 'DescribeInstances'
version = '2017-03-12'
algorithm = 'TC3-HMAC-SHA256'
# timestamp = Time.now.to_i
timestamp = 1551113065
date = Time.at(timestamp).utc.strftime('%Y-%m-%d')

# ***** 步骤 1: 拼接规范请求串 *****
http_request_method = 'POST'
canonical_uri = '/'
canonical_querystring = ''
canonical_headers = "content-type:application/json; charset=utf-8\nhost:#{host}\n-x-tc-action:#{action.downcase}\n"
signed_headers = 'content-type;host;x-tc-action'
# params = { 'Limit' => 1, 'Filters' => [{ 'Name' => 'instance-name', 'Values' => ['未命名'] }] }
# payload = JSON.generate(params, { 'ascii_only' => true, 'space' => ' ' })
# json will generate in random order, to get specified result in example, we hard-code it here.
payload = '{"Limit": 1, "Filters": [{"Values": ["\u672a\u547d\u540d"], "Name": "instance-name"}]}'
hashed_request_payload = Digest::SHA256.hexdigest(payload)
canonical_request = [
  http_request_method,
  canonical_uri,
  canonical_querystring,
  canonical_headers,
  signed_headers,
  hashed_request_payload,
].join("\n")

puts canonical_request

# ***** 步骤 2: 拼接待签名字符串 *****
credential_scope = date + '/' + service + '/' + 'tc3_request'
hashed_request_payload = Digest::SHA256.hexdigest(canonical_request)
string_to_sign = [
  algorithm,
  timestamp.to_s,
  credential_scope,
  hashed_request_payload,
].join("\n")
puts string_to_sign

# ***** 步骤 3: 计算签名 *****
digest = OpenSSL::Digest.new('sha256')
secret_date = OpenSSL::HMAC.digest(digest, 'TC3' + secret_key, date)
```



```
secret_service = OpenSSL::HMAC.digest(digest, secret_date, service)
secret_signing = OpenSSL::HMAC.digest(digest, secret_service, 'tc3_request')
signature = OpenSSL::HMAC.hexdigest(digest, secret_signing, string_to_sign)
puts signature

# ***** 步骤 4: 拼接 Authorization *****
authorization = "#{algorithm} Credential=#{secret_id}/#{credential_scope}, SignedHeaders=#{signed_headers}, Signature=#{signature}"
puts authorization

puts 'curl -X POST ' + endpoint \
+ ' -H "Authorization: ' + authorization + '"' \
+ ' -H "Content-Type: application/json; charset=utf-8"' \
+ ' -H "Host: ' + host + '"' \
+ ' -H "X-TC-Action: ' + action + '"' \
+ ' -H "X-TC-Timestamp: ' + timestamp.to_s + '"' \
+ ' -H "X-TC-Version: ' + version + '"' \
+ ' -H "X-TC-Region: ' + region + '"' \
+ " -d '" + payload + '"'
```

## DotNet

```
using System;
using System.Collections.Generic;
using System.Security.Cryptography;
using System.Text;

public class Application
{
    public static string SHA256Hex(string s)
    {
        using (SHA256 algo = SHA256.Create())
        {
            byte[] hashbytes = algo.ComputeHash(Encoding.UTF8.GetBytes(s));
            StringBuilder builder = new StringBuilder();
            for (int i = 0; i < hashbytes.Length; ++i)
            {
                builder.Append(hashbytes[i].ToString("x2"));
            }
            return builder.ToString();
        }
    }

    public static byte[] HmacSHA256(byte[] key, byte[] msg)
    {
        using (HMACSHA256 mac = new HMACSHA256(key))
        {
            return mac.ComputeHash(msg);
        }
    }

    public static Dictionary<String, String> BuildHeaders(string secretid,
        string secretkey, string service, string endpoint, string region,
        string action, string version, DateTime date, string requestPayload)
    {
        string datestr = date.ToString("yyyy-MM-dd");
```

```
DateTime startTime = new DateTime(1970, 1, 1, 0, 0, 0, 0, DateTimeKind.Utc);
long requestTimestamp = (long)Math.Round((date - startTime).TotalMilliseconds, MidpointRounding.AwayFromZero) / 1000;
// ***** 步骤 1: 拼接规范请求串 *****

string algorithm = "TC3-HMAC-SHA256";
string httpRequestMethod = "POST";
string canonicalUri = "/";
string canonicalQueryString = "";
string contentType = "application/json";
string canonicalHeaders = "content-type:" + contentType + "; charset=utf-8\n"
+ "host:" + endpoint + "\n"
+ "x-tc-action:" + action.ToLower() + "\n";
string signedHeaders = "content-type;host;x-tc-action";
string hashedRequestPayload = SHA256Hex(requestPayload);
string canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload;
Console.WriteLine(canonicalRequest);

// ***** 步骤 2: 拼接待签名字符串 *****

string credentialScope = datestr + "/" + service + "/" + "tc3_request";
string hashedCanonicalRequest = SHA256Hex(canonicalRequest);
string stringToSign = algorithm + "\n"
+ requestTimestamp.ToString() + "\n"
+ credentialScope + "\n"
+ hashedCanonicalRequest;
Console.WriteLine(stringToSign);

// ***** 步骤 3: 计算签名 *****

byte[] tc3SecretKey = Encoding.UTF8.GetBytes("TC3" + secretkey);
byte[] secretDate = HmacSHA256(tc3SecretKey, Encoding.UTF8.GetBytes(datestr));
byte[] secretService = HmacSHA256(secretDate, Encoding.UTF8.GetBytes(service));
byte[] secretSigning = HmacSHA256(secretService, Encoding.UTF8.GetBytes("tc3_request"));
byte[] signatureBytes = HmacSHA256(secretSigning, Encoding.UTF8.GetBytes(stringToSign));
string signature = BitConverter.ToString(signatureBytes).Replace("-", "").ToLower();
Console.WriteLine(signature);

// ***** 步骤 4: 拼接 Authorization *****

string authorization = algorithm + " "
+ "Credential=" + secretid + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", "
+ "Signature=" + signature;
Console.WriteLine(authorization);

Dictionary<string, string> headers = new Dictionary<string, string>();
headers.Add("Authorization", authorization);
headers.Add("Host", endpoint);
headers.Add("Content-Type", contentType + "; charset=utf-8");
headers.Add("X-TC-Timestamp", requestTimestamp.ToString());
headers.Add("X-TC-Version", version);
headers.Add("X-TC-Action", action);
headers.Add("X-TC-Region", region);
return headers;
}

public static void Main(string[] args)
```

```
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
string SECRET_ID = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
string SECRET_KEY = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_KEY");

string service = "cvm";
string endpoint = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";

// 此处由于示例规范的原因, 采用时间戳2019-02-26 00:44:25, 此参数作为示例, 如果在项目中, 您应当使用:
// DateTime date = DateTime.UtcNow;
// 注意时区, 建议此时间统一采用UTC时间戳, 则容易出错
DateTime date = new DateTime(1970, 1, 1, 0, 0, 0, 0, DateTimeKind.Utc).AddSeconds(1551113065);
string requestPayload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]\"}";

Dictionary<string, string> headers = BuildHeaders(SECRET_ID, SECRET_KEY, service
, endpoint, region, action, version, date, requestPayload);

Console.WriteLine("POST https://cvm.tencentcloudapi.com");
foreach (KeyValuePair<string, string> kv in headers)
{
Console.WriteLine(kv.Key + ": " + kv.Value);
}
Console.WriteLine();
Console.WriteLine(requestPayload);
}
}
```

## NodeJS

```
const crypto = require('crypto');

function sha256(message, secret = '', encoding) {
const hmac = crypto.createHmac('sha256', secret)
return hmac.update(message).digest(encoding)
}

function getHash(message, encoding = 'hex') {
const hash = crypto.createHash('sha256')
return hash.update(message).digest(encoding)
}

function getDate(timestamp) {
const date = new Date(timestamp * 1000)
const year = date.getUTCFullYear()
const month = ('0' + (date.getUTCMonth() + 1)).slice(-2)
const day = ('0' + date.getUTCDate()).slice(-2)
return `${year}-${month}-${day}`
}

function main(){
```

```
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
const SECRET_ID = process.env.TENCENTCLOUD_SECRET_ID
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
const SECRET_KEY = process.env.TENCENTCLOUD_SECRET_KEY

const endpoint = "cvm.tencentcloudapi.com"
const service = "cvm"
const region = "ap-guangzhou"
const action = "DescribeInstances"
const version = "2017-03-12"
//const timestamp = getTime()
const timestamp = 1551113065
//时间处理, 获取世界时间日期
const date = getDate(timestamp)

// ***** 步骤 1: 拼接规范请求串 *****
const payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}]}"

const hashedRequestPayload = getHash(payload);
const httpRequestMethod = "POST"
const canonicalUri = "/"
const canonicalQueryString = ""
const canonicalHeaders = "content-type:application/json; charset=utf-8\\n"
+ "host:" + endpoint + "\\n"
+ "x-tc-action:" + action.toLowerCase() + "\\n"
const signedHeaders = "content-type;host;x-tc-action"

const canonicalRequest = httpRequestMethod + "\\n"
+ canonicalUri + "\\n"
+ canonicalQueryString + "\\n"
+ canonicalHeaders + "\\n"
+ signedHeaders + "\\n"
+ hashedRequestPayload
console.log(canonicalRequest)

// ***** 步骤 2: 拼接待签名字符串 *****
const algorithm = "TC3-HMAC-SHA256"
const hashedCanonicalRequest = getHash(canonicalRequest);
const credentialScope = date + "/" + service + "/" + "tc3_request"
const stringToSign = algorithm + "\\n" +
timestamp + "\\n" +
credentialScope + "\\n" +
hashedCanonicalRequest
console.log(stringToSign)

// ***** 步骤 3: 计算签名 *****
const kDate = sha256(date, 'TC3' + SECRET_KEY)
const kService = sha256(service, kDate)
const kSigning = sha256('tc3_request', kService)
const signature = sha256(stringToSign, kSigning, 'hex')
console.log(signature)

// ***** 步骤 4: 拼接 Authorization *****
const authorization = algorithm + " " +
"Credential=" + SECRET_ID + "/" + credentialScope + ", " +
"SignedHeaders=" + signedHeaders + ", " +
```

```

"Signature=" + signature
console.log(authorization)

const curlcmd = 'curl -X POST ' + "https://" + endpoint
+ ' -H "Authorization: ' + authorization + '"'
+ ' -H "Content-Type: application/json; charset=utf-8"'
+ ' -H "Host: ' + endpoint + '"'
+ ' -H "X-TC-Action: ' + action + '"'
+ ' -H "X-TC-Timestamp: ' + timestamp.toString() + '"'
+ ' -H "X-TC-Version: ' + version + '"'
+ ' -H "X-TC-Region: ' + region + '"'
+ " -d '" + payload + '"'
console.log(curlcmd)
}
main()

```

## C++

```

#include <algorithm>
#include <cstdlib>
#include <iostream>
#include <iomanip>
#include <sstream>
#include <string>
#include <stdio.h>
#include <time.h>
#include <openssl/sha.h>
#include <openssl/hmac.h>

using namespace std;

string get_data(int64_t &timestamp)
{
    string utcDate;
    char buff[20] = {0};
    // time_t timenow;
    struct tm sttime;
    sttime = *gmtime(&timestamp);
    strftime(buff, sizeof(buff), "%Y-%m-%d", &sttime);
    utcDate = string(buff);
    return utcDate;
}

string int2str(int64_t n)
{
    std::stringstream ss;
    ss << n;
    return ss.str();
}

string sha256Hex(const string &str)
{
    char buf[3];
    unsigned char hash[SHA256_DIGEST_LENGTH];
    SHA256_CTX sha256;
    SHA256_Init(&sha256);

```

```
SHA256_Update(&sha256, str.c_str(), str.size());
SHA256_Final(hash, &sha256);
std::string NewString = "";
for(int i = 0; i < SHA256_DIGEST_LENGTH; i++)
{
    snprintf(buf, sizeof(buf), "%02x", hash[i]);
    NewString = NewString + buf;
}
return NewString;
}

string HmacSha256(const string &key, const string &input)
{
    unsigned char hash[32];

    HMAC_CTX *h;
    #if OPENSSL_VERSION_NUMBER < 0x10100000L
    HMAC_CTX hmac;
    HMAC_CTX_init(&hmac);
    h = &hmac;
    #else
    h = HMAC_CTX_new();
    #endif

    HMAC_Init_ex(h, &key[0], key.length(), EVP_sha256(), NULL);
    HMAC_Update(h, ( unsigned char* )&input[0], input.length());
    unsigned int len = 32;
    HMAC_Final(h, hash, &len);

    #if OPENSSL_VERSION_NUMBER < 0x10100000L
    HMAC_CTX_cleanup(h);
    #else
    HMAC_CTX_free(h);
    #endif

    std::stringstream ss;
    ss << std::setfill('0');
    for (int i = 0; i < len; i++)
    {
        ss << hash[i];
    }

    return (ss.str());
}

string HexEncode(const string &input)
{
    static const char* const lut = "0123456789abcdef";
    size_t len = input.length();

    string output;
    output.reserve(2 * len);
    for (size_t i = 0; i < len; ++i)
    {
        const unsigned char c = input[i];
        output.push_back(lut[c >> 4]);
        output.push_back(lut[c & 15]);
    }
}
```

```
}
return output;
}

int main()
{
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
string SECRET_ID = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
string SECRET_KEY = getenv("TENCENTCLOUD_SECRET_KEY");

string service = "cvm";
string host = "cvm.tencentcloudapi.com";
string region = "ap-guangzhou";
string action = "DescribeInstances";
string version = "2017-03-12";
int64_t timestamp = 1551113065;
string date = get_data(timestamp);

// ***** 步骤 1: 拼接规范请求串 *****
string httpRequestMethod = "POST";
string canonicalUri = "/";
string canonicalQueryString = "";
string lower = action;
std::transform(action.begin(), action.end(), lower.begin(), ::tolower);
string canonicalHeaders = string("content-type:application/json; charset=utf-8\n")
+ "host:" + host + "\n"
+ "x-tc-action:" + lower + "\n";
string signedHeaders = "content-type;host;x-tc-action";
string payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}] }";
string hashedRequestPayload = sha256Hex(payload);
string canonicalRequest = httpRequestMethod + "\n"
+ canonicalUri + "\n"
+ canonicalQueryString + "\n"
+ canonicalHeaders + "\n"
+ signedHeaders + "\n"
+ hashedRequestPayload;
cout << canonicalRequest << endl;

// ***** 步骤 2: 拼接待签名字符串 *****
string algorithm = "TC3-HMAC-SHA256";
string RequestTimestamp = int2str(timestamp);
string credentialScope = date + "/" + service + "/" + "tc3_request";
string hashedCanonicalRequest = sha256Hex(canonicalRequest);
string stringToSign = algorithm + "\n" + RequestTimestamp + "\n" + credentialScope + "\n" + hashedCanonicalRequest;
cout << stringToSign << endl;

// ***** 步骤 3: 计算签名 *****
string kKey = "TC3" + SECRET_KEY;
string kDate = HmacSha256(kKey, date);
string kService = HmacSha256(kDate, service);
string kSigning = HmacSha256(kService, "tc3_request");
string signature = HexEncode(HmacSha256(kSigning, stringToSign));
cout << signature << endl;

// ***** 步骤 4: 拼接 Authorization *****
```

```
string authorization = algorithm + " " + "Credential=" + SECRET_ID + "/" + credentialScope + ", "
+ "SignedHeaders=" + signedHeaders + ", " + "Signature=" + signature;
cout << authorization << endl;

string curlcmd = "curl -X POST https://" + host + "\n"
+ " -H \"Authorization: \" + authorization + "\"\n"
+ " -H \"Content-Type: application/json; charset=utf-8\" + "\n"
+ " -H \"Host: \" + host + "\"\n"
+ " -H \"X-TC-Action: \" + action + "\"\n"
+ " -H \"X-TC-Timestamp: \" + RequestTimestamp + "\"\n"
+ " -H \"X-TC-Version: \" + version + "\"\n"
+ " -H \"X-TC-Region: \" + region + "\"\n"
+ " -d '\" + payload + '\"";
cout << curlcmd << endl;
return 0;
};
```

**C**

```
#include <ctype.h>
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
#include <stdint.h>
#include <openssl/sha.h>
#include <openssl/hmac.h>

void get_utc_date(int64_t timestamp, char* utc, int len)
{
    // time_t timenow;
    struct tm sttime;
    sttime = *gmtime(&timestamp);
    strftime(utc, len, "%Y-%m-%d", &sttime);
}

void sha256_hex(const char* str, char* result)
{
    char buf[3];
    unsigned char hash[SHA256_DIGEST_LENGTH];
    SHA256_CTX sha256;
    SHA256_Init(&sha256);
    SHA256_Update(&sha256, str, strlen(str));
    SHA256_Final(hash, &sha256);
    for(int i = 0; i < SHA256_DIGEST_LENGTH; i++)
    {
        snprintf(buf, sizeof(buf), "%02x", hash[i]);
        strcat(result, buf);
    }
}

void hmac_sha256(const char* key, int key_len,
const char* input, int input_len,
unsigned char* output, unsigned int* output_len)
{
    HMAC_CTX *h;
```



```
#if OPENSSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX hmac;
HMAC_CTX_init(&hmac);
h = &hmac;
#else
h = HMAC_CTX_new();
#endif

HMAC_Init_ex(h, key, key_len, EVP_sha256(), NULL);
HMAC_Update(h, ( unsigned char* )input, input_len);
HMAC_Final(h, output, output_len);

#if OPENSSSL_VERSION_NUMBER < 0x10100000L
HMAC_CTX_cleanup(h);
#else
HMAC_CTX_free(h);
#endif

}

void hex_encode(const char* input, int input_len, char* output)
{
    static const char* const lut = "0123456789abcdef";

    char add_out[128] = {0};
    char temp[2] = {0};
    for (size_t i = 0; i < input_len; ++i)
    {
        const unsigned char c = input[i];
        temp[0] = lut[c >> 4];
        strcat(add_out, temp);
        temp[0] = lut[c & 15];
        strcat(add_out, temp);
    }
    strncpy(output, add_out, 128);
}

void lowercase(const char * src, char * dst)
{
    for (int i = 0; src[i]; i++)
    {
        dst[i] = tolower(src[i]);
    }
}

int main()
{
    // 密钥参数
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    const char* SECRET_ID = getenv("TENCENTCLOUD_SECRET_ID");
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
    const char* SECRET_KEY = getenv("TENCENTCLOUD_SECRET_KEY");
    const char* service = "cvm";
    const char* host = "cvm.tencentcloudapi.com";
    const char* region = "ap-guangzhou";
    const char* action = "DescribeInstances";
```

```
const char* version = "2017-03-12";
int64_t timestamp = 1551113065;
char date[20] = {0};
get_utc_date(timestamp, date, sizeof(date));

// ***** 步骤 1: 拼接规范请求串 *****
const char* http_request_method = "POST";
const char* canonical_uri = "/";
const char* canonical_query_string = "";
char canonical_headers[100] = {"Content-type:application/json; charset=utf-8\nhost:"};
strcat(canonical_headers, host);
strcat(canonical_headers, "\nX-TC-Action:");
char value[100] = {0};
lowercase(action, value);
strcat(canonical_headers, value);
strcat(canonical_headers, "\n");
const char* signed_headers = "content-type;host;x-tc-action";
const char* payload = "{\"Limit\": 1, \"Filters\": [{\"Values\": [\"\\u672a\\u547d\\u540d\"], \"Name\": \"instance-name\"}] }";
char hashed_request_payload[100] = {0};
sha256_hex(payload, hashed_request_payload);

char canonical_request[256] = {0};
sprintf(canonical_request, "%s\n%s\n%s\n%s\n%s\n%s", http_request_method,
canonical_uri, canonical_query_string, canonical_headers,
signed_headers, hashed_request_payload);
printf("%s\n", canonical_request);

// ***** 步骤 2: 拼接待签名字符串 *****
const char* algorithm = "TC3-HMAC-SHA256";
char request_timestamp[16] = {0};
sprintf(request_timestamp, "%d", timestamp);
char credential_scope[64] = {0};
strcat(credential_scope, date);
sprintf(credential_scope, "%s/%s/tc3_request", date, service);
char hashed_canonical_request[100] = {0};
sha256_hex(canonical_request, hashed_canonical_request);
char string_to_sign[256] = {0};
sprintf(string_to_sign, "%s\n%s\n%s\n%s", algorithm, request_timestamp,
credential_scope, hashed_canonical_request);
printf("%s\n", string_to_sign);

// ***** 步骤 3: 计算签名 *****
char k_key[64] = {0};
sprintf(k_key, "%s", "TC3", SECRET_KEY);
unsigned char k_date[64] = {0};
unsigned int output_len = 0;
hmac_sha256(k_key, strlen(k_key), date, strlen(date), k_date, &output_len);
unsigned char k_service[64] = {0};
hmac_sha256(k_date, output_len, service, strlen(service), k_service, &output_len);
unsigned char k_signing[64] = {0};
hmac_sha256(k_service, output_len, "tc3_request", strlen("tc3_request"), k_signing, &output_len);
unsigned char k_hmac_sha_sign[64] = {0};
hmac_sha256(k_signing, output_len, string_to_sign, strlen(string_to_sign), k_hmac_sha_sign, &output_len);

char signature[128] = {0};
```

```
hex_encode(k_hmac_sha_sign, output_len, signature);
printf("%s\n", signature);

// ***** 步骤 4: 拼接 Authorization *****
char authorization[512] = {0};
sprintf(authorization, "%s Credential=%s/%s, SignedHeaders=%s, Signature=%s",
algorithm, SECRET_ID, credential_scope, signed_headers, signature);
printf("%s\n", authorization);

char curlcmd[10240] = {0};
sprintf(curlcmd, "curl -X POST https://%s\n \
-H \"Authorization: %s\"\n \
-H \"Content-Type: application/json; charset=utf-8\"\n \
-H \"Host: %s\"\n \
-H \"X-TC-Action: %s\"\n \
-H \"X-TC-Timestamp: %s\"\n \
-H \"X-TC-Version: %s\"\n \
-H \"X-TC-Region: %s\"\n \
-d '%s'",
host, authorization, host, action, request_timestamp, version, region, payload);
printf("%s\n", curlcmd);
return 0;
}
```

## 其他语言

- Lua: [GitHub](#)
- Swift: [GitHub](#)
- Dart: [GitHub](#)
- Shell(Bash): [GitHub](#)

## 签名失败

存在以下签名失败的错误码，请根据实际情况处理。

| 错误码                          | 错误描述                                                     |
|------------------------------|----------------------------------------------------------|
| AuthFailure.SignatureExpire  | 签名过期。Timestamp 与服务器接收到请求的时间相差不得超过五分钟。                    |
| AuthFailure.SecretIdNotFound | 密钥不存在。请到控制台查看密钥是否被禁用，是否少复制了字符或者多了字符。                     |
| AuthFailure.SignatureFailure | 签名错误。可能是签名计算错误，或者签名与实际发送的内容不相符合，也有可能是密钥 SecretKey 错误导致的。 |
| AuthFailure.TokenFailure     | 临时证书 Token 错误。                                           |
| AuthFailure.InvalidSecretId  | 密钥非法（不是云 API 密钥类型）。                                      |

# 签名方法

最近更新时间：2024-12-25 01:51:19

签名方法 v1 简单易用，但是功能和安全性都不如签名方法 v3，推荐使用签名方法 v3。

首次接触，建议使用 [API Explorer](#) 中的“签名串生成”功能，选择签名版本为“API 3.0 签名 v1”，可以生成签名过程进行验证，并提供了部分编程语言的签名示例，也可直接生成 SDK 代码。推荐使用腾讯云 API 配套的 8 种常见的编程语言 SDK，已经封装了签名和请求过程，均已开源，支持 [Python](#)、[Java](#)、[PHP](#)、[Go](#)、[NodeJS](#)、[.NET](#)、[C++](#)、[Ruby](#)。

推荐使用 API Explorer

</> 点击调试

您可以通过 API Explorer 的【签名串生成】模块查看每个接口签名的生成过程。

腾讯云 API 会对每个访问请求进行身份验证，即每个请求都需要在公共请求参数中包含签名信息（Signature）以验证请求者身份。  
签名信息由安全凭证生成，安全凭证包括 SecretId 和 SecretKey；若用户还没有安全凭证，请前往 [云API密钥页面](#) 申请，否则无法调用云 API 接口。

## 1. 申请安全凭证

在第一次使用云 API 之前，请前往 [云 API 密钥页面](#) 申请安全凭证。  
安全凭证包括 SecretId 和 SecretKey：

- SecretId 用于标识 API 调用者身份
- SecretKey 用于加密签名字符串和服务器端验证签名字符串的密钥。
- 用户必须严格保管安全凭证，避免泄露。

申请安全凭证的具体步骤如下：

- 登录 [腾讯云管理中心控制台](#)。
- 前往 [云 API 密钥](#) 的控制台页面
- 在 [云 API 密钥](#) 页面，单击【新建密钥】即可以创建一对 SecretId/SecretKey。

注意：每个账号最多可以拥有两对 SecretId/SecretKey。

## 2. 生成签名串

有了安全凭证 SecretId 和 SecretKey 后，就可以生成签名串了。以下是使用签名方法 v1 生成签名串的详细过程：

假设用户的 SecretId 和 SecretKey 分别是：

- SecretId: AKID\*\*\*\*\*
- SecretKey: \*\*\*\*\*

注意：这里只是示例，请根据用户实际申请的 SecretId 和 SecretKey 进行后续操作！

以云服务器查看实例列表（DescribeInstances）请求为例，当用户调用这一接口时，其请求参数可能如下：

| 参数名称          | 中文        | 参数值               |
|---------------|-----------|-------------------|
| Action        | 方法名       | DescribeInstances |
| SecretId      | 密钥 ID     | AKID*****         |
| Timestamp     | 当前时间戳     | 1465185768        |
| Nonce         | 随机正整数     | 11886             |
| Region        | 实例所在区域    | ap-guangzhou      |
| InstanceIds.0 | 待查询的实例 ID | ins-09dx96dg      |
| Offset        | 偏移量       | 0                 |
| Limit         | 最大允许输出    | 20                |
| Version       | 接口版本号     | 2017-03-12        |

这里只展示了部分公共参数和接口输入参数，用户可以根据实际需要添加其他参数，例如 Language 和 Token 公共参数。

## 2.1. 对参数排序

首先对所有请求参数按参数名的字典序（ASCII 码）升序排序。注意：1）只按参数名进行排序，参数值保持对应即可，不参与比大小；2）按 ASCII 码比大小，如 InstanceIds.2 要排在 InstanceIds.12 后面，不是按字母表，也不是按数值。用户可以借助编程语言中的相关排序函数来实现这一功能，如 PHP 中的 ksort 函数。上述示例参数的排序结果如下：

```
{
  'Action' : 'DescribeInstances',
  'InstanceIds.0' : 'ins-09dx96dg',
  'Limit' : 20,
  'Nonce' : 11886,
  'Offset' : 0,
  'Region' : 'ap-guangzhou',
  'SecretId' : 'AKID*****',
  'Timestamp' : 1465185768,
  'Version': '2017-03-12',
}
```

使用其它程序设计语言开发时，可对上面示例中的参数进行排序，得到的结果一致即可。

## 2.2. 拼接请求字符串

此步骤生成请求字符串。

将把上一步排序好的请求参数格式化成“参数名称=参数值”的形式，如对 Action 参数，其参数名称为 "Action"，参数值为 "DescribeInstances"，因此格式化后即为 Action=DescribeInstances。

**注意：“参数值”为原始值而非 url 编码后的值。**

然后将格式化后的各个参数用"&"拼接在一起，最终生成的请求字符串为：

```
Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-guangzhou&SecretId=AKID*****
&Timestamp=1465185768&Version=2017-03-12
```

## 2.3. 拼接签名原文字符串

此步骤生成签名原文字符串。

签名原文字符串由以下几个参数构成：

1. 请求方法: 支持 POST 和 GET 方式，这里使用 GET 请求，注意方法为全大写。
2. 请求主机: 查看实例列表(DescribeInstances)的请求域名为：cvm.tencentcloudapi.com。实际的请求域名根据接口所属模块的不同而不同，详见各接口说明。
3. 请求路径: 当前版本云API的请求路径固定为 /。
4. 请求字符串: 即上一步生成的请求字符串。

签名原串的拼接规则为：请求方法 + 请求主机 + 请求路径 + ? + 请求字符串。

示例的拼接结果为：

```
GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-gu
angzhou&SecretId=AKID*****&Timestamp=1465185768&Version=2017-03-12
```

## 2.4. 生成签名串

此步骤生成签名串。

首先使用 HMAC-SHA1 算法对上一步中获得的签名原文字符串进行签名，然后将生成的签名串使用 Base64 进行编码，即可获得最终的签名串。

具体代码如下，以 PHP 语言为例：

```
$secretKey = '*****';
$srcStr = 'GETcvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&R
egion=ap-guangzhou&SecretId=AKID*****&Timestamp=1465185768&Version=2017-03-12';
```

```
$signStr = base64_encode(hash_hmac('sha1', $srcStr, $secretKey, true));  
echo $signStr;
```

最终得到的签名串为：

```
7RAM2xfNMO9EiVTNmPg06MRnCvQ=
```

使用其它程序设计语言开发时，可用上面示例中的原文进行签名验证，得到的签名串与例子中的一致即可。

### 3. 签名串编码

生成的签名串并不能直接作为请求参数，需要对其进行 URL 编码。

如上一步生成的签名串为 7RAM2xfNMO9EiVTNmPg06MRnCvQ=，最终得到的签名串请求参数（Signature）为：

7RAM2xfNMO9EiVTNmPg06MRnCvQ%3D，它将用于生成最终的请求 URL。

**注意：**如果用户的请求方法是 GET，或者请求方法为 POST 同时 Content-Type 为 application/x-www-form-urlencoded，则发送请求时所有请求参数的值均需要做 URL 编码，参数键和=符号不需要编码。非 ASCII 字符在 URL 编码前需要先以 UTF-8 进行编码。

**注意：**有些编程语言的网络库会自动为所有参数进行 urlencode，在这种情况下，就不需要对签名串进行 URL 编码了，否则两次 URL 编码会导致签名失败。

**注意：**其他参数值也需要进行编码，编码采用 [RFC 3986](#)。使用 %XY 对特殊字符例如汉字进行百分比编码，其中“X”和“Y”为十六进制字符（0-9 和大写字母 A-F），使用小写将引发错误。

### 4. 签名失败

根据实际情况，存在以下签名失败的错误码，请根据实际情况处理。

| 错误代码                         | 错误描述               |
|------------------------------|--------------------|
| AuthFailure.SignatureExpire  | 签名过期               |
| AuthFailure.SecretIdNotFound | 密钥不存在              |
| AuthFailure.SignatureFailure | 签名错误               |
| AuthFailure.TokenFailure     | token 错误           |
| AuthFailure.InvalidSecretId  | 密钥非法（不是云 API 密钥类型） |

### 5. 签名演示

在实际调用 API 3.0 时，推荐使用配套的腾讯云 SDK 3.0，SDK 封装了签名的过程，开发时只关注产品提供的具体接口即可。详细信息参见 [SDK 中心](#)。当前支持的编程语言有：

- [Python](#)
- [Java](#)
- [PHP](#)
- [Go](#)
- [NodeJS](#)
- [.NET](#)
- [C++](#)
- [Ruby](#)

下面提供了不同产品的生成签名 demo，您可以找到对应的产品参考签名的生成：

- [Signature Demo](#)

为了更清楚的解释签名过程，下面以实际编程语言为例，将上述的签名过程具体实现。请求的域名、调用的接口和参数的取值都以上述签名过程为准，代码只为解释签名过程，并不具备通用性，实际开发请尽量使用 SDK。

最终输出的 url 可能为：`https://cvm.tencentcloudapi.com/?Action=DescribeInstances&InstanceIds.0=ins-09dx96dg&Limit=20&Nonce=11886&Offset=0&Region=ap-`

guangzhou&SecretId=AKID\*\*\*\*\*&Signature=7RAM2xfNMO9EiVTNmPg06MRnCvQ%3D&Timestamp=1465185768&Version=2017-03-12。

注意：由于示例中的密钥是虚构的，时间戳也不是系统当前时间，因此如果将此 url 在浏览器中打开或者用 curl 等命令调用时会返回鉴权错误：签名过期。为了得到一个可以正常返回的 url，需要修改示例中的 SecretId 和 SecretKey 为真实的密钥，并使用系统当前时间戳作为 Timestamp。

注意：在下面的示例中，不同编程语言，甚至同一语言每次执行得到的 url 可能都有所不同，表现为参数的顺序不同，但这并不影响正确性。只要所有参数都在，且签名计算正确即可。

注意：以下代码仅适用于 API 3.0，不能直接用于其他的签名流程，请以对应的实际文档为准。

## Java

```
import java.io.UnsupportedEncodingException;
import java.net.URLEncoder;
import java.util.Random;
import java.util.TreeMap;
import javax.crypto.Mac;
import javax.crypto.spec.SecretKeySpec;
import javax.xml.bind.DatatypeConverter;

public class TencentCloudAPIDemo {
    private final static String CHARSET = "UTF-8";

    public static String sign(String s, String key, String method) throws Exception {
        Mac mac = Mac.getInstance(method);
        SecretKeySpec secretKeySpec = new SecretKeySpec(key.getBytes(CHARSET), mac.getAlgorithm());
        mac.init(secretKeySpec);
        byte[] hash = mac.doFinal(s.getBytes(CHARSET));
        return DatatypeConverter.printBase64Binary(hash);
    }

    public static String getStringToSign(TreeMap<String, Object> params) {
        StringBuilder s2s = new StringBuilder("GETcvm.tencentcloudapi.com/?");
        // 签名时要求对参数进行字典排序，此处用TreeMap保证顺序
        for (String k : params.keySet()) {
            s2s.append(k).append("=").append(params.get(k).toString()).append("&");
        }
        return s2s.toString().substring(0, s2s.length() - 1);
    }

    public static String getUrl(TreeMap<String, Object> params) throws UnsupportedEncodingException {
        StringBuilder url = new StringBuilder("https://cvm.tencentcloudapi.com/?");
        // 实际请求的url中对参数顺序没有要求
        for (String k : params.keySet()) {
            // 需要对请求串进行urlencode，由于key都是英文字母，故此处仅对其value进行urlencode
            url.append(k).append("=").append(URLEncoder.encode(params.get(k).toString(), CHARSET)).append("&");
        }
        return url.toString().substring(0, url.length() - 1);
    }

    public static void main(String[] args) throws Exception {
        TreeMap<String, Object> params = new TreeMap<String, Object>(); // TreeMap可以自动排序
        // 实际调用时应当使用随机数，例如：params.put("Nonce", new Random().nextInt(java.lang.Integer.MAX_VALUE));
        params.put("Nonce", 11886); // 公共参数
        // 实际调用时应当使用系统当前时间，例如：params.put("Timestamp", System.currentTimeMillis() / 1000);
        params.put("Timestamp", 1465185768); // 公共参数
        // 需要设置环境变量 TENCENTCLOUD_SECRET_ID，值为示例的 AKID*****
        params.put("SecretId", System.getenv("TENCENTCLOUD_SECRET_ID")); // 公共参数
    }
}
```

```
params.put("Action", "DescribeInstances"); // 公共参数
params.put("Version", "2017-03-12"); // 公共参数
params.put("Region", "ap-guangzhou"); // 公共参数
params.put("Limit", 20); // 业务参数
params.put("Offset", 0); // 业务参数
params.put("InstanceIds.0", "ins-09dx96dg"); // 业务参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
params.put("Signature", sign(getStringToSign(params), System.getenv("TENCENTCLOUD_SECRET_KEY"), "HmacSHA1")); // 公共参数
System.out.println(getUrl(params));
}
}
```

## Python

注意：如果是在 Python 2 环境中运行，需要先安装 requests 依赖包： `pip install requests` 。

```
# -*- coding: utf8 -*-
import base64
import hashlib
import hmac
import os
import time

import requests

# 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
secret_id = os.environ.get("TENCENTCLOUD_SECRET_ID")
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
secret_key = os.environ.get("TENCENTCLOUD_SECRET_KEY")

def get_string_to_sign(method, endpoint, params):
    s = method + endpoint + "?"
    query_str = "&".join("%s=%s" % (k, params[k]) for k in sorted(params))
    return s + query_str

def sign_str(key, s, method):
    hmac_str = hmac.new(key.encode("utf8"), s.encode("utf8"), method).digest()
    return base64.b64encode(hmac_str)

if __name__ == '__main__':
    endpoint = "cvm.tencentcloudapi.com"
    data = {
        'Action': 'DescribeInstances',
        'InstanceIds.0': 'ins-09dx96dg',
        'Limit': 20,
        'Nonce': 11886,
        'Offset': 0,
        'Region': 'ap-guangzhou',
        'SecretId': secret_id,
        'Timestamp': 1465185768, # int(time.time())
        'Version': '2017-03-12'
    }
    s = get_string_to_sign("GET", endpoint, data)
    data["Signature"] = sign_str(secret_key, s, hashlib.sha1)
    print(data["Signature"])
    # 此处会实际调用，成功后可能产生计费
```



```
# resp = requests.get("https://" + endpoint, params=data)
# print(resp.url)
```

## Golang

```
package main

import (
    "bytes"
    "crypto/hmac"
    "crypto/sha1"
    "encoding/base64"
    "fmt"
    "os"
    "sort"
    "strconv"
)

func main() {
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    secretId := os.Getenv("TENCENTCLOUD_SECRET_ID")
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
    secretKey := os.Getenv("TENCENTCLOUD_SECRET_KEY")
    params := map[string]string{
        "Nonce": "11886",
        "Timestamp": strconv.Itoa(1465185768),
        "Region": "ap-guangzhou",
        "SecretId": secretId,
        "Version": "2017-03-12",
        "Action": "DescribeInstances",
        "InstanceIds.0": "ins-09dx96dg",
        "Limit": strconv.Itoa(20),
        "Offset": strconv.Itoa(0),
    }

    var buf bytes.Buffer
    buf.WriteString("GET")
    buf.WriteString("cvm.tencentcloudapi.com")
    buf.WriteString("/")
    buf.WriteString("?")

    // sort keys by ascii asc order
    keys := make([]string, 0, len(params))
    for k, _ := range params {
        keys = append(keys, k)
    }
    sort.Strings(keys)

    for i := range keys {
        k := keys[i]
        buf.WriteString(k)
        buf.WriteString("=")
        buf.WriteString(params[k])
        buf.WriteString("&")
    }
    buf.Truncate(buf.Len() - 1)
```

```
hashed := hmac.New(sha1.New, []byte(secretKey))
hashed.Write(buf.Bytes())

fmt.Println(base64.StdEncoding.EncodeToString(hashed.Sum(nil)))
}
```

## PHP

```
<?php
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
$secretId = getenv("TENCENTCLOUD_SECRET_ID");
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
$secretKey = getenv("TENCENTCLOUD_SECRET_KEY");
$params["Nonce"] = 11886;//rand();
$params["Timestamp"] = 1465185768;//time();
$params["Region"] = "ap-guangzhou";
$params["SecretId"] = $secretId;
$params["Version"] = "2017-03-12";
$params["Action"] = "DescribeInstances";
$params["InstanceIds.0"] = "ins-09dx96dg";
$params["Limit"] = 20;
$params["Offset"] = 0;

ksort($params);

$signStr = "GETcvm.tencentcloudapi.com/?";
foreach ( $params as $key => $value ) {
    $signStr = $signStr . $key . "=" . $value . "&";
}
$signStr = substr($signStr, 0, -1);

$signature = base64_encode(hash_hmac("sha1", $signStr, $secretKey, true));
echo $signature.PHP_EOL;
// need to install and enable curl extension in php.ini
// $params["Signature"] = $signature;
// $url = "https://cvm.tencentcloudapi.com/?".http_build_query($params);
// echo $url.PHP_EOL;
// $ch = curl_init();
// curl_setopt($ch, CURLOPT_URL, $url);
// $output = curl_exec($ch);
// curl_close($ch);
// echo json_decode($output);
```

## Ruby

```
# -*- coding: UTF-8 -*-
# require ruby>=2.3.0
require 'time'
require 'openssl'
require 'base64'

# 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
secret_id = ENV["TENCENTCLOUD_SECRET_ID"]
# 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
```

```
secret_key = ENV["TENCENTCLOUD_SECRET_KEY"]

method = 'GET'
endpoint = 'cvm.tencentcloudapi.com'
data = {
  'Action' => 'DescribeInstances',
  'InstanceIds.0' => 'ins-09dx96dg',
  'Limit' => 20,
  'Nonce' => 11886,
  'Offset' => 0,
  'Region' => 'ap-guangzhou',
  'SecretId' => secret_id,
  'Timestamp' => 1465185768, # Time.now.to_i
  'Version' => '2017-03-12',
}
sign = method + endpoint + '/?'
params = []
data.sort.each do |item|
  params << "#{item[0]}=#{item[1]}"
end
sign += params.join('&')
digest = OpenSSL::Digest.new('sha1')
data['Signature'] = Base64.encode64(OpenSSL::HMAC.digest(digest, secret_key, sign))
puts data['Signature']

# require 'net/http'
# uri = URI('https://' + endpoint)
# uri.query = URI.encode_www_form(data)
# p uri
# res = Net::HTTP.get_response(uri)
# puts res.body
```

## DotNet

```
using System;
using System.Collections.Generic;
using System.Net;
using System.Security.Cryptography;
using System.Text;

public class Application {
  public static string Sign(string signKey, string secret)
  {
    string signRet = string.Empty;
    using (HMACSHA1 mac = new HMACSHA1(Encoding.UTF8.GetBytes(signKey)))
    {
      byte[] hash = mac.ComputeHash(Encoding.UTF8.GetBytes(secret));
      signRet = Convert.ToBase64String(hash);
    }
    return signRet;
  }

  public static string MakeSignPlainText(SortedDictionary<string, string> requestParams, string requestMethod, string requestHost, string requestPath)
  {
    string retStr = "";
    retStr += requestMethod;
```

```
retStr += requestHost;
retStr += requestPath;
retStr += "?";
string v = "";
foreach (string key in requestParams.Keys)
{
    v += string.Format("{0}={1}&", key, requestParams[key]);
}
retStr += v.TrimEnd('&');
return retStr;
}

public static void Main(string[] args)
{
    // 密钥参数
    // 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
    string SECRET_ID = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_ID");
    // 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
    string SECRET_KEY = Environment.GetEnvironmentVariable("TENCENTCLOUD_SECRET_KEY");

    string endpoint = "cvm.tencentcloudapi.com";
    string region = "ap-guangzhou";
    string action = "DescribeInstances";
    string version = "2017-03-12";
    double RequestTimestamp = 1465185768; // 时间戳 2019-02-26 00:44:25, 此参数作为示例, 以实际为准
    // long timestamp = ToTimestamp() / 1000;
    // string requestTimestamp = timestamp.ToString();
    Dictionary<string, string> param = new Dictionary<string, string>();
    param.Add("Limit", "20");
    param.Add("Offset", "0");
    param.Add("InstanceIds.0", "ins-09dx96dg");
    param.Add("Action", action);
    param.Add("Nonce", "11886");
    // param.Add("Nonce", Math.Abs(new Random().Next()).ToString());

    param.Add("Timestamp", RequestTimestamp.ToString());
    param.Add("Version", version);

    param.Add("SecretId", SECRET_ID);
    param.Add("Region", region);
    SortedDictionary<string, string> headers = new SortedDictionary<string, string>(param, StringComparer.Ordinal);
    string sigInParam = MakeSignPlainText(headers, "GET", endpoint, "/");
    string sigOutParam = Sign(SECRET_KEY, sigInParam);
    Console.WriteLine(sigOutParam);
}
}
```

## NodeJS

```
const crypto = require('crypto');

function get_req_url(params, endpoint){
    params['Signature'] = encodeURIComponent(params['Signature']);
    const url_strParam = sort_params(params)
    return "https://" + endpoint + "/" + "?" + url_strParam.slice(1);
}
```

```
function formatSignString(reqMethod, endpoint, path, strParam){
let strSign = reqMethod + endpoint + path + "?" + strParam.slice(1);
return strSign;
}

function sha1(secretKey, strsign){
let signMethodMap = {'HmacSHA1': "sha1"};
let hmac = crypto.createHmac(signMethodMap['HmacSHA1'], secretKey || "");
return hmac.update(Buffer.from(strsign, 'utf8')).digest('base64')
}

function sort_params(params){
let strParam = "";
let keys = Object.keys(params);
keys.sort();
for (let k in keys) {
//k = k.replace(/_/g, '.');
strParam += ("%&" + keys[k] + "=" + params[keys[k]]);
}
return strParam
}

function main(){
// 密钥参数
// 需要设置环境变量 TENCENTCLOUD_SECRET_ID, 值为示例的 AKID*****
const SECRET_ID = process.env.TENCENTCLOUD_SECRET_ID
// 需要设置环境变量 TENCENTCLOUD_SECRET_KEY, 值为示例的 *****
const SECRET_KEY = process.env.TENCENTCLOUD_SECRET_KEY

const endpoint = "cvm.tencentcloudapi.com"
const Region = "ap-guangzhou"
const Version = "2017-03-12"
const Action = "DescribeInstances"
const Timestamp = 1465185768 // 时间戳 2016-06-06 12:02:48, 此参数作为示例, 以实际为准
// const Timestamp = Math.round(Date.now() / 1000)
const Nonce = 11886 // 随机正整数
//const nonce = Math.round(Math.random() * 65535)

let params = {};
params['Action'] = Action;
params['InstanceIds.0'] = 'ins-09dx96dg';
params['Limit'] = 20;
params['Offset'] = 0;
params['Nonce'] = Nonce;
params['Region'] = Region;
params['SecretId'] = SECRET_ID;
params['Timestamp'] = Timestamp;
params['Version'] = Version;

// 1. 对参数排序, 并拼接请求字符串
strParam = sort_params(params)

// 2. 拼接签名原字符串
const reqMethod = "GET";
const path = "/";
strSign = formatSignString(reqMethod, endpoint, path, strParam)
// console.log(strSign)
```

```
// 3. 生成签名串
params['Signature'] = sha1(SECRET_KEY, strSign)
console.log(params['Signature'])

// 4. 进行url编码并拼接请求url
// const req_url = get_req_url(params, endpoint)
// console.log(params['Signature'])
// console.log(req_url)
}
main()
```

## 返回结果

最近更新时间：2024-03-12 01:42:28

云 API 3.0 接口默认返回 JSON 数据，返回非 JSON 格式的接口会在文档中做出说明。返回 JSON 数据时最大限制为 50 MB，如果返回的数据超过最大限制，请求会失败并返回内部错误。请根据接口文档中给出的过滤功能（例如时间范围）或者分页功能，控制返回数据不要过大。

注意：目前只要请求被服务端正常处理了，响应的 HTTP 状态码均为 200。例如返回的消息体里的错误码是签名失败，但 HTTP 状态码是 200，而不是 401。

### 正确返回结果

以云服务器的接口查看实例状态列表 (DescribeInstancesStatus) 2017-03-12 版本为例，若调用成功，其可能的返回如下为：

```
{
  "Response": {
    "TotalCount": 0,
    "InstanceStatusSet": [],
    "RequestId": "b5b41468-520d-4192-b42f-595cc34b6c1c"
  }
}
```

- Response 及其内部的 RequestId 是固定的字段，无论请求成功与否，只要 API 处理了，则必定会返回。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系 [腾讯云客服](#) 或 [提交工单](#)，并提供该 ID 来解决问题。
- 除了固定的字段外，其余均为具体接口定义的字段，不同的接口所返回的字段参见接口文档中的定义。此例中的 TotalCount 和 InstanceStatusSet 均为 DescribeInstancesStatus 接口定义的字段，由于调用请求的用户暂时还没有云服务器实例，因此 TotalCount 在此情况下的返回值为 0，InstanceStatusSet 列表为空。

### 错误返回结果

若调用失败，其返回值示例如下为：

```
{
  "Response": {
    "Error": {
      "Code": "AuthFailure.SignatureFailure",
      "Message": "The provided credentials could not be validated. Please check your signature is correct."
    },
    "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
  }
}
```

- Error 的出现代表着该请求调用失败。Error 字段连同其内部的 Code 和 Message 字段在调用失败时是必定返回的。
- Code 表示具体出错的错误码，当请求出错时可以先根据该错误码在公共错误码和当前接口对应的错误码列表里面查找对应原因和解决方案。
- Message 显示出了这个错误发生的具体原因，随着业务发展或体验优化，此文本可能会经常保持变更或更新，用户不应依赖这个返回值。
- RequestId 用于一个 API 请求的唯一标识，如果 API 出现异常，可以联系 [腾讯云客服](#) 或 [提交工单](#)，并提供该 ID 来解决问题。

### 公共错误码

返回结果中如果存在 Error 字段，则表示调用 API 接口失败。Error 中的 Code 字段表示错误码，所有业务都可能出现的错误码为公共错误码。完整的错误码列表请参考本产品“API 文档”目录下的“错误码”页面。

## 参数类型

最近更新时间：2022-08-10 06:42:19

目前腾讯云 API 3.0 输入参数和输出参数支持如下几种数据格式：

- String: 字符串。
- Integer: 整型，上限为无符号64位整数。SDK 3.0 不同编程语言支持的类型有所差异，建议以所使用编程语言的最大整型定义，例如 Golang 的 `uint64`。
- Boolean: 布尔型。
- Float: 浮点型。
- Double: 双精度浮点型。
- Date: 字符串，日期格式。例如：2022-01-01。
- Timestamp: 字符串，时间格式。例如：2022-01-01 00:00:00。
- Timestamp ISO8601: ISO 8601 是由国际标准化组织（International Organization for Standardization，ISO）发布的关于日期和时间格式的国际标准，对应国标《GB/T 7408-2005数据元和交换格式信息交换日期和时间表示法》。建议以所使用编程语言的标准库进行格式解析。例如：2022-01-01T00:00:00+08:00。
- Binary: 二进制内容，需要以特定协议请求和解析。



# 文本生成相关接口

## 对联生成

最近更新时间：2025-03-20 12:04:45

### 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

对联生成接口根据用户输入的命题关键词，智能生成一副完整的春联，包括上联、下联和横批。该接口利用先进的自然语言处理技术，确保生成的春联既符合传统对仗、对韵、对义的要求，又具有新意和创意，为用户提供独特的春节祝福。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

### 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称       | 必选 | 类型      | 描述                                                                             |
|------------|----|---------|--------------------------------------------------------------------------------|
| Action     | 是  | String  | <a href="#">公共参数</a> ，本接口取值：ComposeCouplet。                                    |
| Version    | 是  | String  | <a href="#">公共参数</a> ，本接口取值：2019-04-08。                                        |
| Region     | 否  | String  | <a href="#">公共参数</a> ，本接口不需要传递此参数。                                             |
| Text       | 是  | String  | 生成对联的关键词。长度需>=2，当长度>2时，自动截取前两个字作为关键字。内容需为常用汉字（不含有数字、英文、韩语、日语、符号等其他）。<br>示例值：小鸡 |
| TargetType | 否  | Integer | 返回的文本结果为繁体还是简体。0：简体；1：繁体。默认为0。<br>示例值：0                                        |

### 3. 输出参数

| 参数名称        | 类型              | 描述                                                                                     |
|-------------|-----------------|----------------------------------------------------------------------------------------|
| TopScroll   | String          | 横批。<br>示例值：满园春色                                                                        |
| Content     | Array of String | 上联与下联。                                                                                 |
| RandomCause | String          | 当对联随机生成时，展示随机生成原因。                                                                     |
| RequestId   | String          | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

### 4. 示例

#### 示例1 对联生成调用示例

对联生成。

#### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ComposeCouplet
<公共请求参数>
```

```
{
  "Text": "阳光",
  "TargetType": 0
}
```

#### 输出示例

```
{
  "Response": {
    "Content": [
      "阳春白雪江山秀",
      "光彩红梅岁月长"
    ],
    "RandomCause": "",
    "RequestId": "e9cc9a63-e2e9-4f92-a419-0fd8bf25cdf6",
    "TopScroll": "盛世新春"
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                 | 描述                  |
|-------------------------------------|---------------------|
| FailedOperation.BalanceInsufficient | 余额不足，开通失败，请充值后再开通。  |
| FailedOperation.NoCouplets          | 暂无春联生成，请更换关键词重试。    |
| FailedOperation.NotFoundData        | 未查询到结果。             |
| FailedOperation.RequestTimeout      | 后端服务超时。             |
| FailedOperation.RpcFail             | RPC请求失败，一般为算法微服务故障。 |

| 错误码                                    | 描述                              |
|----------------------------------------|---------------------------------|
| FailedOperation.UnKnowError            | 内部错误。                           |
| InternalError                          | 内部错误。                           |
| InternalError.ServiceCallError         | 内部服务调用错误。                       |
| InternalError.ServiceError             | 内部服务调用失败。                       |
| InvalidParameterValue.InvalidParameter | 参数不合法。                          |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。                   |
| InvalidParameterValue.TargetType       | TargetType非法，请参考TargetType参数说明。 |
| InvalidParameterValue.Text             | Text非法，请参考Text参数说明。             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。                       |
| ResourceNotFound                       | 资源不存在。                          |
| ResourceUnavailable.Freeze             | 账号已被冻结。                         |
| ResourceUnavailable.InArrears          | 账号已欠费。                          |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。                    |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。          |
| ResourceUnavailable.Recover            | 资源已被回收。                         |
| ResourceUnavailable.StopUsing          | 账号已停服。                          |
| UnsupportedOperation.PkgExhausted      | 资源包余量已用尽。                       |

# 诗词生成

最近更新时间：2025-03-20 12:04:45

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

诗词生成接口利用现代的自然语言处理和深度学习技术，模仿了古代著名诗人的风格，为用户产生独特的诗词。用户只需输入的命题关键词，接口就能自动生成一首七言律诗或五言律诗。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称       | 必选 | 类型      | 描述                                            |
|------------|----|---------|-----------------------------------------------|
| Action     | 是  | String  | <a href="#">公共参数</a> ，本接口取值：ComposePoetry。    |
| Version    | 是  | String  | <a href="#">公共参数</a> ，本接口取值：2019-04-08。       |
| Region     | 否  | String  | <a href="#">公共参数</a> ，本接口不需要传递此参数。            |
| Text       | 是  | String  | 生成诗词的关键词。<br>示例值：阳光                           |
| PoetryType | 否  | Integer | 生成诗词的类型。0：藏头或藏身；1：藏头；2：藏身。默认为0。<br>示例值：0      |
| Genre      | 否  | Integer | 诗的体裁。0：五言律诗或七言律诗；5：五言律诗；7：七言律诗。默认为0。<br>示例值：7 |

## 3. 输出参数

| 参数名称      | 类型              | 描述                                                                                     |
|-----------|-----------------|----------------------------------------------------------------------------------------|
| Title     | String          | 诗题，即输入的生成诗词的关键词。<br>示例值：阳光                                                             |
| Content   | Array of String | 诗的内容。                                                                                  |
| RequestId | String          | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 诗词生成调用示例

诗词生成。

#### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ComposePoetry
<公共请求参数>
```

```
{
  "Text": "阳光",
  "PoetryType": 2,
  "Genre": 5
}
```

#### 输出示例

```
{
  "Response": {
    "Content": [
      "洛阳无人见",
      "江城日光流",
      "月明山上路",
      "柳暗水中洲"
    ],
    "RequestId": "a4878a27-7a5c-4cd4-b17d-6a1ff403f626",
    "Title": "阳光"
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#) [Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                 | 描述                 |
|-------------------------------------|--------------------|
| FailedOperation                     | 操作失败。              |
| FailedOperation.BalanceInsufficient | 余额不足，开通失败，请充值后再开通。 |
| FailedOperation.NoPoetry            | 暂无诗词生成，请更换关键词重试。   |
| FailedOperation.NotFoundData        | 未查询到结果。            |
| FailedOperation.RequestTimeout      | 后端服务超时。            |

| 错误码                                       | 描述                                   |
|-------------------------------------------|--------------------------------------|
| FailedOperation.RpcFail                   | RPC请求失败，一般为算法微服务故障。                  |
| FailedOperation.UnKnowError               | 内部错误。                                |
| InternalError                             | 内部错误。                                |
| InternalError.ServiceCallError            | 内部服务调用错误。                            |
| InternalError.ServiceError                | 内部服务调用失败。                            |
| InvalidParameterValue.Genre               | Genre非法，请参考Genre参数说明。                |
| InvalidParameterValue.InvalidParameter    | 参数不合法。                               |
| InvalidParameterValue.PoetryType          | PoetryType非法，请参考PoetryType参数说明。      |
| InvalidParameterValue.SensitiveText       | Text输入含有敏感信息。                        |
| LimitExceeded.ResourceReachedLimit        | 资源用量达到上限。                            |
| ResourceInsufficient.QuotaRunOut          | 额度用尽，请充值后重试                          |
| ResourceNotFound                          | 资源不存在。                               |
| ResourceUnavailable                       | 资源不可用。                               |
| ResourceUnavailable.Freeze                | 账号已被冻结。                              |
| ResourceUnavailable.InArrears             | 账号已欠费。                               |
| ResourceUnavailable.IsOpening             | 服务正在开通中，请稍等。                         |
| ResourceUnavailable.NotExist              | 计费状态未知，请确认是否已在控制台开通服务。               |
| ResourceUnavailable.Recover               | 资源已被回收。                              |
| ResourceUnavailable.ServiceNotOpenedError | 您的账号尚未开通NLP服务，请登录腾讯云NLP控制台进行服务开通后再使用 |
| ResourceUnavailable.StopUsing             | 账号已停服。                               |
| UnsupportedOperation.PkgExhausted         | 资源包余量已用尽。                            |

# 文本润色

最近更新时间：2025-03-20 12:04:44

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

运用先进的自然语言处理技术，对原始文本进行优化润色，提升文本的通顺性、表达力和语言质量。

默认接口请求频率限制：1次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称       | 必选 | 类型      | 描述                                                                                                                       |
|------------|----|---------|--------------------------------------------------------------------------------------------------------------------------|
| Action     | 是  | String  | <a href="#">公共参数</a> ，本接口取值：TextEmbellish。                                                                               |
| Version    | 是  | String  | <a href="#">公共参数</a> ，本接口取值：2019-04-08。                                                                                  |
| Region     | 否  | String  | <a href="#">公共参数</a> ，本接口不需要传递此参数。                                                                                       |
| Text       | 是  | String  | 待润色的文本。中文文本长度需<=50字符；英文文本长度需<=30个单词。<br>示例值：中国奥运健儿击败强敌，勇夺冬奥会首金                                                           |
| SourceLang | 是  | String  | 待润色文本的语言类型，支持语言如下：<br>zh：中文<br>en：英文<br>示例值：zh                                                                           |
| Number     | 是  | Integer | 返回润色结果的个数。数量需>=1且<=5。<br>( 注意实际结果可能小于指定个数 )<br>示例值：1                                                                     |
| Style      | 否  | String  | 控制润色类型，类型如下：<br>both：同时返回改写和扩写<br>expansion：扩写<br>rewriting：改写<br>m2a：从现代文改写为古文<br>a2m：从古文改写为现代文<br>默认为both。<br>示例值：both |

## 3. 输出参数

| 参数名称          | 类型                                 | 描述                                                                                     |
|---------------|------------------------------------|----------------------------------------------------------------------------------------|
| EmbellishList | Array of <a href="#">Embellish</a> | 润色结果列表。                                                                                |
| RequestId     | String                             | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 文本润色调用示例

对中文文本进行扩写。

## 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action:TextEmbellish
<公共请求参数>

{
  "Text": "中国奥运健儿击败强敌，勇夺冬奥会首金",
  "SourceLang": "zh",
  "Number": 2,
  "Style": "expansion"
}
```

## 输出示例

```
{
  "Response": {
    "EmbellishList": [
      {
        "EmbellishType": "expansion",
        "Text": "现在的中国跳水奥运健儿一路高歌猛进击败强敌，不负众望的勇夺温哥华冬奥会单板滑雪首金"
      },
      {
        "EmbellishType": "expansion",
        "Text": "年轻的中国冰雪奥运健儿一次次击败强敌，历史性的勇夺温哥华冬奥会史无前例的首金"
      }
    ],
    "RequestId": "fd345650-b62b-4949-bdd7-d774152dba9a"
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)



## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                    | 描述                     |
|----------------------------------------|------------------------|
| FailedOperation.BalanceInsufficient    | 余额不足，开通失败，请充值后再开通。     |
| FailedOperation.InnerError             | 服务内部错误，请重试。            |
| FailedOperation.NoResults              | 暂无结果，请更换文本重试。          |
| FailedOperation.NotFoundData           | 未查询到结果。                |
| FailedOperation.RequestTimeout         | 后端服务超时。                |
| FailedOperation.ResourceBusy           | 服务器繁忙，请稍后再试。           |
| FailedOperation.RpcFail                | RPC请求失败，一般为算法微服务故障。    |
| FailedOperation.Unknown                | 未知错误。                  |
| InternalError                          | 内部错误。                  |
| InternalError.ServiceCallError         | 内部服务调用错误。              |
| InternalError.ServiceError             | 内部服务调用失败。              |
| InvalidParameterValue.InvalidParameter | 参数不合法。                 |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。          |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。              |
| ResourceNotFound                       | 资源不存在。                 |
| ResourceUnavailable.Freeze             | 账号已被冻结。                |
| ResourceUnavailable.InArrears          | 账号已欠费。                 |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover            | 资源已被回收。                |
| ResourceUnavailable.StopUsing          | 账号已停服。                 |

# 文本补全

最近更新时间：2025-03-20 12:04:44

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

通过自动补全文本片段，帮助用户快速生成高质量、连贯的完整文本，提高创作效率。

默认接口请求频率限制：2次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称       | 必选 | 类型      | 描述                                                                                                                                                       |
|------------|----|---------|----------------------------------------------------------------------------------------------------------------------------------------------------------|
| Action     | 是  | String  | <a href="#">公共参数</a> ，本接口取值：TextWriting。                                                                                                                 |
| Version    | 是  | String  | <a href="#">公共参数</a> ，本接口取值：2019-04-08。                                                                                                                  |
| Region     | 否  | String  | <a href="#">公共参数</a> ，本接口不需要传递此参数。                                                                                                                       |
| Text       | 是  | String  | 待续写的句子，文本统一使用utf-8格式编码，长度不超过200字符。<br>示例值：今天天气不错                                                                                                         |
| SourceLang | 是  | String  | 待续写文本的语言类型，支持语言如下：<br>zh：中文<br>en：英文<br>示例值：zh                                                                                                           |
| Number     | 是  | Integer | 返回续写结果的个数。数量需>=1且<=5。<br>( 注意实际结果可能小于指定个数 )<br>示例值：2                                                                                                     |
| Domain     | 否  | String  | 指定续写领域，支持领域如下：<br>general：通用领域，支持中英文补全<br>academic：学术领域，仅支持英文补全<br>默认为general（通用领域）。<br>示例值：general                                                      |
| Style      | 否  | String  | 指定续写风格，支持风格如下：<br>science_fiction：科幻<br>military_history：军事<br>xuanhuan_wuxia：武侠<br>urban_officialdom：职场<br>默认为xuanhuan_wuxia（武侠）。<br>示例值：xuanhuan_wuxia |

## 3. 输出参数

| 参数名称        | 类型                               | 描述                                                                                     |
|-------------|----------------------------------|----------------------------------------------------------------------------------------|
| WritingList | Array of <a href="#">Writing</a> | 续写结果列表。                                                                                |
| RequestId   | String                           | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

## 示例1 文本续写调用示例

对中文文本进行续写。

### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action:TextWriting
<公共请求参数>

{
  "Text": "今天天气不错",
  "SourceLang": "zh",
  "Number": 2,
  "Domain": "general",
  "Style": "xuanhuan_wuxia"
}
```

### 输出示例

```
{
  "Response": {
    "RequestId": "40abc589-8603-458b-82a7-fe174e799085",
    "WritingList": [
      {
        "PrefixText": "此事还得从这说起，",
        "TargetText": "此事还得从这说起，他一路行来，不知不觉已到了洛阳城，这一日，他来到一处客栈，打听一下萧月生的消息。“师父，他来了！”江南云忽然开口，声音清脆悦耳，仿"
      },
      {
        "PrefixText": "此事还得从这说起，",
        "TargetText": "此事还得从这说起，那日在山东大会上，萧月生一眼看出，这个赵小四是武林中人，心中不由一动，觉得此人极有可能是华山派的弟子。“师父，他的轻功如何？”江南"
      }
    ]
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#) [Gitee](#)

- Tencent Cloud SDK 3.0 for C++: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                    | 描述                     |
|----------------------------------------|------------------------|
| FailedOperation.BalanceInsufficient    | 余额不足，开通失败，请充值后再开通。     |
| FailedOperation.InnerError             | 服务内部错误，请重试。            |
| FailedOperation.NoResults              | 暂无结果，请更换文本重试。          |
| FailedOperation.NotFoundData           | 未查询到结果。                |
| FailedOperation.RequestTimeout         | 后端服务超时。                |
| FailedOperation.ResourceBusy           | 服务器繁忙，请稍后再试。           |
| FailedOperation.RpcFail                | RPC请求失败，一般为算法微服务故障。    |
| FailedOperation.UnKnowError            | 内部错误。                  |
| FailedOperation.Unknown                | 未知错误。                  |
| InternalError                          | 内部错误。                  |
| InternalError.ServiceCallError         | 内部服务调用错误。              |
| InternalError.ServiceError             | 内部服务调用失败。              |
| InvalidParameterValue.InvalidParameter | 参数不合法。                 |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。          |
| InvalidParameterValue.TextNumTooMuch   | 输入文本超出数量限制             |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。              |
| ResourceInsufficient.QuotaRunOut       | 额度用尽，请充值后重试            |
| ResourceNotFound                       | 资源不存在。                 |
| ResourceUnavailable.Freeze             | 账号已被冻结。                |
| ResourceUnavailable.InArrears          | 账号已欠费。                 |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover            | 资源已被回收。                |
| ResourceUnavailable.StopUsing          | 账号已停服。                 |

# 句子生成

最近更新时间：2025-03-20 12:04:45

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

根据提供的关键词，生成简洁明了的关键句子，便于用户快速获取核心观点。

默认接口请求频率限制：1次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称       | 必选 | 类型              | 描述                                                                                  |
|------------|----|-----------------|-------------------------------------------------------------------------------------|
| Action     | 是  | String          | <a href="#">公共参数</a> ，本接口取值：GenerateKeywordSentence。                                |
| Version    | 是  | String          | <a href="#">公共参数</a> ，本接口取值：2019-04-08。                                             |
| Region     | 否  | String          | <a href="#">公共参数</a> ，本接口不需要传递此参数。                                                  |
| WordList.N | 是  | Array of String | 生成句子的关键词，关键词个数需不超过4个，中文关键词长度应不超过10字符，英文关键词长度不超过3个单词。关键词中不可包含标点符号。                   |
| Number     | 是  | Integer         | 返回生成句子的个数。数量需>=1且<=5。（注意实际结果可能小于指定个数）                                               |
| Domain     | 否  | String          | 指定生成句子的领域，支持领域如下：<br>general：通用领域，支持中英文<br>academic：学术领域，仅支持英文<br>默认为general（通用领域）。 |

## 3. 输出参数

| 参数名称                | 类型                                       | 描述                                                                                     |
|---------------------|------------------------------------------|----------------------------------------------------------------------------------------|
| KeywordSentenceList | Array of <a href="#">KeywordSentence</a> | 生成的句子列表。                                                                               |
| RequestId           | String                                   | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 关键词句子生成调用示例

通过输入两个中文关键词生成句子。

#### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action:GenerateKeywordSentence
<公共请求参数>

{
```

```
"WordList": [
  "老鼠",
  "月光"
],
"Number": 2,
"Domain": "general"
}
```

#### 输出示例

```
{
  "Response": {
    "KeywordSentenceList": [
      {
        "TargetText": "12月下旬，老鼠渐渐苏醒，月光温柔，生肖鼠喜从天降。"
      },
      {
        "TargetText": "在这里，老鼠看到了月亮，这就是月光。"
      }
    ],
    "RequestId": "033daa29-dd1a-4f18-aa3b-b7a6b4d4dada"
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#) [Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                 | 描述                 |
|-------------------------------------|--------------------|
| FailedOperation.BalanceInsufficient | 余额不足，开通失败，请充值后再开通。 |
| FailedOperation.InnerError          | 服务内部错误，请重试。        |
| FailedOperation.NoResults           | 暂无结果，请更换文本重试。      |

| 错误码                                    | 描述                     |
|----------------------------------------|------------------------|
| FailedOperation.NotFoundData           | 未查询到结果。                |
| FailedOperation.RequestTimeout         | 后端服务超时。                |
| FailedOperation.ResourceBusy           | 服务器繁忙，请稍后再试。           |
| FailedOperation.RpcFail                | RPC请求失败，一般为算法微服务故障。    |
| FailedOperation.UnKnowError            | 内部错误。                  |
| FailedOperation.Unknown                | 未知错误。                  |
| InternalError                          | 内部错误。                  |
| InternalError.ServiceCallError         | 内部服务调用错误。              |
| InternalError.ServiceError             | 内部服务调用失败。              |
| InvalidParameter.ServiceError          | 服务调用失败。                |
| InvalidParameterValue.InvalidParameter | 参数不合法。                 |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。          |
| InvalidParameterValue.TextNumTooMuch   | 输入文本超出数量限制             |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。              |
| RequestLimitExceeded.UinLimitExceeded  | 请求的次数超过了频率限制。          |
| ResourceNotFound                       | 资源不存在。                 |
| ResourceUnavailable.Freeze             | 账号已被冻结。                |
| ResourceUnavailable.InArrears          | 账号已欠费。                 |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover            | 资源已被回收。                |
| ResourceUnavailable.StopUsing          | 账号已停服。                 |

# 句子纠错

最近更新时间：2025-03-20 12:04:45

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

智能识别并纠正句子中的语法、拼写、用词等错误，确保文本的准确性和可读性。

默认接口请求频率限制：5次/秒。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称       | 必选 | 类型              | 描述                                                                                                            |
|------------|----|-----------------|---------------------------------------------------------------------------------------------------------------|
| Action     | 是  | String          | <a href="#">公共参数</a> ，本接口取值：SentenceCorrection。                                                               |
| Version    | 是  | String          | <a href="#">公共参数</a> ，本接口取值：2019-04-08。                                                                       |
| Region     | 否  | String          | <a href="#">公共参数</a> ，本接口不需要传递此参数。                                                                            |
| TextList.N | 是  | Array of String | 待纠错的句子列表。可以以数组方式在一次请求中填写多个待纠错的句子。文本统一使用utf-8格式编码，每个中文句子的长度不超过150字符，每个英文句子的长度不超过100个单词，且数组长度需小于30，即句子总数需少于30句。 |

## 3. 输出参数

| 参数名称           | 类型                                      | 描述                                                                                     |
|----------------|-----------------------------------------|----------------------------------------------------------------------------------------|
| CorrectionList | Array of <a href="#">CorrectionItem</a> | 纠错结果列表。<br>( 注意仅展示错误句子的纠错结果，若句子无错则不展示，若全部待纠错句子都被认为无错，则可能返回数组为空 )                       |
| RequestId      | String                                  | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 句子纠错调用示例

对包含错别字的中文句子以及包含语法错误的英文句子进行纠错。

#### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: SentenceCorrection
<公共请求参数>

{
  "TextList": [
    "这位外国网友说：这是地球最安全的锅加，在这里，我从未感到过害啦！",
    "We always rent a little cottages from a sheep farmer and now we know his family."
  ]
}
```



## 输出示例

```
{
  "Response": {
    "CorrectionList": [
      {
        "BeginOffset": 16,
        "Confidence": 0,
        "CorrectWord": [
          "国家"
        ],
        "CorrectionType": 0,
        "DescriptionEn": null,
        "DescriptionZh": null,
        "Len": 2,
        "Order": 1,
        "Word": "锅加"
      },
      {
        "BeginOffset": 30,
        "Confidence": 0,
        "CorrectWord": [
          "怕"
        ],
        "CorrectionType": 0,
        "DescriptionEn": null,
        "DescriptionZh": null,
        "Len": 1,
        "Order": 1,
        "Word": "啪"
      },
      {
        "BeginOffset": 24,
        "Confidence": 0,
        "CorrectWord": [
          "cottage"
        ],
        "CorrectionType": 0,
        "DescriptionEn": "It appears that the noun \"cottage\" doesn't agree with \"a little\".",
        "DescriptionZh": "在当前语境下，“cottages”作为名词和量词“a little”没有保持单复数一致，建议使用“cottage”以保持名词一致性。",
        "Len": 8,
        "Order": 2,
        "Word": "cottages"
      }
    ],
    "RequestId": "b183f9d5-b35a-4880-8a36-86883b9eea11"
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

## SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#) [Gitee](#)

## 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                    | 描述                     |
|----------------------------------------|------------------------|
| FailedOperation.BalanceInsufficient    | 余额不足，开通失败，请充值后再开通。     |
| FailedOperation.InnerError             | 服务内部错误，请重试。            |
| FailedOperation.NoResults              | 暂无结果，请更换文本重试。          |
| FailedOperation.NotFoundData           | 未查询到结果。                |
| FailedOperation.RequestTimeout         | 后端服务超时。                |
| FailedOperation.ResourceBusy           | 服务器繁忙，请稍后再试。           |
| FailedOperation.RpcFail                | RPC请求失败，一般为算法微服务故障。    |
| FailedOperation.UnKnowError            | 内部错误。                  |
| FailedOperation.Unknown                | 未知错误。                  |
| InternalError                          | 内部错误。                  |
| InternalError.ServiceCallError         | 内部服务调用错误。              |
| InternalError.ServiceError             | 内部服务调用失败。              |
| InvalidParameter.TextTooLongCode       | 文本长度超过限制。              |
| InvalidParameterValue.InvalidParameter | 参数不合法。                 |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。          |
| InvalidParameterValue.TextNumTooMuch   | 输入文本超出数量限制             |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。              |
| RequestLimitExceeded.UinLimitExceeded  | 请求的次数超过了频率限制。          |
| ResourceNotFound                       | 资源不存在。                 |
| ResourceUnavailable.Freeze             | 账号已被冻结。                |
| ResourceUnavailable.InArrears          | 账号已欠费。                 |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。 |

| 错误码                               | 描述        |
|-----------------------------------|-----------|
| ResourceUnavailable.Recover       | 资源已被回收。   |
| ResourceUnavailable.StopUsing     | 账号已停服。    |
| UnsupportedOperation.PkgExhausted | 资源包余量已用尽。 |

## 文本处理相关接口

# 文本分类V2

最近更新时间：2025-03-20 12:04:47

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

文本分类接口能够对用户输入的文章进行自动分类，将其映射到具体的类目上，用户只需要提供待分类的文本，而无需关注具体实现。该功能定义了一套较为完备的[三级分类体系](#)，积累了数百万的语料，经过多轮迭代优化打造了较先进的深度学习模型，以保证效果不断提升。

推荐使用 API Explorer

<> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称      | 必选 | 类型              | 描述                                                 |
|-----------|----|-----------------|----------------------------------------------------|
| Action    | 是  | String          | <a href="#">公共参数</a> ，本接口取值：ClassifyContent。       |
| Version   | 是  | String          | <a href="#">公共参数</a> ，本接口取值：2019-04-08。            |
| Region    | 否  | String          | <a href="#">公共参数</a> ，本接口不需要传递此参数。                 |
| Title     | 是  | String          | 待分类的文章的标题（仅支持UTF-8格式，不超过100字符）。                    |
| Content.N | 是  | Array of String | 待分类文章的内容，每个元素对应一个段落。（仅支持UTF-8格式，文章内容长度总和不超过2000字符） |

## 3. 输出参数

| 参数名称                 | 类型                       | 描述                                                                                     |
|----------------------|--------------------------|----------------------------------------------------------------------------------------|
| FirstClassification  | <a href="#">Category</a> | 一级分类。（请参见附录 <a href="#">三级分类体系表</a> ）                                                  |
| SecondClassification | <a href="#">Category</a> | 二级分类。（请参见附录 <a href="#">三级分类体系表</a> ）                                                  |
| ThirdClassification  | <a href="#">Category</a> | 三级分类。（请参见附录 <a href="#">三级分类体系表</a> ）<br>注意：此字段可能返回 null，表示取不到有效值。                     |
| RequestId            | String                   | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 文本分类调用示例

文本分类。

#### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ClassifyContent
<公共请求参数>

{
  "Title": "巴萨队内危机重重，曝梅西跟董事会已决裂，巴托梅乌留他目的不纯",
  "Content": [
    "甘伯杯比赛，巴萨1-0击败艾尔切拿到了冠军，这是球队在甘伯杯上第43次拿下冠军，也是实现了8连冠的伟业，这支巴萨用一个冠军开启了新赛季的征途，本赛季能
```

```
否从皇马手中夺得西甲冠军，非常让人期待。",
"目前这支巴萨，虽然用冠军拉开了新赛季的帷幕，但谁都知道，球队问题依然很大，内部目前是危机重重，第一个是转会市场动作缓慢，第二个是球队更新换代几乎为零，第三个是新帅更换，战术体系重新调整，第四个就是高层面临调整，第五个就是梅西的心情问题。",
"这5个问题里面，最后一个尤其重要，梅西的状态决定着这支巴萨的上限和未来。大家都知道，梅西这个夏天闹出离队，折腾了11天，最终是在合同约定下留在了巴萨，但回来后的他，在训练场及热身赛里，基本是没有笑容。",
"没有人能知道梅西目前的心情，他似乎是不开心的，而一个不开心的梅西，对于巴萨来说，等于是没有牙和爪子的老虎，没有太多的威胁力，而来自零点电台的消息，梅西跟巴萨董事会的关系已经决裂，有球是跟巴托梅乌，至今没跟巴托梅乌说过话。",
"巴托梅乌最近在接受采访的时候表示“不会和梅西介入冲突，坚决不会卖掉梅西，要让梅西在巴萨终老。”这番言论为巴托梅乌引来吐槽声音，事实上巴托梅乌不卖梅西的原因是一己私利而已，他不想在自己做主席的时候让梅西离开，这样他会成为千古罪人，所以他态度十分强硬，如果不是这个原因，早就卖掉梅西了。",
"在梅西离队后，巴托梅乌试图去缓解董事会跟梅西的关系，但双方裂痕已经产生了，根本无法愈合了，接下来摆在巴萨面前的只有两个选择：要不本赛季跟梅西续约，要不梅西下赛季自由身离队了。"
}
}
```

#### 输出示例

```
{
  "Response": {
    "FirstClassification": {
      "Id": 110,
      "Label": "sports",
      "Name": "体育",
      "Score": 0.9999808073043823
    },
    "RequestId": "cb3fe220-82f5-4359-9a48-9450573186e8",
    "SecondClassification": {
      "Id": 11004,
      "Label": "sports_footballexternal",
      "Name": "体育_国际足球",
      "Score": 0.8565764427185059
    },
    "ThirdClassification": {
      "Id": 11004002,
      "Label": "sports_footballexternal_xijia",
      "Name": "体育_国际足球_西甲",
      "Score": 0.8565764427185059
    }
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub](#) [Gitee](#)

- Tencent Cloud SDK 3.0 for .NET: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub](#) [Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub](#) [Gitee](#)

#### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                    | 描述                     |
|----------------------------------------|------------------------|
| FailedOperation.BalanceInsufficient    | 余额不足，开通失败，请充值后再开通。     |
| FailedOperation.InnerError             | 服务内部错误，请重试。            |
| FailedOperation.NoResults              | 暂无结果，请更换文本重试。          |
| FailedOperation.NotFoundData           | 未查询到结果。                |
| FailedOperation.RequestTimeout         | 后端服务超时。                |
| FailedOperation.ResourceBusy           | 服务器繁忙，请稍后再试。           |
| FailedOperation.RpcFail                | RPC请求失败，一般为算法微服务故障。    |
| FailedOperation.Unknown                | 未知错误。                  |
| InternalError                          | 内部错误。                  |
| InternalError.ServiceCallError         | 内部服务调用错误。              |
| InternalError.ServiceError             | 内部服务调用失败。              |
| InternalError.TextClassifyError        | 文本分类异常。                |
| InternalError.TextParsingError         | 文本解析异常。                |
| InvalidParameterValue.InvalidParameter | 参数不合法。                 |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。          |
| InvalidParameterValue.Text             | Text非法，请参考Text参数说明。    |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。              |
| ResourceNotFound                       | 资源不存在。                 |
| ResourceUnavailable.Freeze             | 账号已被冻结。                |
| ResourceUnavailable.InArrears          | 账号已欠费。                 |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover            | 资源已被回收。                |
| ResourceUnavailable.StopUsing          | 账号已停服。                 |

# 情感分析V2

最近更新时间：2025-03-20 12:04:47

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

情感分析接口能够对带有情感色彩的主观性文本进行分析、处理、归纳和推理，识别出用户的情感倾向，是积极、中性还是消极，并且提供各自概率。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称    | 必选 | 类型     | 描述                                            |
|---------|----|--------|-----------------------------------------------|
| Action  | 是  | String | <a href="#">公共参数</a> ，本接口取值：AnalyzeSentiment。 |
| Version | 是  | String | <a href="#">公共参数</a> ，本接口取值：2019-04-08。       |
| Region  | 否  | String | <a href="#">公共参数</a> ，本接口不需要传递此参数。            |
| Text    | 是  | String | 待分析的文本（仅支持UTF-8格式，不超过200字）。                   |

## 3. 输出参数

| 参数名称      | 类型     | 描述                                                                                     |
|-----------|--------|----------------------------------------------------------------------------------------|
| Positive  | Float  | 正面情感概率。                                                                                |
| Neutral   | Float  | 中性情感概率。                                                                                |
| Negative  | Float  | 负面情感概率。                                                                                |
| Sentiment | String | 情感分类结果：<br>positive：正面情感<br>negative：负面情感<br>neutral：中性、无情感                            |
| RequestId | String | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 情感分析调用示例

情感分析。

#### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: AnalyzeSentiment
<公共请求参数>

{
```

```
"Text": "我真开心。"
}
```

#### 输出示例

```
{
  "Response": {
    "Negative": 0.004367333836853504,
    "Neutral": 0.06927284598350525,
    "Positive": 0.9263598322868347,
    "RequestId": "848b909b-5ed7-44ad-b4d0-72bcf0be4f2a",
    "Sentiment": "positive"
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                 | 描述                  |
|-------------------------------------|---------------------|
| FailedOperation.BalanceInsufficient | 余额不足，开通失败，请充值后再开通。  |
| FailedOperation.InnerError          | 服务内部错误，请重试。         |
| FailedOperation.NoResults           | 暂无结果，请更换文本重试。       |
| FailedOperation.NotFoundData        | 未查询到结果。             |
| FailedOperation.RequestTimeout      | 后端服务超时。             |
| FailedOperation.ResourceBusy        | 服务器繁忙，请稍后再试。        |
| FailedOperation.RpcFail             | RPC请求失败，一般为算法微服务故障。 |
| FailedOperation.Unknown             | 未知错误。               |



| 错误码                                    | 描述                     |
|----------------------------------------|------------------------|
| InternalServerError                    | 内部错误。                  |
| InternalServerError.ServiceCallError   | 内部服务调用错误。              |
| InternalServerError.ServiceError       | 内部服务调用失败。              |
| InternalServerError.TextClassifyError  | 文本分类异常。                |
| InternalServerError.TextParsingError   | 文本解析异常。                |
| InvalidParameterValue.InvalidParameter | 参数不合法。                 |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。          |
| InvalidParameterValue.Text             | Text非法，请参考Text参数说明。    |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。              |
| ResourceNotFound                       | 资源不存在。                 |
| ResourceUnavailable.Freeze             | 账号已被冻结。                |
| ResourceUnavailable.InArrears          | 账号已欠费。                 |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover            | 资源已被回收。                |
| ResourceUnavailable.StopUsing          | 账号已停服。                 |
| UnsupportedOperation.PkgExhausted      | 资源包余量已用尽。              |

# 句子相似度V2

最近更新时间：2025-03-20 12:04:46

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

通过计算句子间的语义相似性，帮助您快速找到文本中重复或相似的句子，用于文本聚类、相似问题检索等应用场景。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称               | 必选 | 类型                                    | 描述                                                      |
|--------------------|----|---------------------------------------|---------------------------------------------------------|
| Action             | 是  | String                                | <a href="#">公共参数</a> ，本接口取值：EvaluateSentenceSimilarity。 |
| Version            | 是  | String                                | <a href="#">公共参数</a> ，本接口取值：2019-04-08。                 |
| Region             | 否  | String                                | <a href="#">公共参数</a> ，本接口不需要传递此参数。                      |
| SentencePairList.N | 是  | Array of <a href="#">SentencePair</a> | 待分析的句子对数组。句子对应不超过1对，仅支持中文文本，原句子与目标句子均不应超过500字符。         |

## 3. 输出参数

| 参数名称      | 类型             | 描述                                                                                     |
|-----------|----------------|----------------------------------------------------------------------------------------|
| ScoreList | Array of Float | 每个句子对的相似度分值。                                                                           |
| RequestId | String         | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 句子相似度调用示例

分析句子相似度

#### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: EvaluateSentenceSimilarity
<公共请求参数>

{
  "SentencePairList": [
    {
      "SourceText": "北京到上海的火车票",
      "TargetText": "上海到北京的火车票"
    }
  ]
}
```

## 输出示例

```
{
  "Response": {
    "RequestId": "d37e6fc7-aa12-42e4-b68a-e74dbe7548cb",
    "ScoreList": [
      0.903129852566213
    ]
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                  | 描述                  |
|--------------------------------------|---------------------|
| FailedOperation.BalanceInsufficient  | 余额不足，开通失败，请充值后再开通。  |
| FailedOperation.InnerError           | 服务内部错误，请重试。         |
| FailedOperation.NoResults            | 暂无结果，请更换文本重试。       |
| FailedOperation.NotFoundData         | 未查询到结果。             |
| FailedOperation.RequestTimeout       | 后端服务超时。             |
| FailedOperation.ResourceBusy         | 服务器繁忙，请稍后再试。        |
| FailedOperation.RpcFail              | RPC请求失败，一般为算法微服务故障。 |
| FailedOperation.UnKnowError          | 内部错误。               |
| FailedOperation.Unknown              | 未知错误。               |
| InternalServerError                  | 内部错误。               |
| InternalServerError.ServiceCallError | 内部服务调用错误。           |
| InternalServerError.ServiceError     | 内部服务调用失败。           |

| 错误码                                    | 描述                     |
|----------------------------------------|------------------------|
| InternalError.TextParsingError         | 文本解析异常。                |
| InvalidParameterValue.InvalidParameter | 参数不合法。                 |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。          |
| InvalidParameterValue.TextNumTooMuch   | 输入文本超出数量限制             |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。              |
| ResourceInsufficient.QuotaRunOut       | 额度用尽，请充值后重试            |
| ResourceNotFound                       | 资源不存在。                 |
| ResourceUnavailable.Freeze             | 账号已被冻结。                |
| ResourceUnavailable.InArrears          | 账号已欠费。                 |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover            | 资源已被回收。                |
| ResourceUnavailable.StopUsing          | 账号已停服。                 |
| UnsupportedOperation.PkgExhausted      | 资源包余量已用尽。              |

# 词相似度V2

最近更新时间：2025-03-20 12:04:46

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

评估两个词语在语义空间的相似程度，为您的场景应用提供有力支持，如关键词过滤、热门话题挖掘等。（目前仅支持中文）

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称       | 必选 | 类型     | 描述                                                  |
|------------|----|--------|-----------------------------------------------------|
| Action     | 是  | String | <a href="#">公共参数</a> ，本接口取值：EvaluateWordSimilarity。 |
| Version    | 是  | String | <a href="#">公共参数</a> ，本接口取值：2019-04-08。             |
| Region     | 否  | String | <a href="#">公共参数</a> ，本接口不需要传递此参数。                  |
| SourceWord | 是  | String | 计算相似度的源词。（仅支持UTF-8格式，不超过10字符）                       |
| TargetWord | 是  | String | 计算相似度的目标词。（仅支持UTF-8格式，不超过10字符）                      |

## 3. 输出参数

| 参数名称       | 类型     | 描述                                                                                     |
|------------|--------|----------------------------------------------------------------------------------------|
| Similarity | Float  | 词相似度分值。                                                                                |
| RequestId  | String | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 词相似度调用示例

分析词相似度

#### 输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: EvaluateWordSimilarity
<公共请求参数>

{
  "SourceWord": "香港",
  "TargetWord": "北京"
}
```

#### 输出示例

```
{
  "Response": {
    "RequestId": "02bd6796-515a-452e-a455-23523fd6cfd9",
    "Similarity": 0.5935152769088745
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                    | 描述                  |
|----------------------------------------|---------------------|
| FailedOperation.BalanceInsufficient    | 余额不足，开通失败，请充值后再开通。  |
| FailedOperation.InnerError             | 服务内部错误，请重试。         |
| FailedOperation.NoResults              | 暂无结果，请更换文本重试。       |
| FailedOperation.NotFoundData           | 未查询到结果。             |
| FailedOperation.RequestTimeout         | 后端服务超时。             |
| FailedOperation.ResourceBusy           | 服务器繁忙，请稍后再试。        |
| FailedOperation.RpcFail                | RPC请求失败，一般为算法微服务故障。 |
| FailedOperation.UnKnowError            | 内部错误。               |
| FailedOperation.Unknown                | 未知错误。               |
| InternalError                          | 内部错误。               |
| InternalError.ServiceCallError         | 内部服务调用错误。           |
| InternalError.ServiceError             | 内部服务调用失败。           |
| InvalidParameterValue.InvalidParameter | 参数不合法。              |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。       |

| 错误码                                  | 描述                     |
|--------------------------------------|------------------------|
| InvalidParameterValue.TextNumTooMuch | 输入文本超出数量限制             |
| InvalidParameterValue.TextTooLong    | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit   | 资源用量达到上限。              |
| ResourceInsufficient.QuotaRunOut     | 额度用尽，请充值后重试            |
| ResourceNotFound                     | 资源不存在。                 |
| ResourceUnavailable.Freeze           | 账号已被冻结。                |
| ResourceUnavailable.InArrears        | 账号已欠费。                 |
| ResourceUnavailable.IsOpening        | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist         | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover          | 资源已被回收。                |
| ResourceUnavailable.StopUsing        | 账号已停服。                 |

# 词法分析V2

最近更新时间：2025-03-20 12:04:46

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

通过精准地对文本进行分词、词性标注、命名实体识别等功能，助您更好地理解文本内容，挖掘出潜在的价值信息。

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称    | 必选 | 类型     | 描述                                      |
|---------|----|--------|-----------------------------------------|
| Action  | 是  | String | <a href="#">公共参数</a> ，本接口取值：ParseWords。 |
| Version | 是  | String | <a href="#">公共参数</a> ，本接口取值：2019-04-08。 |
| Region  | 否  | String | <a href="#">公共参数</a> ，本接口不需要传递此参数。      |
| Text    | 是  | String | 待分析的文本（支持中英文文本，不超过500字符）                |

## 3. 输出参数

| 参数名称                | 类型                                          | 描述                                                                                     |
|---------------------|---------------------------------------------|----------------------------------------------------------------------------------------|
| NormalText          | String                                      | 输入文本正则化的结果。（包括对英文文本中的开头和实体进行大写等）                                                       |
| BasicParticiples    | Array of <a href="#">BasicParticiple</a>    | 基础粒度分词和词性标注的结果。（请参见附录 <a href="#">词性表</a> ）                                            |
| CompoundParticiples | Array of <a href="#">CompoundParticiple</a> | 复合粒度分词和词性标注的结果。（请参见附录 <a href="#">词性表</a> ）                                            |
| Entities            | Array of <a href="#">Entity</a>             | 实体识别结果。（请参见附录 <a href="#">实体类型数据</a> ）                                                 |
| RequestId           | String                                      | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 词法分析调用示例

词法分析

输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action: ParseWords
<公共请求参数>

{
  "Text": "我很喜欢看流浪地球这个电影"
}
```



## 输出示例

```
{
  "Response": {
    "BasicParticiples": [
      {
        "BeginOffset": 0,
        "Length": 1,
        "Pos": "PN",
        "Word": "我"
      },
      {
        "BeginOffset": 1,
        "Length": 1,
        "Pos": "AD",
        "Word": "很"
      },
      {
        "BeginOffset": 2,
        "Length": 2,
        "Pos": "VV",
        "Word": "喜欢"
      },
      {
        "BeginOffset": 4,
        "Length": 1,
        "Pos": "VV",
        "Word": "看"
      },
      {
        "BeginOffset": 5,
        "Length": 2,
        "Pos": "VV",
        "Word": "流浪"
      },
      {
        "BeginOffset": 7,
        "Length": 2,
        "Pos": "NN",
        "Word": "地球"
      },
      {
        "BeginOffset": 9,
        "Length": 2,
        "Pos": "DT",
        "Word": "这个"
      },
      {
        "BeginOffset": 11,
        "Length": 2,
        "Pos": "NN",
        "Word": "电影"
      }
    ],
    "CompoundParticiples": [
      {
        "BeginOffset": 0,
```

```
"Length": 1,
"Pos": "PN",
"Word": "我"
},
{
  "BeginOffset": 1,
  "Length": 1,
  "Pos": "AD",
  "Word": "很"
},
{
  "BeginOffset": 2,
  "Length": 2,
  "Pos": "VV",
  "Word": "喜欢"
},
{
  "BeginOffset": 4,
  "Length": 1,
  "Pos": "VV",
  "Word": "看"
},
{
  "BeginOffset": 5,
  "Length": 4,
  "Pos": "NN",
  "Word": "流浪地球"
},
{
  "BeginOffset": 9,
  "Length": 2,
  "Pos": "DT",
  "Word": "这个"
},
{
  "BeginOffset": 11,
  "Length": 2,
  "Pos": "NN",
  "Word": "电影"
}
],
"Entities": [
  {
    "BeginOffset": 5,
    "Length": 4,
    "Name": "电影",
    "Type": "work.movie",
    "Word": "流浪地球"
  }
],
"NormalText": "我很喜欢看流浪地球这个电影",
"RequestId": "6cd06531-5b5e-46ee-8ec2-ab66a9730e74"
}
```

## 5. 开发者资源

## 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

## API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

## SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub Gitee](#)

## 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                    | 描述                  |
|----------------------------------------|---------------------|
| FailedOperation.BalanceInsufficient    | 余额不足，开通失败，请充值后再开通。  |
| FailedOperation.InnerError             | 服务内部错误，请重试。         |
| FailedOperation.NoResults              | 暂无结果，请更换文本重试。       |
| FailedOperation.NotFoundData           | 未查询到结果。             |
| FailedOperation.RequestTimeout         | 后端服务超时。             |
| FailedOperation.ResourceBusy           | 服务器繁忙，请稍后再试。        |
| FailedOperation.RpcFail                | RPC请求失败，一般为算法微服务故障。 |
| FailedOperation.UnKnowError            | 内部错误。               |
| FailedOperation.Unknownn               | 未知错误。               |
| InternalError                          | 内部错误。               |
| InternalError.ServiceCallError         | 内部服务调用错误。           |
| InternalError.ServiceError             | 内部服务调用失败。           |
| InvalidParameterValue.InvalidParameter | 参数不合法。              |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。       |
| InvalidParameterValue.TextNumTooMuch   | 输入文本超出数量限制          |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制          |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。           |
| ResourceInsufficient.QuotaRunOut       | 额度用尽，请充值后重试         |
| ResourceNotFound                       | 资源不存在。              |
| ResourceUnavailable.Freeze             | 账号已被冻结。             |
| ResourceUnavailable.InArrears          | 账号已欠费。              |

| 错误码                               | 描述                     |
|-----------------------------------|------------------------|
| ResourceUnavailable.IsOpening     | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist      | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover       | 资源已被回收。                |
| ResourceUnavailable.StopUsing     | 账号已停服。                 |
| UnsupportedOperation.PkgExhausted | 资源包余量已用尽。              |

# 相似词召回

最近更新时间：2025-03-20 12:04:46

## 1. 接口描述

接口请求域名：nlp.tencentcloudapi.com。

基于大数据和深度学习技术，可以快速地找到与给定词语高度相似的其他词语，有助于提高搜索和推荐的准确性。（目前仅支持中文）

推荐使用 API Explorer

</> 点击调试

API Explorer 提供了在线调用、签名验证、SDK 代码生成和快速检索接口等能力。您可查看每次调用的请求内容和返回结果以及自动生成 SDK 调用示例。

## 2. 输入参数

以下请求参数列表仅列出了接口请求参数和部分公共参数，完整公共参数列表见 [公共请求参数](#)。

| 参数名称    | 必选 | 类型      | 描述                                                |
|---------|----|---------|---------------------------------------------------|
| Action  | 是  | String  | <a href="#">公共参数</a> ，本接口取值：RetrieveSimilarWords。 |
| Version | 是  | String  | <a href="#">公共参数</a> ，本接口取值：2019-04-08。           |
| Region  | 否  | String  | <a href="#">公共参数</a> ，本接口不需要传递此参数。                |
| Text    | 是  | String  | 输入的词语。（仅支持UTF-8格式，不超过10字符）                        |
| Number  | 是  | Integer | 召回的相似词个数，取值范围为1-20。                               |

## 3. 输出参数

| 参数名称      | 类型              | 描述                                                                                     |
|-----------|-----------------|----------------------------------------------------------------------------------------|
| WordList  | Array of String | 召回的相似词数组。                                                                              |
| RequestId | String          | 唯一请求 ID，由服务端生成，每次请求都会返回（若请求因其他原因未能抵达服务端，则该次请求不会获得 RequestId）。定位问题时需要提供该次请求的 RequestId。 |

## 4. 示例

### 示例1 相似词召回调用示例

相似词召回

输入示例

```
POST / HTTP/1.1
Host: nlp.tencentcloudapi.com
Content-Type: application/json
X-TC-Action:RetrieveSimilarWords
<公共请求参数>

{
  "Text": "林丹",
  "Number": 10
}
```

输出示例

```
{
  "Response": {
    "RequestId": "e246e84f-de88-46d0-b5dd-68ff94db40ad",
    "WordList": [
      "李宗伟",
      "桃田贤斗",
      "马龙",
      "纳达尔",
      "刘翔",
      "张本智和",
      "费德勒",
      "孙杨",
      "宁泽涛",
      "朱婷"
    ]
  }
}
```

## 5. 开发者资源

### 腾讯云 API 平台

[腾讯云 API 平台](#) 是综合 API 文档、错误码、API Explorer 及 SDK 等资源的统一查询平台，方便您从同一入口查询及使用腾讯云提供的所有 API 服务。

### API Inspector

用户可通过 [API Inspector](#) 查看控制台每一步操作关联的 API 调用情况，并自动生成各语言版本的 API 代码，也可前往 [API Explorer](#) 进行在线调试。

### SDK

云 API 3.0 提供了配套的开发工具集（SDK），支持多种编程语言，能更方便的调用 API。

- Tencent Cloud SDK 3.0 for Python: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Java: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for PHP: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Go: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Node.js: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for .NET: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for C++: [GitHub Gitee](#)
- Tencent Cloud SDK 3.0 for Ruby: [GitHub Gitee](#)

### 命令行工具

- [Tencent Cloud CLI 3.0](#)

## 6. 错误码

以下仅列出了接口业务逻辑相关的错误码，其他错误码详见 [公共错误码](#)。

| 错误码                                 | 描述                 |
|-------------------------------------|--------------------|
| FailedOperation.BalanceInsufficient | 余额不足，开通失败，请充值后再开通。 |
| FailedOperation.InnerError          | 服务内部错误，请重试。        |
| FailedOperation.NoResults           | 暂无结果，请更换文本重试。      |
| FailedOperation.NotFoundData        | 未查询到结果。            |
| FailedOperation.RequestTimeout      | 后端服务超时。            |
| FailedOperation.ResourceBusy        | 服务器繁忙，请稍后再试。       |

| 错误码                                    | 描述                     |
|----------------------------------------|------------------------|
| FailedOperation.RpcFail                | RPC请求失败，一般为算法微服务故障。    |
| FailedOperation.UnKnowError            | 内部错误。                  |
| FailedOperation.Unknown                | 未知错误。                  |
| InternalError                          | 内部错误。                  |
| InternalError.ServiceCallError         | 内部服务调用错误。              |
| InternalError.ServiceError             | 内部服务调用失败。              |
| InvalidParameterValue.InvalidParameter | 参数不合法。                 |
| InvalidParameterValue.SensitiveText    | Text输入含有敏感信息。          |
| InvalidParameterValue.TextNumTooMuch   | 输入文本超出数量限制             |
| InvalidParameterValue.TextTooLong      | 输入文本超出长度限制             |
| LimitExceeded.ResourceReachedLimit     | 资源用量达到上限。              |
| ResourceInsufficient.QuotaRunOut       | 额度用尽，请充值后重试            |
| ResourceNotFound                       | 资源不存在。                 |
| ResourceUnavailable.Freeze             | 账号已被冻结。                |
| ResourceUnavailable.InArrears          | 账号已欠费。                 |
| ResourceUnavailable.IsOpening          | 服务正在开通中，请稍等。           |
| ResourceUnavailable.NotExist           | 计费状态未知，请确认是否已在控制台开通服务。 |
| ResourceUnavailable.Recover            | 资源已被回收。                |
| ResourceUnavailable.StopUsing          | 账号已停服。                 |
| UnsupportedOperation.PkgExhausted      | 资源包余量已用尽。              |

# 数据结构

最近更新时间：2023-09-15 02:14:29

## BasicParticiple

基础粒度分词和词性标注的结果

被如下接口引用：ParseWords。

| 名称          | 类型      | 描述                    |
|-------------|---------|-----------------------|
| Word        | String  | 基础词。                  |
| BeginOffset | Integer | 基础词在NormalText中的起始位置。 |
| Length      | Integer | 基础词的长度。               |
| Pos         | String  | 词性。                   |

## Category

分类详细信息

被如下接口引用：ClassifyContent。

| 名称    | 类型      | 描述                                  |
|-------|---------|-------------------------------------|
| Id    | Integer | 分类id。<br>注意：此字段可能返回 null，表示取不到有效值。  |
| Label | String  | 分类英文名。<br>注意：此字段可能返回 null，表示取不到有效值。 |
| Name  | String  | 分类中文名。<br>注意：此字段可能返回 null，表示取不到有效值。 |
| Score | Float   | 分类置信度。<br>注意：此字段可能返回 null，表示取不到有效值。 |

## CompoundParticiple

复合粒度分词和词性标注的结果。

被如下接口引用：ParseWords。

| 名称          | 类型      | 描述                    |
|-------------|---------|-----------------------|
| Word        | String  | 基础词。                  |
| BeginOffset | Integer | 基础词在NormalText中的起始位置。 |
| Length      | Integer | 基础词的长度。               |
| Pos         | String  | 词性。                   |

## CorrectionItem

纠错结果列表

被如下接口引用：SentenceCorrection。

| 名称          | 类型      | 描述            |
|-------------|---------|---------------|
| Order       | Integer | 纠错句子的序号。      |
| BeginOffset | Integer | 错误的起始位置，从0开始。 |



| 名称             | 类型              | 描述                                                  |
|----------------|-----------------|-----------------------------------------------------|
| Len            | Integer         | 错误内容长度。                                             |
| Word           | String          | 错误内容。                                               |
| CorrectWord    | Array of String | 纠错结果，当为删除类错误时，结果为null。<br>注意：此字段可能返回 null，表示取不到有效值。 |
| CorrectionType | Integer         | 纠错类型。0：替换；1：插入；2：删除。                                |
| Confidence     | Integer         | 纠错信息置信度。0：error；1：warning；error的置信度更高。（仅供参考）        |
| DescriptionZh  | String          | 纠错信息中文描述。<br>注意：此字段可能返回 null，表示取不到有效值。              |
| DescriptionEn  | String          | 纠错信息英文描述。<br>注意：此字段可能返回 null，表示取不到有效值。              |

## Embellish

文本润色结果

被如下接口引用：TextEmbellish。

| 名称            | 类型     | 描述                                                                                                                                                     |
|---------------|--------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| Text          | String | 润色后的文本。<br>注意：此字段可能返回 null，表示取不到有效值。<br>示例值：而这一次的中国奥运健儿击败强敌，毫无悬念地勇夺了中国代表团的冬奥会首金和总分第三的金                                                                 |
| EmbellishType | String | 润色类型。类型如下：<br>expansion：扩写<br>rewriting：改写<br>translation_m2a：从现代文改写为古文<br>translation_a2m：从古文改写为现代文<br><br>注意：此字段可能返回 null，表示取不到有效值。<br>示例值：expansion |

## Entity

实体识别结果。

被如下接口引用：ParseWords。

| 名称          | 类型      | 描述                    |
|-------------|---------|-----------------------|
| Word        | String  | 基础词。                  |
| BeginOffset | Integer | 基础词在NormalText中的起始位置。 |
| Length      | Integer | 基础词的长度。               |
| Type        | String  | 实体类型的标准名字。            |
| Name        | String  | 类型名字的自然语言表达。（中文或英文）   |

## KeywordSentence

通过关键词生成的句子信息

被如下接口引用：GenerateKeywordSentence。

| 名称         | 类型     | 描述          |
|------------|--------|-------------|
| TargetText | String | 通过关键词生成的句子。 |

## SentencePair

待分析的句子对

被如下接口引用：EvaluateSentenceSimilarity。

| 名称         | 类型     | 必选 | 描述                                     |
|------------|--------|----|----------------------------------------|
| SourceText | String | 是  | 需要与目标句子计算相似度的源句子。（仅支持UTF-8格式，不超过500字符） |
| TargetText | String | 是  | 目标句子。（仅支持UTF-8格式，不超过500字符）             |

## Writing

文本续写结果

被如下接口引用：TextWriting。

| 名称         | 类型     | 描述     |
|------------|--------|--------|
| TargetText | String | 续写的文本。 |
| PrefixText | String | 续写的前缀。 |

# 错误码

最近更新时间：2024-12-20 01:45:11

## 功能说明

如果返回结果中存在 Error 字段，则表示调用 API 接口失败。例如：

```
{
  "Response": {
    "Error": {
      "Code": "AuthFailure.SignatureFailure",
      "Message": "The provided credentials could not be validated. Please check your signature is correct.",
    },
    "RequestId": "ed93f3cb-f35e-473f-b9f3-0d451b8b79c6"
  }
}
```

Error 中的 Code 表示错误码，Message 表示该错误的具体信息。

## 错误码列表

### 公共错误码

| 错误码                               | 说明                                                       |
|-----------------------------------|----------------------------------------------------------|
| ActionOffline                     | 接口已下线。                                                   |
| AuthFailure.InvalidAuthorization  | 请求头部的 Authorization 不符合腾讯云标准。                            |
| AuthFailure.InvalidSecretId       | 密钥非法（不是云 API 密钥类型）。                                      |
| AuthFailure.MFAFailure            | MFA 错误。                                                  |
| AuthFailure.SecretIdNotFound      | 密钥不存在。请在 控制台 检查密钥是否已被删除或者禁用，如状态正常，请检查密钥是否填写正确，注意前后不得有空格。 |
| AuthFailure.SignatureExpire       | 签名过期。Timestamp 和服务器时间相差不得超过五分钟，请检查本地时间是否和标准时间同步。         |
| AuthFailure.SignatureFailure      | 签名错误。签名计算错误，请对照调用方式中的签名方法文档检查签名计算过程。                     |
| AuthFailure.TokenFailure          | token 错误。                                                |
| AuthFailure.UnauthorizedOperation | 请求未授权。请参考 CAM 文档对鉴权的说明。                                  |
| DryRunOperation                   | DryRun 操作，代表请求将会是成功的，只是多传了 DryRun 参数。                    |
| FailedOperation                   | 操作失败。                                                    |
| InternalError                     | 内部错误。                                                    |
| InvalidAction                     | 接口不存在。                                                   |
| InvalidParameter                  | 参数错误（包括参数格式、类型等错误）。                                      |
| InvalidParameterValue             | 参数取值错误。                                                  |
| InvalidRequest                    | 请求 body 的 multipart 格式错误。                                |
| IpInBlacklist                     | IP 地址在黑名单中。                                              |
| IpNotInWhitelist                  | IP 地址不在白名单中。                                             |
| LimitExceeded                     | 超过配额限制。                                                  |
| MissingParameter                  | 缺少参数。                                                    |

| 错误码                                               | 说明                                |
|---------------------------------------------------|-----------------------------------|
| NoSuchProduct                                     | 产品不存在                             |
| NoSuchVersion                                     | 接口版本不存在。                          |
| RequestLimitExceeded                              | 请求的次数超过了频率限制。                     |
| RequestLimitExceeded.GlobalRegionUinLimitExceeded | 主账号超过频率限制。                        |
| RequestLimitExceeded.IPLimitExceeded              | IP 限频。                            |
| RequestLimitExceeded.UinLimitExceeded             | 主账号限频。                            |
| RequestSizeLimitExceeded                          | 请求包超过限制大小。                        |
| ResourceInUse                                     | 资源被占用。                            |
| ResourceInsufficient                              | 资源不足。                             |
| ResourceNotFound                                  | 资源不存在。                            |
| ResourceUnavailable                               | 资源不可用。                            |
| ResponseSizeLimitExceeded                         | 返回包超过限制大小。                        |
| ServiceUnavailable                                | 当前服务暂时不可用。                        |
| UnauthorizedOperation                             | 未授权操作。                            |
| UnknownParameter                                  | 未知参数错误，用户多传未定义的参数会导致错误。           |
| UnsupportedOperation                              | 操作不支持。                            |
| UnsupportedProtocol                               | http(s) 请求协议错误，只支持 GET 和 POST 请求。 |
| UnsupportedRegion                                 | 接口不支持所传地域。                        |

## 业务错误码

| 错误码                                 | 说明                  |
|-------------------------------------|---------------------|
| FailedOperation.BalanceInsufficient | 余额不足，开通失败，请充值后再开通。  |
| FailedOperation.InnerError          | 服务内部错误，请重试。         |
| FailedOperation.NoCouplets          | 暂无春联生成，请更换关键词重试。    |
| FailedOperation.NoPoetry            | 暂无诗词生成，请更换关键词重试。    |
| FailedOperation.NoResults           | 暂无结果，请更换文本重试。       |
| FailedOperation.NotFoundData        | 未查询到结果。             |
| FailedOperation.RequestTimeout      | 后端服务超时。             |
| FailedOperation.ResourceBusy        | 服务器繁忙，请稍后再试。        |
| FailedOperation.RpcFail             | RPC请求失败，一般为算法微服务故障。 |
| FailedOperation.UnKnowError         | 内部错误。               |
| FailedOperation.Unknown             | 未知错误。               |
| InternalError.ServiceCallError      | 内部服务调用错误。           |
| InternalError.ServiceError          | 内部服务调用失败。           |
| InternalError.TextClassifyError     | 文本分类异常。             |
| InternalError.TextParsingError      | 文本解析异常。             |
| InvalidParameter.ServiceError       | 服务调用失败。             |

| 错误码                                       | 说明                                   |
|-------------------------------------------|--------------------------------------|
| InvalidParameter.TextTooLongCode          | 文本长度超过限制。                            |
| InvalidParameterValue.Genre               | Genre非法，请参考Genre参数说明。                |
| InvalidParameterValue.InvalidParameter    | 参数不合法。                               |
| InvalidParameterValue.PoetryType          | PoetryType非法，请参考PoetryType参数说明。      |
| InvalidParameterValue.SensitiveText       | Text输入含有敏感信息。                        |
| InvalidParameterValue.TargetType          | TargetType非法，请参考TargetType参数说明。      |
| InvalidParameterValue.Text                | Text非法，请参考Text参数说明。                  |
| InvalidParameterValue.TextNumTooMuch      | 输入文本超出数量限制                           |
| InvalidParameterValue.TextTooLong         | 输入文本超出长度限制                           |
| LimitExceeded.ResourceReachedLimit        | 资源用量达到上限。                            |
| ResourceInsufficient.QuotaRunOut          | 额度用尽，请充值后重试                          |
| ResourceUnavailable.Freeze                | 账号已被冻结。                              |
| ResourceUnavailable.InArrears             | 账号已欠费。                               |
| ResourceUnavailable.IsOpening             | 服务正在开通中，请稍等。                         |
| ResourceUnavailable.NotExist              | 计费状态未知，请确认是否已在控制台开通服务。               |
| ResourceUnavailable.Recover               | 资源已被回收。                              |
| ResourceUnavailable.ServiceNotOpenedError | 您的账号尚未开通NLP服务，请登录腾讯云NLP控制台进行服务开通后再使用 |
| ResourceUnavailable.StopUsing             | 账号已停服。                               |
| UnsupportedOperation.PkgExhausted         | 资源包余量已用尽。                            |

# 附录

## 词性表

最近更新时间：2023-04-25 13:41:21

### 中文词性标签集合

| Tag | Description                                        | Example           |
|-----|----------------------------------------------------|-------------------|
| VA  | Predicative adjective                              | 典型[VA] 的例子        |
| VC  | Copula                                             | 他 是[VC] 学生        |
| VE  | you3 as the main verb                              | 有[VE] 数十 人        |
| VV  | Other verb                                         | 缩短[VV] 了 研制 周期    |
| NR  | Proper Noun                                        | 四川省[NR] 政府        |
| NT  | Temporal Noun                                      | 十二月[NT]           |
| NN  | Other Noun                                         | 深圳 证券[NN] 交易所[NN] |
| LC  | Localizer                                          | 八十年代 以来[LC]       |
| PN  | Pronoun                                            | 他[PN] 说           |
| DT  | Determiner                                         | 这[DT] 一 成果        |
| CD  | Cardinal Number                                    | 一亿[CD]            |
| OD  | Ordinal Number                                     | 第一[OD]            |
| M   | Measure word                                       | 三 名[M] 运动员        |
| AD  | Adverb                                             | 十分[AD] 重要         |
| P   | Preposition                                        | 到[P] 目前 为止        |
| CC  | Coordinating conjunction                           | 工作 或[CC] 学习       |
| CS  | Subordinating conjunction                          | 虽然[CS] 历经 风雨      |
| DEC | de5 as a complementizer or a nominalizer           | 重要 的[DEC] 能源      |
| DEG | de5 as a genitive marker and an associative marker | 合作 的[DEG] 新 篇章    |
| DER | Resultative de5                                    | 他 跑 得[DER] 很 快    |
| DEV | Manner de5                                         | 高兴 地[DEV] 说       |
| AS  | Aspect Particle                                    | 拓宽 了[AS] 贸易 渠道    |
| SP  | Sentence-final particle                            | 他 还 好 吧[SP]       |
| ETC | used for word 等 and 等等                             | 科技 文教 等[ETC] 领域   |
| MSP | Other particle                                     | 他 所[MSP] 需要 的     |
| IJ  | Interjection                                       | 啊[IJ]             |
| ON  | Onomatopoeia                                       | 雨 哗哗[ON] 地 下      |
| LB  | bei4 in long bei-construction                      | 他 被[LB] 我 训 了     |
| SB  | bei4 in short bei-construction                     | 他 被[SB] 训 了       |
| BA  | ba3 in ba-construction                             | 他 把[BA] 你 骗 了     |
| JJ  | other noun-modifier                                | 共同[JJ] 的 目标       |

| Tag | Description         | Example          |
|-----|---------------------|------------------|
| FW  | Foreign Word        | 卡拉OK[FW]         |
| PU  | Punctuation         | , [PU]           |
| EM  | Emoticons           | 呵呵： ) [EM]       |
| IC  | Typo                | -                |
| NOI | Grammar error       | -                |
| URL | URL                 | tencent.com[URL] |
| X   | Multiplication sign | 130 x[X] 194 cm  |

### 英文词性标签集合

| Tag   | Description                              | Example                     |
|-------|------------------------------------------|-----------------------------|
| CC    | Coordinating conjunction                 | and, but                    |
| CD    | Cardinal number                          | one, 4000                   |
| DT    | Determiner                               | the, a                      |
| EX    | Existential "there"                      | there                       |
| FW    | Foreign word                             | de facto                    |
| IN    | Preposition or subordinating conjunction | in, of                      |
| JJ    | Adjective                                | Canadian, many              |
| JJR   | Adjective, comparative                   | more, higher                |
| JJS   | Adjective, superlative                   | most, best                  |
| LS    | List item marker                         | a. b. c.                    |
| MD    | Modal                                    | can, will                   |
| NN    | Noun, singular or mass                   | spokesman, school           |
| NNS   | Noun, plural                             | lawyers, skills             |
| NNP   | Proper noun, singular                    | Los Angeles, Hurricane Hugo |
| NNPS  | Proper noun, plural                      | Americans, Asians           |
| PDT   | Predeterminer                            | all, such                   |
| POS   | Possessive ending                        | 's                          |
| PRP   | Personal pronoun                         | he, she                     |
| PRP\$ | Possessive pronoun                       | his, her                    |
| RB    | Adverb                                   | ultimately, generally       |
| RBR   | Adverb, comparative                      | earlier, further            |
| RBS   | Adverb, superlative                      | most, least                 |
| RP    | Particle                                 | through, off                |
| SYM   | Symbol                                   | /, -                        |
| TO    | "to"                                     | to                          |
| UH    | Interjection                             | Hello, Well                 |
| VB    | Verb, base form                          | make, draw                  |

| Tag   | Description                                            | Example        |
|-------|--------------------------------------------------------|----------------|
| VBD   | Verb, past tense                                       | was, had       |
| VBG   | Verb, gerund or present participle                     | going, getting |
| VBN   | Verb, past participle                                  | been, taken    |
| VBP   | Verb, non-3rd person singular present                  | are, have      |
| VBZ   | Verb, 3rd person singular present                      | is, has        |
| WDT   | Wh-determiner                                          | which, that    |
| WP    | Wh-pronoun                                             | who, what      |
| WP\$  | Possessive wh-pronoun                                  | whose          |
| WRB   | Wh-adverb                                              | where, when    |
| \$    | Dollar symbol                                          | \$             |
| "     | Double quotes                                          | "              |
| ,     | Comma                                                  | ,              |
| -LRB- | Left round bracket                                     | (              |
| -RRB- | Right round bracket                                    | )              |
| .     | Period                                                 | .              |
| :     | Colon                                                  | :              |
| AFX   | Unbound affix                                          | non            |
| HYPH  | Unbound hyphen                                         | -              |
| NFP   | Non-functional (superfluous or decorative) punctuation | /*             |



# 实体类型数据

最近更新时间：2023-07-27 01:16:52

粗粒度实体类型数据见下表，细粒度实体类型数据可通过以下地址下载查看：

- [texsmart-ont-0.3.5.tar.gz](#)
- [texsmart-ont-0.3.5.zip](#)

粗粒度实体类型表

| Tag              | Description                        | Example                      |
|------------------|------------------------------------|------------------------------|
| person.generic   | 人物，既包括实际存在的人，也包括出现于作品或传说中的角色、神仙鬼怪等 | 刘德华、孙悟空、王先生                  |
| loc.generic      | 地点，包括国家、行政区（省市县区镇等）、山脉河流、景点等       | 中国、南山区、科技路、黄河、喜马拉雅山、故宫、火星、中东 |
| org.generic      | 机构，包括公司、学校、政府机构、运动队、俱乐部、乐队等        | 中国驻美大使馆、腾讯、清华大学、曼联、湖南卫视      |
| product.generic  | 产品，包括车、数码电子、工具、衣服、游戏、软件等           | 华为Mate20、牛仔裤、锄头、王者荣耀         |
| work.generic     | 作品，包括小说、散文、诗词、电影、音乐、绘画、雕塑、戏剧、民间传说等 | 红楼梦、流浪地球、乱世佳人、望庐山瀑布          |
| life.organism    | 生物，包括动物、植物、微生物等                    | 狗、蜜蜂、柳树、草履虫                  |
| food.generic     | 食物，包括食品、饮料、水果、蔬菜、菜名等               | 柠檬汁、大米、香蕉、芹菜、鱼香肉丝            |
| medicine         | 医药相关实体，包括疾病、药物、症状等                 | 新冠肺炎、头孢、皮肤红肿                 |
| event.generic    | 事件，包括历史事件、国际会议、体育赛事等               | 第二次世界大战、亚洲羽毛球锦标赛             |
| quantity.generic | 数量，包括数值、数值加单位、数值加量词、数量范围、序数词等      | 2万、三米五、一百匹、三到五个、一些           |
| time.generic     | 时间，包括日期、节日、时长、岁数等                  | 9月中旬、上星期三、五年、12岁、今天          |
| other            | 其他实体，如星座、语言、颜色、技术等                 | 摩羯座、汉语、C++、红色、自然语言处理         |

# 三级分类体系表

最近更新时间：2023-06-27 01:34:14

三级分类体系表

| 新分类ID    | 新分类中文名      | 新分类英文名               | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名     |
|----------|-------------|----------------------|------|--------|------|-------------|
| 100      | 搞笑          | funny                | 搞笑   | funny  |      |             |
| 10000    | 搞笑_其他       | funny_other          | 搞笑   | funny  | 其他   | other       |
| 10001    | 搞笑_段子       | funny_duanzi         | 搞笑   | funny  | 段子   | duanzi      |
| 10002    | 搞笑_神回复      | funny_reply          | 搞笑   | funny  | 神回复  | reply       |
| 10003    | 搞笑_糗事       | funny_qiushi         | 搞笑   | funny  | 糗事   | qiushi      |
| 10004    | 搞笑_神操作      | funny_caozuo         | 搞笑   | funny  | 神操作  | caozuo      |
| 10005    | 搞笑_内涵图      | funny_pic            | 搞笑   | funny  | 内涵图  | pic         |
| 10006    | 搞笑_神剪辑      | funny_remix          | 搞笑   | funny  | 神剪辑  | remix       |
| 10007    | 搞笑_整蛊       | funny_tease          | 搞笑   | funny  | 整蛊   | tease       |
| 10008    | 搞笑_搞怪       | funny_gaoguai        | 搞笑   | funny  | 搞怪   | gaoguai     |
| 10009    | 搞笑_搞笑街访     | funny_interview      | 搞笑   | funny  | 搞笑街访 | interview   |
| 10010    | 搞笑_表演秀      | funny_biaoyan        | 搞笑   | funny  | 表演秀  | biaoyan     |
| 10011    | 搞笑_搞笑影视     | funny_film           | 搞笑   | funny  | 搞笑影视 | film        |
| 10012    | 搞笑_熊孩子      | funny_kid            | 搞笑   | funny  | 熊孩子  | kid         |
| 10013    | 搞笑_爆笑动物     | funny_pet            | 搞笑   | funny  | 爆笑动物 | pet         |
| 10014    | 搞笑_趣漫       | funny_quman          | 搞笑   | funny  | 趣漫   | quman       |
| 101      | 电影          | movie                | 电影   | movie  |      |             |
| 10101    | 电影_电影混剪     | movie_remix          | 电影   | movie  | 电影混剪 | remix       |
| 10102    | 电影_电影片段     | movie_pianduan       | 电影   | movie  | 电影片段 | pianduan    |
| 10103    | 电影_电影解说     | movie_jieshuo        | 电影   | movie  | 电影解说 | jieshuo     |
| 10104    | 电影_恶搞配音     | movie_egaopeiyyin    | 电影   | movie  | 恶搞配音 | egaopeiyyin |
| 10105    | 电影_电影资讯     | movie_info           | 电影   | movie  | 电影资讯 | info        |
| 10106    | 电影_预告花絮     | movie_preview        | 电影   | movie  | 预告花絮 | preview     |
| 10107    | 电影_微电影      | movie_weidianying    | 电影   | movie  | 微电影  | weidianying |
| 10108    | 电影_电影周边     | movie_other          | 电影   | movie  | 电影周边 | other       |
| 10108001 | 电影_电影周边_电影节 | movie_other_festival | 电影   | movie  | 电影周边 | other       |

| 新分类ID    | 新分类中文名           | 新分类英文名                 | 一级分类 | 一级分类英文 | 二级分类  | 二级分类英文名         |
|----------|------------------|------------------------|------|--------|-------|-----------------|
| 10108002 | 电影_电影周边_影星红毯(获奖) | movie_other_redcarpet  | 电影   | movie  | 电影周边  | other           |
| 102      | 电视剧              | tv                     | 电视剧  | tv     |       |                 |
| 10201    | 电视剧_预告资讯         | tv_domesticpreview     | 电视剧  | tv     | 预告资讯  | domesticpreview |
| 10202    | 电视剧_精彩片段         | tv_domestic            | 电视剧  | tv     | 精彩片段  | domestic        |
| 10203    | 电视剧_电视剧解说        | tv_jieshuo             | 电视剧  | tv     | 电视剧解说 | jieshuo         |
| 10204    | 电视剧_自制混剪         | tv_remix               | 电视剧  | tv     | 自制混剪  | remix           |
| 10205    | 电视剧_恶搞配音         | tv_egaopeiyin          | 电视剧  | tv     | 恶搞配音  | egaopeiyin      |
| 10206    | 电视剧_电视剧周边        | tv_other               | 电视剧  | tv     | 电视剧周边 | other           |
| 103      | 综艺               | show                   | 综艺   | show   |       |                 |
| 10301    | 综艺_晚会盛典          | show_wanhui            | 综艺   | show   | 晚会盛典  | wanhui          |
| 10302    | 综艺_内地综艺          | show_domestic          | 综艺   | show   | 内地综艺  | domestic        |
| 10303    | 综艺_港台综艺          | show_gangaotai         | 综艺   | show   | 港台综艺  | gangaotai       |
| 10304    | 综艺_韩国综艺          | show_korea             | 综艺   | show   | 韩国综艺  | korea           |
| 10305    | 综艺_日本综艺          | show_japan             | 综艺   | show   | 日本综艺  | japan           |
| 10306    | 综艺_欧美综艺          | show_western           | 综艺   | show   | 欧美综艺  | western         |
| 10307    | 综艺_综艺周边          | show_other             | 综艺   | show   | 综艺周边  | other           |
| 104      | 娱乐               | ent                    | 娱乐   | ent    |       |                 |
| 10400    | 娱乐_其他            | ent_other              | 娱乐   | ent    | 其他    | other           |
| 10401    | 娱乐_明星直发          | ent_personal           | 娱乐   | ent    | 明星直发  | personal        |
| 10402    | 娱乐_内地明星          | ent_domestic           | 娱乐   | ent    | 内地明星  | domestic        |
| 10402001 | 娱乐_内地明星_明星资讯     | ent_domestic_news      | 娱乐   | ent    | 内地明星  | domestic        |
| 10402003 | 娱乐_内地明星_明星专访     | ent_domestic_zhuanfang | 娱乐   | ent    | 内地明星  | domestic        |
| 10402004 | 娱乐_内地明星_饭制       | ent_domestic_remix     | 娱乐   | ent    | 内地明星  | domestic        |
| 10402005 | 娱乐_内地明星_明星现场     | ent_domestic_live      | 娱乐   | ent    | 内地明星  | domestic        |
| 10402006 | 娱乐_内地明星          | ent_domestic_other     | 娱乐   | ent    | 内地明星  | domestic        |

| 新分类ID    | 新分类中文名       | 新分类英文名                  | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名   |
|----------|--------------|-------------------------|------|--------|------|-----------|
|          | 星_明星其他       |                         |      |        |      |           |
| 10403    | 娱乐_港台明星      | ent_gangaotai           | 娱乐   | ent    | 港台明星 | gangaotai |
| 10403001 | 娱乐_港台明星_明星资讯 | ent_gangaotai_news      | 娱乐   | ent    | 港台明星 | gangaotai |
| 10403003 | 娱乐_港台明星_明星专访 | ent_gangaotai_zhuanfang | 娱乐   | ent    | 港台明星 | gangaotai |
| 10403004 | 娱乐_港台明星_饭制   | ent_gangaotai_remix     | 娱乐   | ent    | 港台明星 | gangaotai |
| 10403005 | 娱乐_港台明星_明星现场 | ent_gangaotai_live      | 娱乐   | ent    | 港台明星 | gangaotai |
| 10403006 | 娱乐_港台明星_明星其他 | ent_gangaotai_other     | 娱乐   | ent    | 港台明星 | gangaotai |
| 10404    | 娱乐_日本明星      | ent_japan               | 娱乐   | ent    | 日本明星 | japan     |
| 10404001 | 娱乐_日本明星_明星资讯 | ent_japan_news          | 娱乐   | ent    | 日本明星 | japan     |
| 10404003 | 娱乐_日本明星_明星专访 | ent_japan_zhuanfang     | 娱乐   | ent    | 日本明星 | japan     |
| 10404004 | 娱乐_日本明星_饭制   | ent_japan_remix         | 娱乐   | ent    | 日本明星 | japan     |
| 10404005 | 娱乐_日本明星_明星现场 | ent_japan_live          | 娱乐   | ent    | 日本明星 | japan     |
| 10404006 | 娱乐_日本明星_明星其他 | ent_japan_other         | 娱乐   | ent    | 日本明星 | japan     |
| 10405    | 娱乐_韩国明星      | ent_korea               | 娱乐   | ent    | 韩国明星 | korea     |
| 10405001 | 娱乐_韩国明星_明星资讯 | ent_korea_news          | 娱乐   | ent    | 韩国明星 | korea     |
| 10405003 | 娱乐_韩国明星_明星专访 | ent_korea_zhuanfang     | 娱乐   | ent    | 韩国明星 | korea     |
| 10405004 | 娱乐_韩国明星_饭制   | ent_korea_remix         | 娱乐   | ent    | 韩国明星 | korea     |
| 10405005 | 娱乐_韩国明星_明星现场 | ent_korea_live          | 娱乐   | ent    | 韩国明星 | korea     |
| 10405006 | 娱乐_韩国明星_明星其他 | ent_korea_other         | 娱乐   | ent    | 韩国明星 | korea     |
| 10406    | 娱乐_欧美明星      | ent_western             | 娱乐   | ent    | 欧美明星 | western   |
| 10406001 | 娱乐_欧美明星_明星资讯 | ent_western_news        | 娱乐   | ent    | 欧美明星 | western   |
| 10406003 | 娱乐_欧美明星_明星专访 | ent_western_zhuanfang   | 娱乐   | ent    | 欧美明星 | western   |
| 10406004 | 娱乐_欧美明星_饭制   | ent_western_remix       | 娱乐   | ent    | 欧美明星 | western   |
| 10406005 | 娱乐_欧美明星_明星现场 | ent_western_live        | 娱乐   | ent    | 欧美明星 | western   |

| 新分类ID    | 新分类中文名         | 新分类英文名                    | 一级分类 | 一级分类英文 | 二级分类   | 二级分类英文名   |
|----------|----------------|---------------------------|------|--------|--------|-----------|
| 10406006 | 娱乐_欧美明星_明星其他   | ent_western_other         | 娱乐   | ent    | 欧美明星   | western   |
| 10407    | 娱乐_其他地区明星      | ent_otherstar             | 娱乐   | ent    | 其他地区明星 | otherstar |
| 10407001 | 娱乐_其他地区明星_明星资讯 | ent_otherstar_news        | 娱乐   | ent    | 其他地区明星 | otherstar |
| 10407003 | 娱乐_其他地区明星_明星专访 | ent_otherstar_zhuanfang   | 娱乐   | ent    | 其他地区明星 | otherstar |
| 10407004 | 娱乐_其他地区明星_饭制   | ent_otherstar_remix       | 娱乐   | ent    | 其他地区明星 | otherstar |
| 10407005 | 娱乐_其他地区明星_明星现场 | ent_otherstar_live        | 娱乐   | ent    | 其他地区明星 | otherstar |
| 10407006 | 娱乐_其他地区明星_明星其他 | ent_otherstar_other       | 娱乐   | ent    | 其他地区明星 | otherstar |
| 10408    | 娱乐_八卦          | ent_bagua                 | 娱乐   | ent    | 八卦     | bagua     |
| 105      | 网红达人           | talent                    | 网红达人 | talent |        |           |
| 10500    | 网红达人_其他        | talent_other              | 网红达人 | talent | 其他     | other     |
| 10501    | 网红达人_才艺技能      | talent_skills             | 网红达人 | talent | 才艺技能   | skills    |
| 10501000 | 网红达人_才艺技能_其他   | talent_skills_other       | 网红达人 | talent | 才艺技能   | skills    |
| 10501001 | 网红达人_才艺技能_魔术   | talent_skills_magic       | 网红达人 | talent | 才艺技能   | skills    |
| 10501002 | 网红达人_才艺技能_转笔   | talent_skills_zhuanbi     | 网红达人 | talent | 才艺技能   | skills    |
| 10501003 | 网红达人_才艺技能_魔方   | talent_skills_cube        | 网红达人 | talent | 才艺技能   | skills    |
| 10501004 | 网红达人_才艺技能_模仿秀  | talent_skills_copy        | 网红达人 | talent | 才艺技能   | skills    |
| 10501005 | 网红达人_才艺技能_配音秀  | talent_skills_cast        | 网红达人 | talent | 才艺技能   | skills    |
| 10501006 | 网红达人_才艺技能_翻唱   | talent_skills_sing        | 网红达人 | talent | 才艺技能   | skills    |
| 10501007 | 网红达人_才艺技能_手势舞  | talent_skills_fingerdance | 网红达人 | talent | 才艺技能   | skills    |
| 10501008 | 网红达人_才艺技能_演奏   | talent_skills_play        | 网红达人 | talent | 才艺技能   | skills    |
| 10501009 | 网红达人_才艺技能_绘画   | talent_skills_draw        | 网红达人 | talent | 才艺技能   | skills    |
| 10501010 | 网红达人_才艺技能_书法   | talent_skills_handwriting | 网红达人 | talent | 才艺技能   | skills    |

| 新分类ID    | 新分类中文名             | 新分类英文名                        | 一级分类 | 一级分类英文 | 二级分类       | 二级分类英文名   |
|----------|--------------------|-------------------------------|------|--------|------------|-----------|
| 10501011 | 网红达人_才艺技能_杂技       | talent_skills_acrobatics      | 网红达人 | talent | 才艺技能       | skills    |
| 10502    | 网红达人_舞蹈            | talent_dance                  | 网红达人 | talent | 舞蹈         | dance     |
| 10502000 | 网红达人_舞蹈_其他         | talent_dance_other            | 网红达人 | talent | 舞蹈         | dance     |
| 10502001 | 网红达人_舞蹈_舞蹈展示       | talent_dance_show             | 网红达人 | talent | 舞蹈         | dance     |
| 10502002 | 网红达人_舞蹈_舞蹈教学       | talent_dance_teaching         | 网红达人 | talent | 舞蹈         | dance     |
| 10503    | 网红达人_美女            | talent_beauty                 | 网红达人 | talent | 美女         | beauty    |
| 10504    | 网红达人_创意拍摄          | talent_creative               | 网红达人 | talent | 创意拍摄       | creative  |
| 10504000 | 网红达人_创意拍摄_其他       | talent_creative_other         | 网红达人 | talent | 创意拍摄       | creative  |
| 10504001 | 网红达人_创意拍摄_运镜流      | talent_creative_yunjing       | 网红达人 | talent | 创意拍摄       | creative  |
| 10504002 | 网红达人_创意拍摄_婚纱照      | talent_creative_weddingphotos | 网红达人 | talent | 创意拍摄       | creative  |
| 10504003 | 网红达人_创意拍摄_特效流      | talent_creative_effect        | 网红达人 | talent | 创意拍摄       | creative  |
| 10505    | 网红达人_原创短剧          | talent_film                   | 网红达人 | talent | 原创短剧       | film      |
| 10505000 | 网红达人_原创短剧_其他       | talent_film_other             | 网红达人 | talent | 原创短剧       | film      |
| 10505001 | 网红达人_原创短剧_自制短剧     | talent_film_zizhi             | 网红达人 | talent | 原创短剧       | film      |
| 10505002 | 网红达人_原创短剧_reaction | talent_film_reaction          | 网红达人 | talent | 原创短剧       | film      |
| 10506    | 网红达人_网红资讯          | talent_news                   | 网红达人 | talent | 网红资讯       | news      |
| 106      | 音乐                 | music                         | 音乐   | music  |            |           |
| 10600    | 音乐_其他              | music_other                   | 音乐   | music  | 其他         | other     |
| 10601    | 音乐_经典              | music_classical               | 音乐   | music  | 经典         | classical |
| 10602    | 音乐_网络歌曲            | music_online                  | 音乐   | music  | 网络歌曲       | online    |
| 10603    | 音乐_b-box<br>蒸汽波音乐  | music_b-box                   | 音乐   | music  | b-box蒸汽波音乐 | b-box     |
| 10604    | 音乐_影视              | music_film                    | 音乐   | music  | 影视         | film      |
| 10605    | 音乐_KTV热歌           | music_ktv                     | 音乐   | music  | KTV热歌      | ktv       |

| 新分类ID    | 新分类中文名           | 新分类英文名                 | 一级分类 | 一级分类英文  | 二级分类 | 二级分类英文名        |
|----------|------------------|------------------------|------|---------|------|----------------|
| 10606    | 音乐_儿歌            | music_children         | 音乐   | music   | 儿歌   | children       |
| 10607    | 音乐_古风            | music_antique          | 音乐   | music   | 古风   | antique        |
| 10608    | 音乐_情歌            | music_love             | 音乐   | music   | 情歌   | love           |
| 10609    | 音乐_城市            | music_urban            | 音乐   | music   | 城市   | urban          |
| 10610    | 音乐_现场音乐          | music_live             | 音乐   | music   | 现场音乐 | live           |
| 10611    | 音乐_背景音乐          | music_background       | 音乐   | music   | 背景音乐 | background     |
| 10612    | 音乐_UP主           | music_up               | 音乐   | music   | UP主  | up             |
| 10613    | 音乐_乐器            | music_instrument       | 音乐   | music   | 乐器   | instrument     |
| 10614    | 音乐_DJ            | music_dj               | 音乐   | music   | DJ   | dj             |
| 107      | 曲艺               | folkart                | 曲艺   | folkart |      |                |
| 10700    | 曲艺_其他            | folkart_other          | 曲艺   | folkart | 其他   | other          |
| 10701    | 曲艺_相声            | folkart_crosstalk      | 曲艺   | folkart | 相声   | crosstalk      |
| 10702    | 曲艺_小品            | folkart_sketch         | 曲艺   | folkart | 小品   | sketch         |
| 10703    | 曲艺_评书            | folkart_storytelling   | 曲艺   | folkart | 评书   | storytelling   |
| 10704    | 曲艺_京剧            | folkart_beijingopera   | 曲艺   | folkart | 京剧   | beijingopera   |
| 10705    | 曲艺_越剧            | folkart_shaoxingopera  | 曲艺   | folkart | 越剧   | shaoxingopera  |
| 10706    | 曲艺_秦腔            | folkart_shanxiopera    | 曲艺   | folkart | 秦腔   | shanxiopera    |
| 10707    | 曲艺_昆曲            | folkart_kunquopera     | 曲艺   | folkart | 昆曲   | kunquopera     |
| 10708    | 曲艺_豫剧            | folkart_henanopera     | 曲艺   | folkart | 豫剧   | henanopera     |
| 10709    | 曲艺_粤剧            | folkart_guangdongopera | 曲艺   | folkart | 粤剧   | guangdongopera |
| 10710    | 曲艺_潮剧            | folkart_chaozhouopera  | 曲艺   | folkart | 潮剧   | chaozhouopera  |
| 10711    | 曲艺_黄梅戏           | folkart_huangmeiopera  | 曲艺   | folkart | 黄梅戏  | huangmeiopera  |
| 108      | 游戏               | game                   | 游戏   | game    |      |                |
| 10800    | 游戏_其他            | game_other             | 游戏   | game    | 其他   | other          |
| 10801    | 游戏_主机游戏          | game_box               | 游戏   | game    | 主机游戏 | box            |
| 10801000 | 游戏_主机游戏_其他       | game_box_other         | 游戏   | game    | 主机游戏 | box            |
| 10801001 | 游戏_主机游戏_PS主机游戏   | game_box_ps            | 游戏   | game    | 主机游戏 | box            |
| 10801002 | 游戏_主机游戏_Xbox主机游戏 | game_box_xbox          | 游戏   | game    | 主机游戏 | box            |
| 10801003 | 游戏_主机游戏_Wii主机游戏  | game_box_wii           | 游戏   | game    | 主机游戏 | box            |
| 10801004 | 游戏_主机游戏_NS主机游戏   | game_box_ns            | 游戏   | game    | 主机游戏 | box            |

| 新分类ID    | 新分类中文名       | 新分类英文名              | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名 |
|----------|--------------|---------------------|------|--------|------|---------|
| 10802    | 游戏_端游        | game_pc             | 游戏   | game   | 端游   | pc      |
| 10802000 | 游戏_端游_其他     | game_pc_other       | 游戏   | game   | 端游   | pc      |
| 10802001 | 游戏_端游_RPG    | game_pc_rpg         | 游戏   | game   | 端游   | pc      |
| 10802002 | 游戏_端游_MMORPG | game_pc_mmorpg      | 游戏   | game   | 端游   | pc      |
| 10802003 | 游戏_端游_射击     | game_pc_shooting    | 游戏   | game   | 端游   | pc      |
| 10802004 | 游戏_端游_MOBA   | game_pc_moba        | 游戏   | game   | 端游   | pc      |
| 10802005 | 游戏_端游_养成     | game_pc_raising     | 游戏   | game   | 端游   | pc      |
| 10802006 | 游戏_端游_单机     | game_pc_offline     | 游戏   | game   | 端游   | pc      |
| 10802007 | 游戏_端游_音乐     | game_pc_music       | 游戏   | game   | 端游   | pc      |
| 10802008 | 游戏_端游_竞速     | game_pc_racing      | 游戏   | game   | 端游   | pc      |
| 10802009 | 游戏_端游_策略     | game_pc_strategy    | 游戏   | game   | 端游   | pc      |
| 10802010 | 游戏_端游_棋牌     | game_pc_chess       | 游戏   | game   | 端游   | pc      |
| 10802011 | 游戏_端游_体育     | game_pc_sports      | 游戏   | game   | 端游   | pc      |
| 10802012 | 游戏_端游_回合     | game_pc_round       | 游戏   | game   | 端游   | pc      |
| 10802013 | 游戏_端游_休闲     | game_pc_casual      | 游戏   | game   | 端游   | pc      |
| 10802014 | 游戏_端游_沙盒     | game_pc_sandbox     | 游戏   | game   | 端游   | pc      |
| 10802015 | 游戏_端游_恐怖     | game_pc_horror      | 游戏   | game   | 端游   | pc      |
| 10803    | 游戏_页游        | game_web            | 游戏   | game   | 页游   | web     |
| 10803000 | 游戏_页游_其他     | game_web_other      | 游戏   | game   | 页游   | web     |
| 10803001 | 游戏_页游_RPG    | game_web_rpg        | 游戏   | game   | 页游   | web     |
| 10803002 | 游戏_页游_养成     | game_web_raising    | 游戏   | game   | 页游   | web     |
| 10803003 | 游戏_页游_竞速     | game_web_racing     | 游戏   | game   | 页游   | web     |
| 10803004 | 游戏_页游_模拟经营   | game_web_simulation | 游戏   | game   | 页游   | web     |
| 10803005 | 游戏_页游_益智     | game_web_puzzle     | 游戏   | game   | 页游   | web     |



| 新分类ID    | 新分类中文名       | 新分类英文名                     | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名 |
|----------|--------------|----------------------------|------|--------|------|---------|
| 10804    | 游戏_手游        | game_mobile                | 游戏   | game   | 手游   | mobile  |
| 10804000 | 游戏_手游_其他     | game_mobile_other          | 游戏   | game   | 手游   | mobile  |
| 10804001 | 游戏_手游_其他RPG  | game_mobile_rpg            | 游戏   | game   | 手游   | mobile  |
| 10804002 | 游戏_手游_ARPG   | game_mobile_arpg           | 游戏   | game   | 手游   | mobile  |
| 10804003 | 游戏_手游_MMORPG | game_mobile_mmorpg         | 游戏   | game   | 手游   | mobile  |
| 10804004 | 游戏_手游_回合RPG  | game_mobile_roundrpg       | 游戏   | game   | 手游   | mobile  |
| 10804005 | 游戏_手游_卡牌RPG  | game_mobile_cardrpg        | 游戏   | game   | 手游   | mobile  |
| 10804006 | 游戏_手游_COC    | game_mobile_coc            | 游戏   | game   | 手游   | mobile  |
| 10804007 | 游戏_手游_MOBA   | game_mobile_moba           | 游戏   | game   | 手游   | mobile  |
| 10804008 | 游戏_手游_模拟经营   | game_mobile_simulation     | 游戏   | game   | 手游   | mobile  |
| 10804009 | 游戏_手游_塔防     | game_mobile_towerdefense   | 游戏   | game   | 手游   | mobile  |
| 10804010 | 游戏_手游_对战卡牌策略 | game_mobile_cardstrategy   | 游戏   | game   | 手游   | mobile  |
| 10804011 | 游戏_手游_战争策略   | game_mobile_warstrategy    | 游戏   | game   | 手游   | mobile  |
| 10804012 | 游戏_手游_其他策略游戏 | game_mobile_strategy       | 游戏   | game   | 手游   | mobile  |
| 10804013 | 游戏_手游_飞行射击   | game_mobile_flightshooting | 游戏   | game   | 手游   | mobile  |
| 10804014 | 游戏_手游_枪战射击   | game_mobile_gunshooting    | 游戏   | game   | 手游   | mobile  |
| 10804015 | 游戏_手游_竞速     | game_mobile_racing         | 游戏   | game   | 手游   | mobile  |
| 10804016 | 游戏_手游_体育     | game_mobile_sports         | 游戏   | game   | 手游   | mobile  |
| 10804017 | 游戏_手游_棋牌     | game_mobile_chess          | 游戏   | game   | 手游   | mobile  |
| 10804018 | 游戏_手游_桌游     | game_mobile_board          | 游戏   | game   | 手游   | mobile  |
| 10804019 | 游戏_手游_跑酷躲避   | game_mobile_parkour        | 游戏   | game   | 手游   | mobile  |
| 10804020 | 游戏_手游_消除     | game_mobile_eliminate      | 游戏   | game   | 手游   | mobile  |
| 10804021 | 游戏_手游_休闲益智类  | game_mobile_casual         | 游戏   | game   | 手游   | mobile  |
| 10804022 | 游戏_手游_音乐舞蹈   | game_mobile_music          | 游戏   | game   | 手游   | mobile  |

| 新分类ID    | 新分类中文名           | 新分类英文名                     | 一级分类 | 一级分类英文 | 二级分类  | 二级分类英文名  |
|----------|------------------|----------------------------|------|--------|-------|----------|
| 10805    | 游戏_街机游戏          | game_arcade                | 游戏   | game   | 街机游戏  | arcade   |
| 10806    | 游戏_便携游戏          | game_portable              | 游戏   | game   | 便携游戏  | portable |
| 10806000 | 游戏_便携游戏_其他       | game_portable_other        | 游戏   | game   | 便携游戏  | portable |
| 10806001 | 游戏_便携游戏_PSP游戏    | game_portable_psp          | 游戏   | game   | 便携游戏  | portable |
| 10806002 | 游戏_便携游戏_NDS游戏    | game_portable_nds          | 游戏   | game   | 便携游戏  | portable |
| 10807    | 游戏_小游戏           | game_mini                  | 游戏   | game   | 小游戏   | mini     |
| 10807000 | 游戏_小游戏_其他        | game_mini_other            | 游戏   | game   | 小游戏   | mini     |
| 10807001 | 游戏_小游戏_微信小游戏     | game_mini_wechat           | 游戏   | game   | 小游戏   | mini     |
| 10807002 | 游戏_小游戏_QQ小游戏     | game_mini_qq               | 游戏   | game   | 小游戏   | mini     |
| 10807003 | 游戏_小游戏_H5小游戏     | game_mini_h5               | 游戏   | game   | 小游戏   | mini     |
| 10808    | 游戏_电竞赛事          | game_esports               | 游戏   | game   | 电竞赛事  | esports  |
| 10808000 | 游戏_电竞赛事_其他       | game_esports_other         | 游戏   | game   | 电竞赛事  | esports  |
| 10808001 | 游戏_电竞赛事_LOL赛事    | game_esports_lol           | 游戏   | game   | 电竞赛事  | esports  |
| 10808002 | 游戏_电竞赛事_qq飞车手游赛事 | game_esports_qqspeed       | 游戏   | game   | 电竞赛事  | esports  |
| 10808003 | 游戏_电竞赛事_dota2赛事  | game_esports_dota2         | 游戏   | game   | 电竞赛事  | esports  |
| 10808004 | 游戏_电竞赛事_cf赛事     | game_esports_cf            | 游戏   | game   | 电竞赛事  | esports  |
| 10808005 | 游戏_电竞赛事_王者荣耀赛事   | game_esports_qqpvp         | 游戏   | game   | 电竞赛事  | esports  |
| 10808006 | 游戏_电竞赛事_绝地求生赛事   | game_esports_battlegrounds | 游戏   | game   | 电竞赛事  | esports  |
| 10809    | 游戏_游戏展会          | game_exh                   | 游戏   | game   | 游戏展会  | exh      |
| 10809000 | 游戏_游戏展会_其他       | game_exh_other             | 游戏   | game   | 游戏展会  | exh      |
| 10809001 | 游戏_游戏展会_chinajoy | game_exh_chinajoy          | 游戏   | game   | 游戏展会  | exh      |
| 10810    | 游戏_游戏泛资讯         | game_news                  | 游戏   | game   | 游戏泛资讯 | news     |

| 新分类ID    | 新分类中文名          | 新分类英文名               | 一级分类 | 一级分类英文 | 二级分类  | 二级分类英文名 |
|----------|-----------------|----------------------|------|--------|-------|---------|
| 10810000 | 游戏_游戏泛资讯_其他     | game_news_other      | 游戏   | game   | 游戏泛资讯 | news    |
| 10810001 | 游戏_游戏泛资讯_游戏玩家新闻 | game_news_player     | 游戏   | game   | 游戏泛资讯 | news    |
| 10810002 | 游戏_游戏泛资讯_游戏产业新闻 | game_news_industry   | 游戏   | game   | 游戏泛资讯 | news    |
| 109      | 动漫              | comic                | 动漫   | comic  |       |         |
| 10900    | 动漫_其他           | comic_other          | 动漫   | comic  | 其他    | other   |
| 10901    | 动漫_国产动漫         | comic_china          | 动漫   | comic  | 国产动漫  | china   |
| 10901000 | 动漫_国产动漫_其他      | comic_china_other    | 动漫   | comic  | 国产动漫  | china   |
| 10901001 | 动漫_国产动漫_热血战斗    | comic_china_hotblood | 动漫   | comic  | 国产动漫  | china   |
| 10901002 | 动漫_国产动漫_恶搞爆笑    | comic_china_funny    | 动漫   | comic  | 国产动漫  | china   |
| 10901003 | 动漫_国产动漫_生活校园    | comic_china_school   | 动漫   | comic  | 国产动漫  | china   |
| 10901004 | 动漫_国产动漫_恋爱后宫    | comic_china_love     | 动漫   | comic  | 国产动漫  | china   |
| 10901005 | 动漫_国产动漫_恐怖悬疑    | comic_china_horror   | 动漫   | comic  | 国产动漫  | china   |
| 10901006 | 动漫_国产动漫_魔幻玄幻    | comic_china_magic    | 动漫   | comic  | 国产动漫  | china   |
| 10901007 | 动漫_国产动漫_耽美百合    | comic_china_tanbi    | 动漫   | comic  | 国产动漫  | china   |
| 10901008 | 动漫_国产动漫_治愈催泪    | comic_china_healing  | 动漫   | comic  | 国产动漫  | china   |
| 10901009 | 动漫_国产动漫_运动竞技    | comic_china_sports   | 动漫   | comic  | 国产动漫  | china   |
| 10901010 | 动漫_国产动漫_游戏改编    | comic_china_game     | 动漫   | comic  | 国产动漫  | china   |
| 10902    | 动漫_日本动漫         | comic_japan          | 动漫   | comic  | 日本动漫  | japan   |
| 10902000 | 动漫_日本动漫_其他      | comic_japan_other    | 动漫   | comic  | 日本动漫  | japan   |
| 10902001 | 动漫_日本动漫_热血战斗    | comic_japan_hotblood | 动漫   | comic  | 日本动漫  | japan   |
| 10902002 | 动漫_日本动漫_恶搞爆笑    | comic_japan_funny    | 动漫   | comic  | 日本动漫  | japan   |
| 10902003 | 动漫_日本动漫_生活校园    | comic_japan_school   | 动漫   | comic  | 日本动漫  | japan   |
| 10902004 | 动漫_日本动漫_恋爱后宫    | comic_japan_love     | 动漫   | comic  | 日本动漫  | japan   |
| 10902005 | 动漫_日本动          | comic_japan_horror   | 动漫   | comic  | 日本动漫  | japan   |

| 新分类ID    | 新分类中文名        | 新分类英文名               | 一级分类 | 一级分类英文 | 二级分类       | 二级分类英文名     |
|----------|---------------|----------------------|------|--------|------------|-------------|
|          | 漫_恐怖悬疑        |                      |      |        |            |             |
| 10902006 | 动漫_日本动漫_魔幻玄幻  | comic_japan_magic    | 动漫   | comic  | 日本动漫       | japan       |
| 10902007 | 动漫_日本动漫_耽美百合  | comic_japan_tanbi    | 动漫   | comic  | 日本动漫       | japan       |
| 10902008 | 动漫_日本动漫_治愈催泪  | comic_japan_healing  | 动漫   | comic  | 日本动漫       | japan       |
| 10902009 | 动漫_日本动漫_运动竞技  | comic_japan_sports   | 动漫   | comic  | 日本动漫       | japan       |
| 10903    | 动漫_欧美动漫       | comic_western        | 动漫   | comic  | 欧美动漫       | western     |
| 10903000 | 动漫_欧美动漫_其他    | comic_western_other  | 动漫   | comic  | 欧美动漫       | western     |
| 10903001 | 动漫_欧美动漫_漫威动漫  | comic_western_marvel | 动漫   | comic  | 欧美动漫       | western     |
| 10903002 | 动漫_欧美动漫_DC动漫  | comic_western_dc     | 动漫   | comic  | 欧美动漫       | western     |
| 10903003 | 动漫_欧美动漫_迪士尼动漫 | comic_western_disney | 动漫   | comic  | 欧美动漫       | western     |
| 10904    | 动漫_韩国动漫       | comic_korea          | 动漫   | comic  | 韩国动漫       | korea       |
| 10905    | 动漫_少儿动漫       | comic_children       | 动漫   | comic  | 少儿动漫       | children    |
| 10906    | 动漫_成人动漫       | comic_adult          | 动漫   | comic  | 成人动漫       | adult       |
| 10907    | 动漫_虚拟偶像       | comic_virtual        | 动漫   | comic  | 虚拟偶像       | virtual     |
| 10908    | 动漫_原创短片       | comic_original       | 动漫   | comic  | 原创短片       | original    |
| 10909    | 动漫_盘点解读       | comic_pandian        | 动漫   | comic  | 盘点解读       | pandian     |
| 10910    | 动漫_声优资讯       | comic_cast           | 动漫   | comic  | 声优资讯       | cast        |
| 10911    | 动漫_漫展资讯       | comic_exhnews        | 动漫   | comic  | 漫展资讯       | exhnews     |
| 10912    | 动漫_动漫壁纸       | comic_wallpaper      | 动漫   | comic  | 动漫壁纸       | wallpaper   |
| 10913    | 动漫_cosplay    | comic_cosplay        | 动漫   | comic  | cosplay    | cosplay     |
| 10914    | 动漫_鬼畜         | comic_remix          | 动漫   | comic  | 鬼畜         | remix       |
| 10915    | 动漫_宅舞         | comic_zhaiwu         | 动漫   | comic  | 宅舞         | zhaiwu      |
| 10916    | 动漫_手绘         | comic_handdrawing    | 动漫   | comic  | 手绘         | handdrawing |
| 10917    | 动漫_MMD        | comic_mmd            | 动漫   | comic  | MMD        | mmd         |
| 10918    | 动漫_AMV/MAD/手书 | comic_awv            | 动漫   | comic  | AMV/MAD/手书 | awv         |

| 新分类ID    | 新分类中文名        | 新分类英文名                             | 一级分类 | 一级分类英文 | 二级分类   | 二级分类英文名         |
|----------|---------------|------------------------------------|------|--------|--------|-----------------|
| 10919    | 动漫_动漫音乐       | comic_music                        | 动漫   | comic  | 动漫音乐   | music           |
| 10920    | 动漫_动漫相关资讯     | comic_news                         | 动漫   | comic  | 动漫相关资讯 | news            |
| 10922    | 动漫_手办         | comic_shouban                      | 动漫   | comic  | 手办     | shouban         |
| 110      | 体育            | sports                             | 体育   | sports |        |                 |
| 11000    | 体育_其他         | sports_other                       | 体育   | sports | 其他     | other           |
| 11001    | 体育_NBA        | sports_nba                         | 体育   | sports | NBA    | nba             |
| 11001001 | 体育_NBA_赛事战报   | sports_nba_report                  | 体育   | sports | NBA    | nba             |
| 11001002 | 体育_NBA_球队球员动态 | sports_nba_dynamics                | 体育   | sports | NBA    | nba             |
| 11001003 | 体育_NBA_数据评析   | sports_nba_dataanalysis            | 体育   | sports | NBA    | nba             |
| 11001004 | 体育_NBA_场外花絮   | sports_nba_gossip                  | 体育   | sports | NBA    | nba             |
| 11002    | 体育_CBA        | sports_cba                         | 体育   | sports | CBA    | cba             |
| 11002001 | 体育_CBA_赛事战报   | sports_cba_report                  | 体育   | sports | CBA    | cba             |
| 11002002 | 体育_CBA_球队球员动态 | sports_cba_dynamics                | 体育   | sports | CBA    | cba             |
| 11002003 | 体育_CBA_数据评析   | sports_cba_dataanalysis            | 体育   | sports | CBA    | cba             |
| 11002004 | 体育_CBA_场外花絮   | sports_cba_gossip                  | 体育   | sports | CBA    | cba             |
| 11003    | 体育_篮球其他       | sports_basketballother             | 体育   | sports | 篮球其他   | basketballother |
| 11003001 | 体育_篮球其他_NCAA  | sports_basketballother_ncaa        | 体育   | sports | 篮球其他   | basketballother |
| 11003002 | 体育_篮球其他_CUBA  | sports_basketballother_cuba        | 体育   | sports | 篮球其他   | basketballother |
| 11003003 | 体育_篮球其他_男篮世界杯 | sports_basketballother_worldcup    | 体育   | sports | 篮球其他   | basketballother |
| 11003004 | 体育_篮球其他_男篮亚洲杯 | sports_basketballother_aisacup     | 体育   | sports | 篮球其他   | basketballother |
| 11003005 | 体育_篮球其他_中国男篮  | sports_basketballother_chinamens   | 体育   | sports | 篮球其他   | basketballother |
| 11003006 | 体育_篮球其他_欧锦赛   | sports_basketballother_europecup   | 体育   | sports | 篮球其他   | basketballother |
| 11003007 | 体育_篮球其他_篮球教学  | sports_basketballother_teaching    | 体育   | sports | 篮球其他   | basketballother |
| 11003008 | 体育_篮球其他_街球    | sports_basketballother_street      | 体育   | sports | 篮球其他   | basketballother |
| 11003009 | 体育_篮球其        | sports_basketballother_chinawomens | 体育   | sports | 篮球其他   | basketballother |

| 新分类ID    | 新分类中文名       | 新分类英文名                             | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名          |
|----------|--------------|------------------------------------|------|--------|------|------------------|
|          | 他_中国女篮       |                                    |      |        |      |                  |
| 11004    | 体育_国际足球      | sports_footballexternal            | 体育   | sports | 国际足球 | footballexternal |
| 11004001 | 体育_国际足球_英超   | sports_footballexternal_yingchao   | 体育   | sports | 国际足球 | footballexternal |
| 11004002 | 体育_国际足球_西甲   | sports_footballexternal_xijia      | 体育   | sports | 国际足球 | footballexternal |
| 11004003 | 体育_国际足球_意甲   | sports_footballexternal_yijia      | 体育   | sports | 国际足球 | footballexternal |
| 11004004 | 体育_国际足球_法甲   | sports_footballexternal_fajia      | 体育   | sports | 国际足球 | footballexternal |
| 11004005 | 体育_国际足球_德甲   | sports_footballexternal_dejia      | 体育   | sports | 国际足球 | footballexternal |
| 11004006 | 体育_国际足球_欧冠   | sports_footballexternal_ucl        | 体育   | sports | 国际足球 | footballexternal |
| 11004007 | 体育_国际足球_欧洲杯  | sports_footballexternal_uec        | 体育   | sports | 国际足球 | footballexternal |
| 11005    | 体育_国内足球      | sports_fooballinternal             | 体育   | sports | 国内足球 | fooballinternal  |
| 11005001 | 体育_国内足球_中超   | sports_fooballinternal_zhongchao   | 体育   | sports | 国内足球 | fooballinternal  |
| 11005002 | 体育_国内足球_中国男足 | sports_fooballinternal_chinamens   | 体育   | sports | 国内足球 | fooballinternal  |
| 11005003 | 体育_国内足球_国青   | sports_fooballinternal_chinayouth  | 体育   | sports | 国内足球 | fooballinternal  |
| 11005004 | 体育_国内足球_亚冠   | sports_fooballinternal_acl         | 体育   | sports | 国内足球 | fooballinternal  |
| 11005005 | 体育_国内足球_中甲   | sports_fooballinternal_zhongjia    | 体育   | sports | 国内足球 | fooballinternal  |
| 11005006 | 体育_国内足球_亚洲杯  | sports_fooballinternal_aac         | 体育   | sports | 国内足球 | fooballinternal  |
| 11005007 | 体育_国内足球_中国女足 | sports_fooballinternal_chinawomens | 体育   | sports | 国内足球 | fooballinternal  |
| 11006    | 体育_足球其他      | sports_footballother               | 体育   | sports | 足球其他 | footballother    |
| 11007    | 体育_综合体育      | sports_comprehensive               | 体育   | sports | 综合体育 | comprehensive    |
| 11007000 | 体育_综合体育_其他   | sports_comprehensive_other         | 体育   | sports | 综合体育 | comprehensive    |
| 11007001 | 体育_综合体育_排球   | sports_comprehensive_volleyball    | 体育   | sports | 综合体育 | comprehensive    |
| 11007002 | 体育_综合体育_乒乓球  | sports_comprehensive_pingpang      | 体育   | sports | 综合体育 | comprehensive    |
| 11007003 | 体育_综合体育_羽毛球  | sports_comprehensive_badmiton      | 体育   | sports | 综合体育 | comprehensive    |
| 11007004 | 体育_综合体育_橄榄球  | sports_comprehensive_rugby         | 体育   | sports | 综合体育 | comprehensive    |

| 新分类ID    | 新分类中文名       | 新分类英文名                                    | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名       |
|----------|--------------|-------------------------------------------|------|--------|------|---------------|
| 11007005 | 体育_综合体育_台球   | sports_comprehensive_taiqiu               | 体育   | sports | 综合体育 | comprehensive |
| 11007006 | 体育_综合体育_射击   | sports_comprehensive_shooting             | 体育   | sports | 综合体育 | comprehensive |
| 11007007 | 体育_综合体育_游泳   | sports_comprehensive_swimming             | 体育   | sports | 综合体育 | comprehensive |
| 11007008 | 体育_综合体育_跳水   | sports_comprehensive_diving               | 体育   | sports | 综合体育 | comprehensive |
| 11007009 | 体育_综合体育_花样游泳 | sports_comprehensive_synchronizedswimming | 体育   | sports | 综合体育 | comprehensive |
| 11007010 | 体育_综合体育_田径   | sports_comprehensive_athletics            | 体育   | sports | 综合体育 | comprehensive |
| 11007011 | 体育_综合体育_体操   | sports_comprehensive_gymnastics           | 体育   | sports | 综合体育 | comprehensive |
| 11008    | 体育_冬季项目      | sports_winter                             | 体育   | sports | 冬季项目 | winter        |
| 11008000 | 体育_冬季项目_其他   | sports_winter_other                       | 体育   | sports | 冬季项目 | winter        |
| 11008001 | 体育_冬季项目_速度滑冰 | sports_winter_speedskating                | 体育   | sports | 冬季项目 | winter        |
| 11008002 | 体育_冬季项目_短道速滑 | sports_winter_shorttrack                  | 体育   | sports | 冬季项目 | winter        |
| 11008003 | 体育_冬季项目_花样滑冰 | sports_winter_figureskating               | 体育   | sports | 冬季项目 | winter        |
| 11008004 | 体育_冬季项目_冰球   | sports_winter_icehockey                   | 体育   | sports | 冬季项目 | winter        |
| 11008005 | 体育_冬季项目_冰壶   | sports_winter_curling                     | 体育   | sports | 冬季项目 | winter        |
| 11008006 | 体育_冬季项目_滑雪   | sports_winter_skiing                      | 体育   | sports | 冬季项目 | winter        |
| 11009    | 体育_大众体育      | sports_popular                            | 体育   | sports | 大众体育 | popular       |
| 11009000 | 体育_大众体育_其他   | sports_popular_other                      | 体育   | sports | 大众体育 | popular       |
| 11009001 | 体育_大众体育_跑步   | sports_popular_paobu                      | 体育   | sports | 大众体育 | popular       |
| 11009002 | 体育_大众体育_健身   | sports_popular_fitness                    | 体育   | sports | 大众体育 | popular       |
| 11009003 | 体育_大众体育_竞技舞蹈 | sports_popular_competitivedance           | 体育   | sports | 大众体育 | popular       |
| 11009004 | 体育_大众体育_健身操  | sports_popular_aerobics                   | 体育   | sports | 大众体育 | popular       |
| 11009005 | 体育_大众体育_钓鱼   | sports_popular_competitivefishing         | 体育   | sports | 大众体育 | popular       |
| 11009006 | 体育_大众体育_竞技街舞 | sports_popular_competitivestreetdance     | 体育   | sports | 大众体育 | popular       |
| 11010    | 体育_户外运       | sports_outdoor                            | 体育   | sports | 户外运动 | outdoor       |

| 新分类ID    | 新分类中文名       | 新分类英文名                        | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名 |
|----------|--------------|-------------------------------|------|--------|------|---------|
|          | 动            |                               |      |        |      |         |
| 11010000 | 体育_户外运动_其他   | sports_outdoor_other          | 体育   | sports | 户外运动 | outdoor |
| 11010001 | 体育_户外运动_攀岩   | sports_outdoor_rockclimbing   | 体育   | sports | 户外运动 | outdoor |
| 11010002 | 体育_户外运动_越野   | sports_outdoor_crosscountry   | 体育   | sports | 户外运动 | outdoor |
| 11010003 | 体育_户外运动_自行车  | sports_outdoor_bicycle        | 体育   | sports | 户外运动 | outdoor |
| 11010004 | 体育_户外运动_登山   | sports_outdoor_mountaineering | 体育   | sports | 户外运动 | outdoor |
| 11010005 | 体育_户外运动_野外生存 | sports_outdoor_wildsurvival   | 体育   | sports | 户外运动 | outdoor |
| 11010006 | 体育_户外运动_速降   | sports_outdoor_downhill       | 体育   | sports | 户外运动 | outdoor |
| 11011    | 体育_极限运动      | sports_extreme                | 体育   | sports | 极限运动 | extreme |
| 11011000 | 体育_极限运动_其他   | sports_extreme_other          | 体育   | sports | 极限运动 | extreme |
| 11011001 | 体育_极限运动_小轮车  | sports_extreme_bmx            | 体育   | sports | 极限运动 | extreme |
| 11011002 | 体育_极限运动_跳伞   | sports_extreme_skydiving      | 体育   | sports | 极限运动 | extreme |
| 11011003 | 体育_极限运动_潜水   | sports_extreme_qianshui       | 体育   | sports | 极限运动 | extreme |
| 11011004 | 体育_极限运动_滑板   | sports_extreme_skate          | 体育   | sports | 极限运动 | extreme |
| 11011005 | 体育_极限运动_速滑   | sports_extreme_speedskating   | 体育   | sports | 极限运动 | extreme |
| 11011006 | 体育_极限运动_冲浪   | sports_extreme_surfing        | 体育   | sports | 极限运动 | extreme |
| 11011007 | 体育_极限运动_滑翔伞  | sports_extreme_paragliding    | 体育   | sports | 极限运动 | extreme |
| 11011008 | 体育_极限运动_跑酷   | sports_extreme_parkour        | 体育   | sports | 极限运动 | extreme |
| 11011009 | 体育_极限运动_蹦极   | sports_extreme_bungee         | 体育   | sports | 极限运动 | extreme |
| 11011010 | 体育_极限运动_轮滑   | sports_extreme_roller         | 体育   | sports | 极限运动 | extreme |
| 11011011 | 体育_极限运动_极限漂流 | sports_extreme_drifting       | 体育   | sports | 极限运动 | extreme |
| 11012    | 体育_大型赛事      | sports_major                  | 体育   | sports | 大型赛事 | major   |
| 11012000 | 体育_大型赛事_其他   | sports_major_other            | 体育   | sports | 大型赛事 | major   |
| 11012001 | 体育_大型赛事_大运会  | sports_major_college          | 体育   | sports | 大型赛事 | major   |



| 新分类ID    | 新分类中文名      | 新分类英文名                    | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名 |
|----------|-------------|---------------------------|------|--------|------|---------|
| 11012002 | 体育_大型赛事_青奥会 | sports_major_youth        | 体育   | sports | 大型赛事 | major   |
| 11012003 | 体育_大型赛事_冬奥会 | sports_major_winter       | 体育   | sports | 大型赛事 | major   |
| 11012004 | 体育_大型赛事_亚运会 | sports_major_asia         | 体育   | sports | 大型赛事 | major   |
| 11012005 | 体育_大型赛事_奥运会 | sports_major_olympic      | 体育   | sports | 大型赛事 | major   |
| 11012006 | 体育_大型赛事_世界杯 | sports_major_worldcup     | 体育   | sports | 大型赛事 | major   |
| 11013    | 体育_搏击       | sports_fight              | 体育   | sports | 搏击   | fight   |
| 11013000 | 体育_搏击_其他    | sports_fight_other        | 体育   | sports | 搏击   | fight   |
| 11013001 | 体育_搏击_拳击    | sports_fight_boxing       | 体育   | sports | 搏击   | fight   |
| 11013002 | 体育_搏击_女子搏击  | sports_fight_woman        | 体育   | sports | 搏击   | fight   |
| 11013003 | 体育_搏击_格斗    | sports_fight_combat       | 体育   | sports | 搏击   | fight   |
| 11013004 | 体育_搏击_摔跤    | sports_fight_wrestling    | 体育   | sports | 搏击   | fight   |
| 11013005 | 体育_搏击_中国功夫  | sports_fight_kungfu       | 体育   | sports | 搏击   | fight   |
| 11013006 | 体育_搏击_摔跤    | sports_fight_prowrestling | 体育   | sports | 搏击   | fight   |
| 11014    | 体育_精英体育     | sports_elite              | 体育   | sports | 精英体育 | elite   |
| 11014000 | 体育_精英体育_其他  | sports_elite_other        | 体育   | sports | 精英体育 | elite   |
| 11014001 | 体育_精英体育_网球  | sports_elite_tennis       | 体育   | sports | 精英体育 | elite   |
| 11014002 | 体育_精英体育_击剑  | sports_elite_fencing      | 体育   | sports | 精英体育 | elite   |
| 11014003 | 体育_精英体育_高尔夫 | sports_elite_golf         | 体育   | sports | 精英体育 | elite   |
| 11014004 | 体育_精英体育_赛车  | sports_elite_racing       | 体育   | sports | 精英体育 | elite   |
| 11014005 | 体育_精英体育_马术  | sports_elite_equestrian   | 体育   | sports | 精英体育 | elite   |
| 11014006 | 体育_精英体育_帆船  | sports_elite_sailing      | 体育   | sports | 精英体育 | elite   |
| 11014007 | 体育_精英体育_棒球  | sports_elite_baseball     | 体育   | sports | 精英体育 | elite   |
| 11015    | 体育_彩票       | sports_lottery            | 体育   | sports | 彩票   | lottery |
| 11015000 | 体育_彩票_其他    | sports_lottery_other      | 体育   | sports | 彩票   | lottery |

| 新分类ID    | 新分类中文名       | 新分类英文名                             | 一级分类 | 一级分类英文   | 二级分类 | 二级分类英文名       |
|----------|--------------|------------------------------------|------|----------|------|---------------|
| 11015001 | 体育_彩票_体彩资讯   | sports_lottery_sportsnews          | 体育   | sports   | 彩票   | lottery       |
| 11015002 | 体育_彩票_体彩竞猜分析 | sports_lottery_sportsanalysis      | 体育   | sports   | 彩票   | lottery       |
| 11015003 | 体育_彩票_福彩资讯   | sports_lottery_welfarenews         | 体育   | sports   | 彩票   | lottery       |
| 11015004 | 体育_彩票_福彩分析   | sports_lottery_welfareanalysis     | 体育   | sports   | 彩票   | lottery       |
| 11016    | 体育_棋牌        | sports_chess                       | 体育   | sports   | 棋牌   | chess         |
| 11016000 | 体育_棋牌_其他     | sports_chess_other                 | 体育   | sports   | 棋牌   | chess         |
| 11016001 | 体育_棋牌_围棋     | sports_chess_go                    | 体育   | sports   | 棋牌   | chess         |
| 11016002 | 体育_棋牌_象棋     | sports_chess_chinesechess          | 体育   | sports   | 棋牌   | chess         |
| 11016003 | 体育_棋牌_麻将     | sports_chess_mahjong               | 体育   | sports   | 棋牌   | chess         |
| 11016004 | 体育_棋牌_桥牌     | sports_chess_bridge                | 体育   | sports   | 棋牌   | chess         |
| 11016005 | 体育_棋牌_扑克     | sports_chess_poker                 | 体育   | sports   | 棋牌   | chess         |
| 11016006 | 体育_棋牌_国际象棋   | sports_chess_chess                 | 体育   | sports   | 棋牌   | chess         |
| 111      | 时事           | politics                           | 时事   | politics |      |               |
| 11100    | 时事_其他        | politics_other                     | 时事   | politics | 其他   | other         |
| 11101    | 时事_国内        | politics_domestic                  | 时事   | politics | 国内   | domestic      |
| 11101000 | 时事_国内_其他     | politics_domestic_other            | 时事   | politics | 国内   | domestic      |
| 11101001 | 时事_国内_国家领导人  | politics_domestic_stateleaders     | 时事   | politics | 国内   | domestic      |
| 11101002 | 时事_国内_反腐新闻   | politics_domestic_corruption       | 时事   | politics | 国内   | domestic      |
| 11101003 | 时事_国内_官员任免   | politics_domestic_appointandremove | 时事   | politics | 国内   | domestic      |
| 11101004 | 时事_国内_台湾     | politics_domestic_taiwan           | 时事   | politics | 国内   | domestic      |
| 11101005 | 时事_国内_港澳     | politics_domestic_gangao           | 时事   | politics | 国内   | domestic      |
| 11101006 | 时事_国内_新时代    | politics_domestic_newage           | 时事   | politics | 国内   | domestic      |
| 11102    | 时事_国际        | politics_international             | 时事   | politics | 国际   | international |
| 11102000 | 时事_国际_其他     | politics_international_other       | 时事   | politics | 国际   | international |
| 11102001 | 时事_国际_美国     | politics_international_usa         | 时事   | politics | 国际   | international |

| 新分类ID    | 新分类中文名       | 新分类英文名                               | 一级分类 | 一级分类英文   | 二级分类 | 二级分类英文名       |
|----------|--------------|--------------------------------------|------|----------|------|---------------|
| 11102003 | 时事_国际_欧洲     | politics_international_europe        | 时事   | politics | 国际   | international |
| 11102004 | 时事_国际_日本     | politics_international_japan         | 时事   | politics | 国际   | international |
| 11102005 | 时事_国际_东南亚    | politics_international_southeastasia | 时事   | politics | 国际   | international |
| 11102006 | 时事_国际_中东     | politics_international_middleeast    | 时事   | politics | 国际   | international |
| 11102007 | 时事_国际_朝鲜     | politics_international_korea         | 时事   | politics | 国际   | international |
| 11102008 | 时事_国际_美洲     | politics_international_america       | 时事   | politics | 国际   | international |
| 11102009 | 时事_国际_非洲     | politics_international_africa        | 时事   | politics | 国际   | international |
| 11102010 | 时事_国际_亚洲     | politics_international_asia          | 时事   | politics | 国际   | international |
| 11102011 | 时事_国际_国际花边   | politics_international_gossip        | 时事   | politics | 国际   | international |
| 11102012 | 时事_国际_国际组织   | politics_international_organizations | 时事   | politics | 国际   | international |
| 11103    | 时事_民生政策      | politics_livelihood                  | 时事   | politics | 民生政策 | livelihood    |
| 11103000 | 时事_民生政策_其他   | politics_livelihood_other            | 时事   | politics | 民生政策 | livelihood    |
| 11103001 | 时事_民生政策_交通   | politics_livelihood_traffic          | 时事   | politics | 民生政策 | livelihood    |
| 11103002 | 时事_民生政策_教育   | politics_livelihood_edu              | 时事   | politics | 民生政策 | livelihood    |
| 11103003 | 时事_民生政策_住房   | politics_livelihood_housing          | 时事   | politics | 民生政策 | livelihood    |
| 11103004 | 时事_民生政策_个税   | politics_livelihood_tax              | 时事   | politics | 民生政策 | livelihood    |
| 11103005 | 时事_民生政策_医疗   | politics_livelihood_medical          | 时事   | politics | 民生政策 | livelihood    |
| 11103006 | 时事_民生政策_社会保障 | politics_livelihood_socialsecurity   | 时事   | politics | 民生政策 | livelihood    |
| 11103007 | 时事_民生政策_法律法规 | politics_livelihood_laws             | 时事   | politics | 民生政策 | livelihood    |
| 11103008 | 时事_民生政策_环保   | politics_livelihood_environment      | 时事   | politics | 民生政策 | livelihood    |
| 11103009 | 时事_民生政策_计划生育 | politics_livelihood_familyplanning   | 时事   | politics | 民生政策 | livelihood    |
| 11103010 | 时事_民生政策_食药安全 | politics_livelihood_fooddrugsafety   | 时事   | politics | 民生政策 | livelihood    |
| 112      | 社会           | social                               | 社会   | social   |      |               |
| 11200    | 社会_其他        | social_other                         | 社会   | social   | 其他   | other         |

| 新分类ID    | 新分类中文名         | 新分类英文名                           | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名          |
|----------|----------------|----------------------------------|------|--------|------|------------------|
| 11201    | 社会_奇闻趣事        | social_interesting               | 社会   | social | 奇闻趣事 | interesting      |
| 11201000 | 社会_奇闻趣事_其他     | social_interesting_other         | 社会   | social | 奇闻趣事 | interesting      |
| 11201001 | 社会_奇闻趣事_奇葩人事   | social_interesting_human         | 社会   | social | 奇闻趣事 | interesting      |
| 11201002 | 社会_奇闻趣事_自然奇观   | social_interesting_nature        | 社会   | social | 奇闻趣事 | interesting      |
| 11201003 | 社会_奇闻趣事_动物奇葩   | social_interesting_animal        | 社会   | social | 奇闻趣事 | interesting      |
| 11201004 | 社会_奇闻趣事_彩票趣闻   | social_interesting_lottery       | 社会   | social | 奇闻趣事 | interesting      |
| 11202    | 社会_法制纠纷        | social_legaldispute              | 社会   | social | 法制纠纷 | legaldispute     |
| 11202000 | 社会_法制纠纷_其他     | social_legaldispute_other        | 社会   | social | 法制纠纷 | legaldispute     |
| 11202001 | 社会_法制纠纷_法制案件   | social_legaldispute_legal        | 社会   | social | 法制纠纷 | legaldispute     |
| 11202002 | 社会_法制纠纷_家庭纠纷   | social_legaldispute_family       | 社会   | social | 法制纠纷 | legaldispute     |
| 11202003 | 社会_法制纠纷_医疗纠纷   | social_legaldispute_medical      | 社会   | social | 法制纠纷 | legaldispute     |
| 11203    | 社会_校园新闻        | social_school                    | 社会   | social | 校园新闻 | school           |
| 11204    | 社会_社区新闻        | social_community                 | 社会   | social | 社区新闻 | community        |
| 11205    | 社会_公益          | social_publicwelfare             | 社会   | social | 公益   | publicwelfare    |
| 11206    | 社会_正能量         | social_positive                  | 社会   | social | 正能量  | positive         |
| 11207    | 社会_事故灾害        | social_accidentdisaster          | 社会   | social | 事故灾害 | accidentdisaster |
| 11207000 | 社会_事故灾害_其他     | social_accidentdisaster_other    | 社会   | social | 事故灾害 | accidentdisaster |
| 11207001 | 社会_事故灾害_自然灾害   | social_accidentdisaster_disaster | 社会   | social | 事故灾害 | accidentdisaster |
| 11207002 | 社会_事故灾害_交通事故   | social_accidentdisaster_accident | 社会   | social | 事故灾害 | accidentdisaster |
| 11207003 | 社会_事故灾害_公共安全事故 | social_accidentdisaster_public   | 社会   | social | 事故灾害 | accidentdisaster |
| 113      | 军事             | mil                              | 军事   | mil    |      |                  |
| 11300    | 军事_其他          | mil_other                        | 军事   | mil    | 其他   | other            |
| 11301    | 军事_武器装备        | mil_equipment                    | 军事   | mil    | 武器装备 | equipment        |
| 11301000 | 军事_武器装备_其他     | mil_equipment_other              | 军事   | mil    | 武器装备 | equipment        |
| 11301001 | 军事_武器装         | mil_equipment_gun                | 军事   | mil    | 武器装备 | equipment        |

| 新分类ID    | 新分类中文名        | 新分类英文名                      | 一级分类 | 一级分类英文 | 二级分类  | 二级分类英文名   |
|----------|---------------|-----------------------------|------|--------|-------|-----------|
|          | 备_枪炮          |                             |      |        |       |           |
| 11301002 | 军事_武器装备_导弹    | mil_equipment_missile       | 军事   | mil    | 武器装备  | equipment |
| 11301003 | 军事_武器装备_舰船舰艇  | mil_equipment_warship       | 军事   | mil    | 武器装备  | equipment |
| 11301004 | 军事_武器装备_坦克装甲车 | mil_equipment_tank          | 军事   | mil    | 武器装备  | equipment |
| 11301005 | 军事_武器装备_爆炸物   | mil_equipment_explosive     | 军事   | mil    | 武器装备  | equipment |
| 11301006 | 军事_武器装备_生化武器  | mil_equipment_biochemical   | 军事   | mil    | 武器装备  | equipment |
| 11301007 | 军事_武器装备_光学武器  | mil_equipment_optical       | 军事   | mil    | 武器装备  | equipment |
| 11302    | 军事_军用飞行器      | mil_aircraft                | 军事   | mil    | 军用飞行器 | aircraft  |
| 11302000 | 军事_军用飞行器_其他   | mil_aircraft_other          | 军事   | mil    | 军用飞行器 | aircraft  |
| 11302001 | 军事_军用飞行器_太空装备 | mil_aircraft_space          | 军事   | mil    | 军用飞行器 | aircraft  |
| 11302002 | 军事_军用飞行器_国内军机 | mil_aircraft_domestic       | 军事   | mil    | 军用飞行器 | aircraft  |
| 11302003 | 军事_军用飞行器_国外军机 | mil_aircraft_foreign        | 军事   | mil    | 军用飞行器 | aircraft  |
| 11303    | 军事_军事历史       | mil_history                 | 军事   | mil    | 军事历史  | history   |
| 11303000 | 军事_军事历史_其他    | mil_history_other           | 军事   | mil    | 军事历史  | history   |
| 11303001 | 军事_军事历史_一战    | mil_history_wwi             | 军事   | mil    | 军事历史  | history   |
| 11303002 | 军事_军事历史_二战    | mil_history_wwii            | 军事   | mil    | 军事历史  | history   |
| 11303003 | 军事_军事历史_抗美援朝  | mil_history_kangmeiyuanchao | 军事   | mil    | 军事历史  | history   |
| 11303004 | 军事_军事历史_抗日战争  | mil_history_antijapanesewar | 军事   | mil    | 军事历史  | history   |
| 11303005 | 军事_军事历史_国共内战  | mil_history_chinesecivilwar | 军事   | mil    | 军事历史  | history   |
| 11303006 | 军事_军事历史_古代战争  | mil_history_ancientwars     | 军事   | mil    | 军事历史  | history   |
| 11303007 | 军事_军事历史_军事人物  | mil_history_militaryfigures | 军事   | mil    | 军事历史  | history   |
| 11303008 | 军事_军事历史_局地冲突  | mil_history_localconflict   | 军事   | mil    | 军事历史  | history   |
| 11303009 | 军事_军事历史       | mil_history_army            | 军事   | mil    | 军事历史  | history   |

| 新分类ID    | 新分类中文名          | 新分类英文名                         | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名      |
|----------|-----------------|--------------------------------|------|--------|------|--------------|
|          | 史_军队史           |                                |      |        |      |              |
| 11304    | 军事_军事军情         | mil_intelligence               | 军事   | mil    | 军事军情 | intelligence |
| 11304000 | 军事_军事军情_其他      | mil_intelligence_other         | 军事   | mil    | 军事军情 | intelligence |
| 11304001 | 军事_军事军情_中国军情    | mil_intelligence_china         | 军事   | mil    | 军事军情 | intelligence |
| 11304002 | 军事_军事军情_朝鲜军情    | mil_intelligence_korea         | 军事   | mil    | 军事军情 | intelligence |
| 11304003 | 军事_军事军情_台湾军情    | mil_intelligence_taiwan        | 军事   | mil    | 军事军情 | intelligence |
| 11304004 | 军事_军事军情_日本军情    | mil_intelligence_japan         | 军事   | mil    | 军事军情 | intelligence |
| 11304005 | 军事_军事军情_美国军情    | mil_intelligence_usa           | 军事   | mil    | 军事军情 | intelligence |
| 11304007 | 军事_军事军情_东南亚军情   | mil_intelligence_southeastasia | 军事   | mil    | 军事军情 | intelligence |
| 11304008 | 军事_军事军情_中东军情    | mil_intelligence_middleeast    | 军事   | mil    | 军事军情 | intelligence |
| 11304009 | 军事_军事军情_南亚军情    | mil_intelligence_southasia     | 军事   | mil    | 军事军情 | intelligence |
| 11304010 | 军事_军事军情_欧洲军情    | mil_intelligence_europe        | 军事   | mil    | 军事军情 | intelligence |
| 11304011 | 军事_军事军情_美洲军情    | mil_intelligence_america       | 军事   | mil    | 军事军情 | intelligence |
| 11304012 | 军事_军事军情_非洲军情    | mil_intelligence_africa        | 军事   | mil    | 军事军情 | intelligence |
| 11305    | 军事_军事专栏         | mil_program                    | 军事   | mil    | 军事专栏 | program      |
| 114      | 科技              | tech                           | 科技   | tech   |      |              |
| 11400    | 科技_其他           | tech_other                     | 科技   | tech   | 其他   | other        |
| 11401    | 科技_数码           | tech_digi                      | 科技   | tech   | 数码   | digi         |
| 11401001 | 科技_数码_手机        | tech_digi_phone                | 科技   | tech   | 数码   | digi         |
| 11401002 | 科技_数码_家电        | tech_digi_jiadian              | 科技   | tech   | 数码   | digi         |
| 11401003 | 科技_数码_游戏设备      | tech_digi_game                 | 科技   | tech   | 数码   | digi         |
| 11401004 | 科技_数码_影像数码      | tech_digi_image                | 科技   | tech   | 数码   | digi         |
| 11401005 | 科技_数码_电脑/笔记本/平板 | tech_digi_computer             | 科技   | tech   | 数码   | digi         |
| 11401006 | 科技_数码_无人机       | tech_digi_uav                  | 科技   | tech   | 数码   | digi         |

| 新分类ID    | 新分类中文名       | 新分类英文名                        | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名 |
|----------|--------------|-------------------------------|------|--------|------|---------|
| 11401007 | 科技_数码_机器人    | tech_digi_robot               | 科技   | tech   | 数码   | digi    |
| 11401008 | 科技_数码_耳机     | tech_digi_headset             | 科技   | tech   | 数码   | digi    |
| 11401009 | 科技_数码_智能音箱   | tech_digi_smartspeaker        | 科技   | tech   | 数码   | digi    |
| 11401010 | 科技_数码_智能穿戴   | tech_digi_smartwear           | 科技   | tech   | 数码   | digi    |
| 11401011 | 科技_数码_其他数码硬件 | tech_digi_other               | 科技   | tech   | 数码   | digi    |
| 11402    | 科技_内容产业      | tech_content                  | 科技   | tech   | 内容产业 | content |
| 11402000 | 科技_内容产业_其他   | tech_content_other            | 科技   | tech   | 内容产业 | content |
| 11402001 | 科技_内容产业_综合视频 | tech_content_longvideo        | 科技   | tech   | 内容产业 | content |
| 11402002 | 科技_内容产业_短视频  | tech_content_shortvideo       | 科技   | tech   | 内容产业 | content |
| 11402003 | 科技_内容产业_直播   | tech_content_live             | 科技   | tech   | 内容产业 | content |
| 11402004 | 科技_内容产业_数字阅读 | tech_content_digireading      | 科技   | tech   | 内容产业 | content |
| 11402005 | 科技_内容产业_资讯   | tech_content_news             | 科技   | tech   | 内容产业 | content |
| 11402006 | 科技_内容产业_知识付费 | tech_content_knowledgepayment | 科技   | tech   | 内容产业 | content |
| 11402007 | 科技_内容产业_数字音乐 | tech_content_digimusic        | 科技   | tech   | 内容产业 | content |
| 11403    | 科技_软件工具      | tech_tools                    | 科技   | tech   | 软件工具 | tools   |
| 11403000 | 科技_软件工具_其他   | tech_tools_other              | 科技   | tech   | 软件工具 | tools   |
| 11403001 | 科技_软件工具_拍照美图 | tech_tools_photograph         | 科技   | tech   | 软件工具 | tools   |
| 11403002 | 科技_软件工具_搜索   | tech_tools_search             | 科技   | tech   | 软件工具 | tools   |
| 11403003 | 科技_软件工具_浏览器  | tech_tools_browser            | 科技   | tech   | 软件工具 | tools   |
| 11403004 | 科技_软件工具_地图   | tech_tools_map                | 科技   | tech   | 软件工具 | tools   |
| 11403005 | 科技_软件工具_翻译   | tech_tools_translate          | 科技   | tech   | 软件工具 | tools   |
| 11403006 | 科技_软件工具_邮箱   | tech_tools_email              | 科技   | tech   | 软件工具 | tools   |
| 11403007 | 科技_软件工具_办公   | tech_tools_office             | 科技   | tech   | 软件工具 | tools   |
| 11403008 | 科技_软件工       | tech_tools_cloudcollaboration | 科技   | tech   | 软件工具 | tools   |

| 新分类ID    | 新分类中文名         | 新分类英文名                          | 一级分类 | 一级分类英文 | 二级分类  | 二级分类英文名      |
|----------|----------------|---------------------------------|------|--------|-------|--------------|
|          | 具_云协作          |                                 |      |        |       |              |
| 11403009 | 科技_软件工具_wifi   | tech_tools_wifi                 | 科技   | tech   | 软件工具  | tools        |
| 11403010 | 科技_软件工具_招聘     | tech_tools_recruitment          | 科技   | tech   | 软件工具  | tools        |
| 11403011 | 科技_软件工具_社交     | tech_tools_social               | 科技   | tech   | 软件工具  | tools        |
| 11403012 | 科技_软件工具_运动健康   | tech_tools_health               | 科技   | tech   | 软件工具  | tools        |
| 11404    | 科技_电商          | tech_ecommerce                  | 科技   | tech   | 电商    | ecommerce    |
| 11404000 | 科技_电商_其他       | tech_ecommerce_other            | 科技   | tech   | 电商    | ecommerce    |
| 11404001 | 科技_电商_综合电商     | tech_ecommerce_integrated       | 科技   | tech   | 电商    | ecommerce    |
| 11404002 | 科技_电商_垂直电商     | tech_ecommerce_vertical         | 科技   | tech   | 电商    | ecommerce    |
| 11404003 | 科技_电商_新零售      | tech_ecommerce_newretail        | 科技   | tech   | 电商    | ecommerce    |
| 11404004 | 科技_电商_物流仓储     | tech_ecommerce_logistics        | 科技   | tech   | 电商    | ecommerce    |
| 11405    | 科技_汽车科技        | tech_automotive                 | 科技   | tech   | 汽车科技  | automotive   |
| 11405000 | 科技_汽车科技_其他     | tech_automotive_other           | 科技   | tech   | 汽车科技  | automotive   |
| 11405001 | 科技_汽车科技_无人驾驶   | tech_automotive_autopilot       | 科技   | tech   | 汽车科技  | automotive   |
| 11405002 | 科技_汽车科技_车联网    | tech_automotive_networking      | 科技   | tech   | 汽车科技  | automotive   |
| 11405003 | 科技_汽车科技_新能源汽车  | tech_automotive_newenergy       | 科技   | tech   | 汽车科技  | automotive   |
| 11406    | 科技_科技大佬        | tech_famous                     | 科技   | tech   | 科技大佬  | famous       |
| 11407    | 科技_互联网金融       | tech_interfinance               | 科技   | tech   | 互联网金融 | interfinance |
| 11407000 | 科技_互联网金融_其他    | tech_interfinance_other         | 科技   | tech   | 互联网金融 | interfinance |
| 11407001 | 科技_互联网金融_P2P   | tech_interfinance_p2p           | 科技   | tech   | 互联网金融 | interfinance |
| 11407002 | 科技_互联网金融_互联网保险 | tech_interfinance_insurance     | 科技   | tech   | 互联网金融 | interfinance |
| 11407004 | 科技_互联网金融_移动支付  | tech_interfinance_mobilepayment | 科技   | tech   | 互联网金融 | interfinance |
| 11408    | 科技_互联网+        | tech_internetplus               | 科技   | tech   | 互联网+  | internetplus |
| 11408000 | 科技_互联网+_其他     | tech_internetplus_other         | 科技   | tech   | 互联网+  | internetplus |



| 新分类ID    | 新分类中文名           | 新分类英文名                               | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名       |
|----------|------------------|--------------------------------------|------|--------|------|---------------|
| 11408001 | 科技_互联网+_生活服务     | tech_internetplus_lifefservice       | 科技   | tech   | 互联网+ | internetplus  |
| 11408002 | 科技_互联网+_家居房产     | tech_internetplus_house              | 科技   | tech   | 互联网+ | internetplus  |
| 11408003 | 科技_互联网+_智慧出行     | tech_internetplus_travel             | 科技   | tech   | 互联网+ | internetplus  |
| 11408004 | 科技_互联网+_医疗       | tech_internetplus_medical            | 科技   | tech   | 互联网+ | internetplus  |
| 11408005 | 科技_互联网+_在线教育     | tech_internetplus_education          | 科技   | tech   | 互联网+ | internetplus  |
| 11409    | 科技_通信            | tech_communication                   | 科技   | tech   | 通信   | communication |
| 11409000 | 科技_通信_其他         | tech_communication_other             | 科技   | tech   | 通信   | communication |
| 11409001 | 科技_通信_运营商        | tech_communication_operator          | 科技   | tech   | 通信   | communication |
| 11409002 | 科技_通信_设备商        | tech_communication_equipmentprovider | 科技   | tech   | 通信   | communication |
| 11409003 | 科技_通信_通信技术       | tech_communication_technology        | 科技   | tech   | 通信   | communication |
| 11410    | 科技_区块链           | tech_blockchain                      | 科技   | tech   | 区块链  | blockchain    |
| 11410000 | 科技_区块链_其他        | tech_blockchain_other                | 科技   | tech   | 区块链  | blockchain    |
| 11410001 | 科技_区块链_数字货币      | tech_blockchain_digitalcurrency      | 科技   | tech   | 区块链  | blockchain    |
| 11410002 | 科技_区块链_区块链应用     | tech_blockchain_application          | 科技   | tech   | 区块链  | blockchain    |
| 11411    | 科技_前沿产业          | tech_frontier                        | 科技   | tech   | 前沿产业 | frontier      |
| 11411000 | 科技_前沿产业_其他       | tech_frontier_other                  | 科技   | tech   | 前沿产业 | frontier      |
| 11411001 | 科技_前沿产业_人工智能     | tech_frontier_artifical              | 科技   | tech   | 前沿产业 | frontier      |
| 11411002 | 科技_前沿产业_虚拟现实（VR） | tech_frontier_vr                     | 科技   | tech   | 前沿产业 | frontier      |
| 11411003 | 科技_前沿产业_增强现实（AR） | tech_frontier_ar                     | 科技   | tech   | 前沿产业 | frontier      |
| 11411004 | 科技_前沿产业_混合现实（MR） | tech_frontier_mr                     | 科技   | tech   | 前沿产业 | frontier      |
| 11411005 | 科技_前沿产业_3D打印     | tech_frontier_3d                     | 科技   | tech   | 前沿产业 | frontier      |
| 11411006 | 科技_前沿产业_云服务      | tech_frontier_yun                    | 科技   | tech   | 前沿产业 | frontier      |
| 11411007 | 科技_前沿产业_大数据      | tech_frontier_megadata               | 科技   | tech   | 前沿产业 | frontier      |

| 新分类ID    | 新分类中文名          | 新分类英文名                      | 一级分类 | 一级分类英文  | 二级分类    | 二级分类英文名       |
|----------|-----------------|-----------------------------|------|---------|---------|---------------|
| 11411008 | 科技_前沿产业_物联网     | tech_frontier_iot           | 科技   | tech    | 前沿产业    | frontier      |
| 11412    | 科技_信息技术         | tech_it                     | 科技   | tech    | 信息技术    | it            |
| 11412000 | 科技_信息技术_其他      | tech_it_other               | 科技   | tech    | 信息技术    | it            |
| 11412001 | 科技_信息技术_软件开发    | tech_it_softwaredevelopment | 科技   | tech    | 信息技术    | it            |
| 11412002 | 科技_信息技术_信息安全    | tech_it_security            | 科技   | tech    | 信息技术    | it            |
| 11412003 | 科技_信息技术_计算机网络   | tech_it_network             | 科技   | tech    | 信息技术    | it            |
| 11412004 | 科技_信息技术_产品设计    | tech_it_product             | 科技   | tech    | 信息技术    | it            |
| 11412005 | 科技_信息技术_IT管理    | tech_it_management          | 科技   | tech    | 信息技术    | it            |
| 11413    | 科技_科技创投         | tech_venture                | 科技   | tech    | 科技创投    | venture       |
| 11413000 | 科技_科技创投_其他      | tech_venture_other          | 科技   | tech    | 科技创投    | venture       |
| 11413001 | 科技_科技创投_创业干货    | tech_venture_experience     | 科技   | tech    | 科技创投    | venture       |
| 11413002 | 科技_科技创投_孵化器     | tech_venture_incubator      | 科技   | tech    | 科技创投    | venture       |
| 11413003 | 科技_科技创投_投融资     | tech_venture_investment     | 科技   | tech    | 科技创投    | venture       |
| 11414    | 科技_科技股公司        | tech_listedcompany          | 科技   | tech    | 科技股公司   | listedcompany |
| 11414000 | 科技_科技股公司_其他     | tech_listedcompany_other    | 科技   | tech    | 科技股公司   | listedcompany |
| 11414001 | 科技_科技股公司_美股公司动态 | tech_listedcompany_usstock  | 科技   | tech    | 科技股公司   | listedcompany |
| 11414002 | 科技_科技股公司_港股公司动态 | tech_listedcompany_hkstock  | 科技   | tech    | 科技股公司   | listedcompany |
| 11414003 | 科技_科技股公司_A股公司动态 | tech_listedcompany_ashare   | 科技   | tech    | 科技股公司   | listedcompany |
| 11415    | 科技_互联网坊间八卦      | tech_gossip                 | 科技   | tech    | 互联网坊间八卦 | gossip        |
| 11416    | 科技_科技奇趣         | tech_interesting            | 科技   | tech    | 科技奇趣    | interesting   |
| 11417    | 科技_航空航天         | tech_aerospace              | 科技   | tech    | 航空航天    | aerospace     |
| 11418    | 科技_机械           | tech_mechanics              | 科技   | tech    | 机械      | mechanics     |
| 115      | 科学              | science                     | 科学   | science |         |               |

| 新分类ID    | 新分类中文名        | 新分类英文名                 | 一级分类 | 一级分类英文  | 二级分类  | 二级分类英文名        |
|----------|---------------|------------------------|------|---------|-------|----------------|
| 11500    | 科学_其他         | science_other          | 科学   | science | 其他    | other          |
| 11501    | 科学_天文宇宙       | science_space          | 科学   | science | 天文宇宙  | space          |
| 11502    | 科学_地理         | science_geography      | 科学   | science | 地理    | geography      |
| 11503    | 科学_动植物世界      | science_nature         | 科学   | science | 动植物世界 | nature         |
| 11504    | 科学_医学         | science_medical        | 科学   | science | 医学    | medical        |
| 11505    | 科学_生物考古       | science_bioarchaeology | 科学   | science | 生物考古  | bioarchaeology |
| 11506    | 科学_科普         | science_popular        | 科学   | science | 科普    | popular        |
| 11507    | 科学_科幻         | science_fiction        | 科学   | science | 科幻    | fiction        |
| 11508    | 科学_科学实验       | science_exp            | 科学   | science | 科学实验  | exp            |
| 116      | 生活            | life                   | 生活   | life    |       |                |
| 11600    | 生活_其他         | life_other             | 生活   | life    | 其他    | other          |
| 11601    | 生活_生活百科       | life_wiki              | 生活   | life    | 生活百科  | wiki           |
| 11601000 | 生活_生活百科_其他    | life_wiki_other        | 生活   | life    | 生活百科  | wiki           |
| 11601001 | 生活_生活百科_生活小窍门 | life_wiki_tips         | 生活   | life    | 生活百科  | wiki           |
| 11601002 | 生活_生活百科_植物盆栽  | life_wiki_plant        | 生活   | life    | 生活百科  | wiki           |
| 11601003 | 生活_生活百科_动物百科  | life_wiki_animal       | 生活   | life    | 生活百科  | wiki           |
| 11601004 | 生活_生活百科_防骗知识  | life_wiki_antifraud    | 生活   | life    | 生活百科  | wiki           |
| 11601005 | 生活_生活百科_辟谣    | life_wiki_rumor        | 生活   | life    | 生活百科  | wiki           |
| 11602    | 生活_生活方式       | life_lifestyle         | 生活   | life    | 生活方式  | lifestyle      |
| 11602000 | 生活_生活方式_其他    | life_lifestyle_other   | 生活   | life    | 生活方式  | lifestyle      |
| 11602001 | 生活_生活方式_减肥    | life_lifestyle_jianfei | 生活   | life    | 生活方式  | lifestyle      |
| 11602002 | 生活_生活方式_广场舞   | life_lifestyle_psdance | 生活   | life    | 生活方式  | lifestyle      |
| 11602003 | 生活_生活方式_瑜伽    | life_lifestyle_yoga    | 生活   | life    | 生活方式  | lifestyle      |
| 11602004 | 生活_生活方式_咖啡    | life_lifestyle_coffee  | 生活   | life    | 生活方式  | lifestyle      |
| 11602005 | 生活_生活方式_茶     | life_lifestyle_tea     | 生活   | life    | 生活方式  | lifestyle      |
| 11602006 | 生活_生活方        | life_lifestyle_liqueur | 生活   | life    | 生活方式  | lifestyle      |

| 新分类ID    | 新分类中文名       | 新分类英文名                           | 一级分类 | 一级分类英文  | 二级分类 | 二级分类英文名     |
|----------|--------------|----------------------------------|------|---------|------|-------------|
|          | 式_酒          |                                  |      |         |      |             |
| 11602007 | 生活_生活方式_手作   | life_lifestyle_handmade          | 生活   | life    | 生活方式 | lifestyle   |
| 11602008 | 生活_生活方式_插花   | life_lifestyle_flowerarrangement | 生活   | life    | 生活方式 | lifestyle   |
| 11602010 | 生活_生活方式_低碳生活 | life_lifestyle_lowcarbon         | 生活   | life    | 生活方式 | lifestyle   |
| 11603    | 生活_日用家居      | life_houseliving                 | 生活   | life    | 日用家居 | houSELIVING |
| 11603000 | 生活_日用家居_其他   | life_houseliving_other           | 生活   | life    | 日用家居 | houSELIVING |
| 11603001 | 生活_日用家居_家居设计 | life_houseliving_homedesign      | 生活   | life    | 日用家居 | houSELIVING |
| 11603002 | 生活_日用家居_家居风水 | life_houseliving_fengshui        | 生活   | life    | 日用家居 | houSELIVING |
| 11603003 | 生活_日用家居_装修攻略 | life_houseliving_decoration      | 生活   | life    | 日用家居 | houSELIVING |
| 11603004 | 生活_日用家居_家用家具 | life_houseliving_gear            | 生活   | life    | 日用家居 | houSELIVING |
| 11603005 | 生活_日用家居_家用电器 | life_houseliving_jiadian         | 生活   | life    | 日用家居 | houSELIVING |
| 11603006 | 生活_日用家居_家居用品 | life_houseliving_goods           | 生活   | life    | 日用家居 | houSELIVING |
| 11604    | 生活_生活记录      | life_diary                       | 生活   | life    | 生活记录 | diary       |
| 11604000 | 生活_生活记录_其他   | life_diary_other                 | 生活   | life    | 生活记录 | diary       |
| 11604001 | 生活_生活记录_生活随拍 | life_diary_suipai                | 生活   | life    | 生活记录 | diary       |
| 11604002 | 生活_生活记录_街访   | life_diary_interview             | 生活   | life    | 生活记录 | diary       |
| 11604003 | 生活_生活记录_VLOG | life_diary_vlog                  | 生活   | life    | 生活记录 | diary       |
| 117      | 时尚           | fashion                          | 时尚   | fashion |      |             |
| 11700    | 时尚_其他        | fashion_other                    | 时尚   | fashion | 其他   | other       |
| 11701    | 时尚_穿搭        | fashion_dressup                  | 时尚   | fashion | 穿搭   | dressup     |
| 11701000 | 时尚_穿搭_其他     | fashion_dressup_other            | 时尚   | fashion | 穿搭   | dressup     |
| 11701001 | 时尚_穿搭_穿搭教程   | fashion_dressup_dpmj             | 时尚   | fashion | 穿搭   | dressup     |
| 11701002 | 时尚_穿搭_明星穿搭   | fashion_dressup_mxcd             | 时尚   | fashion | 穿搭   | dressup     |
| 11702    | 时尚_街拍        | fashion_jiepai                   | 时尚   | fashion | 街拍   | jiepai      |
| 11702000 | 时尚_街拍_其他     | fashion_jiepai_other             | 时尚   | fashion | 街拍   | jiepai      |

| 新分类ID    | 新分类中文名         | 新分类英文名                        | 一级分类 | 一级分类英文  | 二级分类 | 二级分类英文名   |
|----------|----------------|-------------------------------|------|---------|------|-----------|
| 11702001 | 时尚_街拍_时尚街拍     | fashion_jiepai_cljp           | 时尚   | fashion | 街拍   | jiepai    |
| 11703    | 时尚_美妆          | fashion_meizhuang             | 时尚   | fashion | 美妆   | meizhuang |
| 11703000 | 时尚_美妆_其他       | fashion_meizhuang_other       | 时尚   | fashion | 美妆   | meizhuang |
| 11703001 | 时尚_美妆_彩妆教程     | fashion_meizhuang_makeup      | 时尚   | fashion | 美妆   | meizhuang |
| 11703002 | 时尚_美妆_护肤教程     | fashion_meizhuang_skincare    | 时尚   | fashion | 美妆   | meizhuang |
| 11703003 | 时尚_美妆_化妆品评测    | fashion_meizhuang_pingce      | 时尚   | fashion | 美妆   | meizhuang |
| 11703004 | 时尚_美妆_发型       | fashion_meizhuang_hair        | 时尚   | fashion | 美妆   | meizhuang |
| 11703005 | 时尚_美妆_美甲       | fashion_meizhuang_meijia      | 时尚   | fashion | 美妆   | meizhuang |
| 11703006 | 时尚_美妆_妆前妆后     | fashion_meizhuang_beforeafter | 时尚   | fashion | 美妆   | meizhuang |
| 11703007 | 时尚_美妆_仿妆       | fashion_meizhuang_fangzhuang  | 时尚   | fashion | 美妆   | meizhuang |
| 11703008 | 时尚_美妆_按摩教程     | fashion_meizhuang_massage     | 时尚   | fashion | 美妆   | meizhuang |
| 11704    | 时尚_美体整形        | fashion_meiti                 | 时尚   | fashion | 美体整形 | meiti     |
| 11704000 | 时尚_美体整形_其他     | fashion_meiti_other           | 时尚   | fashion | 美体整形 | meiti     |
| 11704001 | 时尚_美体整形_瘦身塑形   | fashion_meiti_sssx            | 时尚   | fashion | 美体整形 | meiti     |
| 11704002 | 时尚_美体整形_面部整形   | fashion_meiti_beauty          | 时尚   | fashion | 美体整形 | meiti     |
| 11704003 | 时尚_美体整形_美容养生   | fashion_meiti_mrys            | 时尚   | fashion | 美体整形 | meiti     |
| 11705    | 时尚_时尚活动        | fashion_events                | 时尚   | fashion | 时尚活动 | events    |
| 11705000 | 时尚_时尚活动_其他     | fashion_events_other          | 时尚   | fashion | 时尚活动 | events    |
| 11705001 | 时尚_时尚活动_T台秀场   | fashion_events_show           | 时尚   | fashion | 时尚活动 | events    |
| 11705002 | 时尚_时尚活动_时尚品牌活动 | fashion_events_brand          | 时尚   | fashion | 时尚活动 | events    |
| 11706    | 时尚_时尚好物        | fashion_goods                 | 时尚   | fashion | 时尚好物 | goods     |
| 11706000 | 时尚_时尚好物_其他     | fashion_goods_other           | 时尚   | fashion | 时尚好物 | goods     |
| 11706001 | 时尚_时尚好物_鞋      | fashion_goods_shoes           | 时尚   | fashion | 时尚好物 | goods     |
| 11706002 | 时尚_时尚好         | fashion_goods_bags            | 时尚   | fashion | 时尚好物 | goods     |

| 新分类ID    | 新分类中文名          | 新分类英文名                   | 一级分类 | 一级分类英文  | 二级分类 | 二级分类英文名     |
|----------|-----------------|--------------------------|------|---------|------|-------------|
|          | 物_箱包            |                          |      |         |      |             |
| 11706003 | 时尚_时尚好物_腕表      | fashion_goods_watch      | 时尚   | fashion | 时尚好物 | goods       |
| 11706004 | 时尚_时尚好物_珠宝首饰    | fashion_goods_accessory  | 时尚   | fashion | 时尚好物 | goods       |
| 11706005 | 时尚_时尚好物_香水      | fashion_goods_perfume    | 时尚   | fashion | 时尚好物 | goods       |
| 11707    | 时尚_男士时尚         | fashion_man              | 时尚   | fashion | 男士时尚 | man         |
| 11708    | 时尚_时尚资讯         | fashion_news             | 时尚   | fashion | 时尚资讯 | news        |
| 11709    | 时尚_时尚大片         | fashion_blockbuster      | 时尚   | fashion | 时尚大片 | blockbuster |
| 118      | 美食              | food                     | 美食   | food    |      |             |
| 11800    | 美食_其他           | food_other               | 美食   | food    | 其他   | other       |
| 11801    | 美食_菜谱           | food_caipu               | 美食   | food    | 菜谱   | caipu       |
| 11801000 | 美食_菜谱_其他        | food_caipu_other         | 美食   | food    | 菜谱   | caipu       |
| 11801001 | 美食_菜谱_家常菜       | food_caipu_homely        | 美食   | food    | 菜谱   | caipu       |
| 11801002 | 美食_菜谱_快手菜       | food_caipu_kuaishou      | 美食   | food    | 菜谱   | caipu       |
| 11801003 | 美食_菜谱_减肥食谱      | food_caipu_jianfei       | 美食   | food    | 菜谱   | caipu       |
| 11801004 | 美食_菜谱_面食        | food_caipu_pasta         | 美食   | food    | 菜谱   | caipu       |
| 11801005 | 美食_菜谱_早餐        | food_caipu_breakfast     | 美食   | food    | 菜谱   | caipu       |
| 11801006 | 美食_菜谱_烘焙        | food_caipu_baking        | 美食   | food    | 菜谱   | caipu       |
| 11801007 | 美食_菜谱_西餐        | food_caipu_foreign       | 美食   | food    | 菜谱   | caipu       |
| 11801008 | 美食_菜谱_饮品        | food_caipu_drink         | 美食   | food    | 菜谱   | caipu       |
| 11801009 | 美食_菜谱_食材        | food_caipu_ingredients   | 美食   | food    | 菜谱   | caipu       |
| 11801010 | 美食_菜谱_烹饪技巧      | food_caipu_pengrenjiqiao | 美食   | food    | 菜谱   | caipu       |
| 11802    | 美食_美食文化         | food_culture             | 美食   | food    | 美食文化 | culture     |
| 11802000 | 美食_美食文化_其他      | food_culture_other       | 美食   | food    | 美食文化 | culture     |
| 11802001 | 美食_美食文化_美食人文/故事 | food_culture_story       | 美食   | food    | 美食文化 | culture     |
| 11802002 | 美食_美食文化_地方菜系    | food_culture_regional    | 美食   | food    | 美食文化 | culture     |

| 新分类ID    | 新分类中文名          | 新分类英文名                    | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名   |
|----------|-----------------|---------------------------|------|--------|------|-----------|
| 11802003 | 美食_美食文化_火锅      | food_culture_hotpot       | 美食   | food   | 美食文化 | culture   |
| 11802004 | 美食_美食文化_烧烤      | food_culture_barbecue     | 美食   | food   | 美食文化 | culture   |
| 11802005 | 美食_美食文化_水果      | food_culture_fruit        | 美食   | food   | 美食文化 | culture   |
| 11802006 | 美食_美食文化_外国美食    | food_culture_foreign      | 美食   | food   | 美食文化 | culture   |
| 11802007 | 美食_美食文化_乡村美食    | food_culture_rural        | 美食   | food   | 美食文化 | culture   |
| 11802008 | 美食_美食文化_养生美食    | food_culture_yangsheng    | 美食   | food   | 美食文化 | culture   |
| 11802009 | 美食_美食文化_小吃零食    | food_culture_xiaochi      | 美食   | food   | 美食文化 | culture   |
| 11802010 | 美食_美食文化_餐饮礼仪及知识 | food_culture_knowledge    | 美食   | food   | 美食文化 | culture   |
| 11803    | 美食_美食猎奇         | food_weird                | 美食   | food   | 美食猎奇 | weird     |
| 11804    | 美食_吃播           | food_broadcast            | 美食   | food   | 吃播   | broadcast |
| 11804000 | 美食_吃播_其他        | food_broadcast_other      | 美食   | food   | 吃播   | broadcast |
| 11804001 | 美食_吃播_大胃王       | food_broadcast_bigstomach | 美食   | food   | 吃播   | broadcast |
| 11804002 | 美食_吃播_美女吃播      | food_broadcast_beauty     | 美食   | food   | 吃播   | broadcast |
| 11805    | 美食_探店攻略         | food_strategy             | 美食   | food   | 探店攻略 | strategy  |
| 11805000 | 美食_探店攻略_其他      | food_strategy_other       | 美食   | food   | 探店攻略 | strategy  |
| 11805001 | 美食_探店攻略_小吃店     | food_strategy_xiaochi     | 美食   | food   | 探店攻略 | strategy  |
| 11805002 | 美食_探店攻略_网红美食    | food_strategy_wanghong    | 美食   | food   | 探店攻略 | strategy  |
| 11805003 | 美食_探店攻略_城市必吃    | food_strategy_city        | 美食   | food   | 探店攻略 | strategy  |
| 11805004 | 美食_探店攻略_特色西餐/料理 | food_strategy_western     | 美食   | food   | 探店攻略 | strategy  |
| 11805005 | 美食_探店攻略_美食试吃    | food_strategy_taste       | 美食   | food   | 探店攻略 | strategy  |
| 11805006 | 美食_探店攻略_吃喝攻略    | food_strategy_guide       | 美食   | food   | 探店攻略 | strategy  |
| 11806    | 美食_美食随拍         | food_suipai               | 美食   | food   | 美食随拍 | suipai    |
| 119      | 育儿              | baby                      | 育儿   | baby   |      |           |
| 11900    | 育儿_其他           | baby_other                | 育儿   | baby   | 其他   | other     |

| 新分类ID    | 新分类中文名       | 新分类英文名                            | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名         |
|----------|--------------|-----------------------------------|------|--------|------|-----------------|
| 11901    | 育儿_备孕        | baby_prepare                      | 育儿   | baby   | 备孕   | prepare         |
| 11902    | 育儿_孕产        | baby_yunchan                      | 育儿   | baby   | 孕产   | yunchan         |
| 11903    | 育儿_产后        | baby_postnatal                    | 育儿   | baby   | 产后   | postnatal       |
| 11904    | 育儿_饮食        | baby_feeding                      | 育儿   | baby   | 饮食   | feeding         |
| 11904000 | 育儿_饮食_其他     | baby_feeding_other                | 育儿   | baby   | 饮食   | feeding         |
| 11904001 | 育儿_饮食_哺乳     | baby_feeding_breastfeeding        | 育儿   | baby   | 饮食   | feeding         |
| 11904002 | 育儿_饮食_营养辅食   | baby_feeding_supplement           | 育儿   | baby   | 饮食   | feeding         |
| 11905    | 育儿_疾病防治      | baby_health                       | 育儿   | baby   | 疾病防治 | health          |
| 11906    | 育儿_护理发育      | baby_nursing                      | 育儿   | baby   | 护理发育 | nursing         |
| 11907    | 育儿_亲子成长      | baby_qinzichengchang              | 育儿   | baby   | 亲子成长 | qinzichengchang |
| 11907000 | 育儿_亲子成长_其他   | baby_qinzichengchang_other        | 育儿   | baby   | 亲子成长 | qinzichengchang |
| 11907001 | 育儿_亲子成长_幼教早教 | baby_qinzichengchang_preschool    | 育儿   | baby   | 亲子成长 | qinzichengchang |
| 11907002 | 育儿_亲子成长_少儿益智 | baby_qinzichengchang_puzzle       | 育儿   | baby   | 亲子成长 | qinzichengchang |
| 11907003 | 育儿_亲子成长_亲子关系 | baby_qinzichengchang_relationship | 育儿   | baby   | 亲子成长 | qinzichengchang |
| 11908    | 育儿_母婴用品      | baby_goods                        | 育儿   | baby   | 母婴用品 | goods           |
| 11909    | 育儿_萌宝晒娃      | baby_cultbaby                     | 育儿   | baby   | 萌宝晒娃 | cultbaby        |
| 120      | 教育           | edu                               | 教育   | edu    |      |                 |
| 12000    | 教育_其他        | edu_other                         | 教育   | edu    | 其他   | other           |
| 12001    | 教育_学前教育      | edu_preschool                     | 教育   | edu    | 学前教育 | preschool       |
| 12002    | 教育_小学教育      | edu_primary                       | 教育   | edu    | 小学教育 | primary         |
| 12003    | 教育_初中教育      | edu_junior                        | 教育   | edu    | 初中教育 | junior          |
| 12004    | 教育_高中教育      | edu_senior                        | 教育   | edu    | 高中教育 | senior          |
| 12004000 | 教育_高中教育_其他   | edu_senior_other                  | 教育   | edu    | 高中教育 | senior          |
| 12004001 | 教育_高中教育_高考   | edu_senior_gaokao                 | 教育   | edu    | 高中教育 | senior          |
| 12005    | 教育_高等教育      | edu_college                       | 教育   | edu    | 高等教育 | college         |
| 12005000 | 教育_高等教       | edu_college_other                 | 教育   | edu    | 高等教育 | college         |



| 新分类ID    | 新分类中文名         | 新分类英文名                      | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名      |
|----------|----------------|-----------------------------|------|--------|------|--------------|
|          | 育_其他           |                             |      |        |      |              |
| 12005001 | 教育_高等教育_本科     | edu_college_bachelor        | 教育   | edu    | 高等教育 | college      |
| 12005002 | 教育_高等教育_考研     | edu_college_kaoyan          | 教育   | edu    | 高等教育 | college      |
| 12005003 | 教育_高等教育_硕士     | edu_college_master          | 教育   | edu    | 高等教育 | college      |
| 12005004 | 教育_高等教育_博士     | edu_college_doctor          | 教育   | edu    | 高等教育 | college      |
| 12005005 | 教育_高等教育_MBA    | edu_college_mba             | 教育   | edu    | 高等教育 | college      |
| 12005006 | 教育_高等教育_商学院    | edu_college_business        | 教育   | edu    | 高等教育 | college      |
| 12006    | 教育_在线教育        | edu_online                  | 教育   | edu    | 在线教育 | online       |
| 12007    | 教育_语言学习        | edu_language                | 教育   | edu    | 语言学习 | language     |
| 12007000 | 教育_语言学习_其他     | edu_language_other          | 教育   | edu    | 语言学习 | language     |
| 12007001 | 教育_语言学习_英语     | edu_language_english        | 教育   | edu    | 语言学习 | language     |
| 12008    | 教育_留学相关        | edu_abroad                  | 教育   | edu    | 留学相关 | abroad       |
| 12008000 | 教育_留学相关_其他     | edu_abroad_other            | 教育   | edu    | 留学相关 | abroad       |
| 12008001 | 教育_留学相关_GMAT   | edu_abroad_gmat             | 教育   | edu    | 留学相关 | abroad       |
| 12008002 | 教育_留学相关_雅思     | edu_abroad_ielts            | 教育   | edu    | 留学相关 | abroad       |
| 12008003 | 教育_留学相关_SAT    | edu_abroad_sat              | 教育   | edu    | 留学相关 | abroad       |
| 12008004 | 教育_留学相关_GRE    | edu_abroad_gre              | 教育   | edu    | 留学相关 | abroad       |
| 12008005 | 教育_留学相关_TOFEL  | edu_abroad_tofel            | 教育   | edu    | 留学相关 | abroad       |
| 12008006 | 教育_留学相关_海外院校   | edu_abroad_oversea          | 教育   | edu    | 留学相关 | abroad       |
| 12009    | 教育_职业考试        | edu_professional            | 教育   | edu    | 职业考试 | professional |
| 12009000 | 教育_职业考试_其他     | edu_professional_other      | 教育   | edu    | 职业考试 | professional |
| 12009001 | 教育_职业考试_公务员    | edu_professional_gongwuyuan | 教育   | edu    | 职业考试 | professional |
| 12009002 | 教育_职业考试_教师资格考试 | edu_professional_teacher    | 教育   | edu    | 职业考试 | professional |
| 12009003 | 教育_职业考试_司法考试   | edu_professional_lawexam    | 教育   | edu    | 职业考试 | professional |

| 新分类ID    | 新分类中文名         | 新分类英文名                        | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名      |
|----------|----------------|-------------------------------|------|--------|------|--------------|
| 12009004 | 教育_职业考试_会计考试   | edu_professional_accountexam  | 教育   | edu    | 职业考试 | professional |
| 12010    | 教育_技能教育        | edu_skills                    | 教育   | edu    | 技能教育 | skills       |
| 12010000 | 教育_技能教育_其他     | edu_skills_other              | 教育   | edu    | 技能教育 | skills       |
| 12010001 | 教育_技能教育_软件使用   | edu_skills_software           | 教育   | edu    | 技能教育 | skills       |
| 12010002 | 教育_技能教育_编程语言   | edu_skills_code               | 教育   | edu    | 技能教育 | skills       |
| 12010003 | 教育_技能教育_产品设计   | edu_skills_design             | 教育   | edu    | 技能教育 | skills       |
| 12010004 | 教育_技能教育_数据     | edu_skills_data               | 教育   | edu    | 技能教育 | skills       |
| 12010005 | 教育_技能教育_系统硬件   | edu_skills_hardware           | 教育   | edu    | 技能教育 | skills       |
| 12011    | 教育_演讲          | edu_speech                    | 教育   | edu    | 演讲   | speech       |
| 12011000 | 教育_演讲_其他       | edu_speech_other              | 教育   | edu    | 演讲   | speech       |
| 12011001 | 教育_演讲_历史演讲     | edu_speech_historical         | 教育   | edu    | 演讲   | speech       |
| 12011002 | 教育_演讲_当代国内名人演讲 | edu_speech_domestic           | 教育   | edu    | 演讲   | speech       |
| 12011003 | 教育_演讲_当代国外名人演讲 | edu_speech_foreign            | 教育   | edu    | 演讲   | speech       |
| 121      | 健康             | health                        | 健康   | health |      |              |
| 12100    | 健康_其他          | health_other                  | 健康   | health | 其他   | other        |
| 12101    | 健康_医疗          | health_medical                | 健康   | health | 医疗   | medical      |
| 12101000 | 健康_医疗_其他       | health_medical_other          | 健康   | health | 医疗   | medical      |
| 12101001 | 健康_医疗_医疗活动     | health_medical_activity       | 健康   | health | 医疗   | medical      |
| 12101002 | 健康_医疗_医疗资讯     | health_medical_news           | 健康   | health | 医疗   | medical      |
| 12102    | 健康_药品          | health_drugs                  | 健康   | health | 药品   | drugs        |
| 12102000 | 健康_药品_其他       | health_drugs_other            | 健康   | health | 药品   | drugs        |
| 12102001 | 健康_药品_新药       | health_drugs_newdrug          | 健康   | health | 药品   | drugs        |
| 12102002 | 健康_药品_药品质量     | health_drugs_quality          | 健康   | health | 药品   | drugs        |
| 12102003 | 健康_药品_用药禁忌     | health_drugs_contraindication | 健康   | health | 药品   | drugs        |
| 12103    | 健康_疾病防         | health_control                | 健康   | health | 疾病防治 | control      |

| 新分类ID    | 新分类中文名       | 新分类英文名                          | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名 |
|----------|--------------|---------------------------------|------|--------|------|---------|
|          | 治            |                                 |      |        |      |         |
| 12103000 | 健康_疾病防治_其他   | health_control_other            | 健康   | health | 疾病防治 | control |
| 12103001 | 健康_疾病防治_内科   | health_control_internal         | 健康   | health | 疾病防治 | control |
| 12103002 | 健康_疾病防治_外科   | health_control_surgical         | 健康   | health | 疾病防治 | control |
| 12103003 | 健康_疾病防治_肝脏   | health_control_liver            | 健康   | health | 疾病防治 | control |
| 12103004 | 健康_疾病防治_肛肠   | health_control_anorectal        | 健康   | health | 疾病防治 | control |
| 12103005 | 健康_疾病防治_血液血管 | health_control_blood            | 健康   | health | 疾病防治 | control |
| 12103006 | 健康_疾病防治_内分泌  | health_control_endocrine        | 健康   | health | 疾病防治 | control |
| 12103007 | 健康_疾病防治_眼科   | health_control_eyes             | 健康   | health | 疾病防治 | control |
| 12103008 | 健康_疾病防治_耳鼻喉科 | health_control_ent              | 健康   | health | 疾病防治 | control |
| 12103009 | 健康_疾病防治_泌尿科  | health_control_urology          | 健康   | health | 疾病防治 | control |
| 12103010 | 健康_疾病防治_呼吸道  | health_control_respiratory      | 健康   | health | 疾病防治 | control |
| 12103011 | 健康_疾病防治_消化科  | health_control_gastroenterology | 健康   | health | 疾病防治 | control |
| 12103012 | 健康_疾病防治_神经科  | health_control_neurology        | 健康   | health | 疾病防治 | control |
| 12103013 | 健康_疾病防治_传染病  | health_control_chuanranbing     | 健康   | health | 疾病防治 | control |
| 12103014 | 健康_疾病防治_骨科   | health_control_bones            | 健康   | health | 疾病防治 | control |
| 12103015 | 健康_疾病防治_三高   | health_control_sangao           | 健康   | health | 疾病防治 | control |
| 12103016 | 健康_疾病防治_皮肤科  | health_control_pifubing         | 健康   | health | 疾病防治 | control |
| 12103017 | 健康_疾病防治_肾脏   | health_control_kidney           | 健康   | health | 疾病防治 | control |
| 12103018 | 健康_疾病防治_免疫   | health_control_immunology       | 健康   | health | 疾病防治 | control |
| 12104    | 健康_癌症        | health_cancer                   | 健康   | health | 癌症   | cancer  |
| 12104000 | 健康_癌症_其他     | health_cancer_other             | 健康   | health | 癌症   | cancer  |
| 12104001 | 健康_癌症_癌症疾病   | health_cancer_disease           | 健康   | health | 癌症   | cancer  |
| 12104002 | 健康_癌症_癌症治疗   | health_cancer_treatment         | 健康   | health | 癌症   | cancer  |

| 新分类ID    | 新分类中文名       | 新分类英文名                           | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名    |
|----------|--------------|----------------------------------|------|--------|------|------------|
| 12105    | 健康_口腔健康      | health_oral                      | 健康   | health | 口腔健康 | oral       |
| 12105000 | 健康_口腔健康_其他   | health_oral_other                | 健康   | health | 口腔健康 | oral       |
| 12105001 | 健康_口腔健康_口腔疾病 | health_oral_disease              | 健康   | health | 口腔健康 | oral       |
| 12105002 | 健康_口腔健康_口腔整形 | health_oral_plasticsurgery       | 健康   | health | 口腔健康 | oral       |
| 12106    | 健康_心理健康      | health_psychology                | 健康   | health | 心理健康 | psychology |
| 12106000 | 健康_心理健康_其他   | health_psychology_other          | 健康   | health | 心理健康 | psychology |
| 12106001 | 健康_心理健康_心理疾病 | health_psychology_disease        | 健康   | health | 心理健康 | psychology |
| 12106002 | 健康_心理健康_心理教育 | health_psychology_edu            | 健康   | health | 心理健康 | psychology |
| 12107    | 健康_饮食健康      | health_food                      | 健康   | health | 饮食健康 | food       |
| 12107000 | 健康_饮食健康_其他   | health_food_other                | 健康   | health | 饮食健康 | food       |
| 12107001 | 健康_饮食健康_食品安全 | health_food_shippinganquan       | 健康   | health | 饮食健康 | food       |
| 12107002 | 健康_饮食健康_营养   | health_food_nutrition            | 健康   | health | 饮食健康 | food       |
| 12108    | 健康_两性健康      | health_sex                       | 健康   | health | 两性健康 | sex        |
| 12108000 | 健康_两性健康_其他   | health_sex_other                 | 健康   | health | 两性健康 | sex        |
| 12108001 | 健康_两性健康_性病   | health_sex_std                   | 健康   | health | 两性健康 | sex        |
| 12108002 | 健康_两性健康_生殖   | health_sex_reproduction          | 健康   | health | 两性健康 | sex        |
| 12109    | 健康_男性健康      | health_man                       | 健康   | health | 男性健康 | man        |
| 12109000 | 健康_男性健康_其他   | health_man_other                 | 健康   | health | 男性健康 | man        |
| 12109001 | 健康_男性健康_男性疾病 | health_man_disease               | 健康   | health | 男性健康 | man        |
| 12110    | 健康_女性健康      | health_woman                     | 健康   | health | 女性健康 | woman      |
| 12110000 | 健康_女性健康_其他   | health_woman_other               | 健康   | health | 女性健康 | woman      |
| 12110001 | 健康_女性健康_妇科疾病 | health_woman_fuke                | 健康   | health | 女性健康 | woman      |
| 12110002 | 健康_女性健康_生理期  | health_woman_physiologicalperiod | 健康   | health | 女性健康 | woman      |
| 12110003 | 健康_女性健康      | health_woman_breast              | 健康   | health | 女性健康 | woman      |

| 新分类ID    | 新分类中文名       | 新分类英文名                           | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名         |
|----------|--------------|----------------------------------|------|--------|------|-----------------|
|          | 康_乳房健康       |                                  |      |        |      |                 |
| 12111    | 健康_老人健康      | health_old                       | 健康   | health | 老人健康 | old             |
| 12111000 | 健康_老人健康_其他   | health_old_other                 | 健康   | health | 老人健康 | old             |
| 12111001 | 健康_老人健康_老年疾病 | health_old_disease               | 健康   | health | 老人健康 | old             |
| 12112    | 健康_养生保健      | health_yangsheng                 | 健康   | health | 养生保健 | yangsheng       |
| 12112000 | 健康_养生保健_其他   | health_yangsheng_other           | 健康   | health | 养生保健 | yangsheng       |
| 12112001 | 健康_养生保健_养生方法 | health_yangsheng_regimen         | 健康   | health | 养生保健 | yangsheng       |
| 12112002 | 健康_养生保健_保健品  | health_yangsheng_products        | 健康   | health | 养生保健 | yangsheng       |
| 12113    | 健康_中医        | health_chinesemedicine           | 健康   | health | 中医   | chinesemedicine |
| 12113000 | 健康_中医_其他     | health_chinesemedicine_other     | 健康   | health | 中医   | chinesemedicine |
| 12113001 | 健康_中医_中医理论   | health_chinesemedicine_theory    | 健康   | health | 中医   | chinesemedicine |
| 12113002 | 健康_中医_中医治疗   | health_chinesemedicine_treatment | 健康   | health | 中医   | chinesemedicine |
| 122      | 旅游           | travel                           | 旅游   | travel |      |                 |
| 12200    | 旅游_其他        | travel_other                     | 旅游   | travel | 其他   | other           |
| 12201    | 旅游_国内        | travel_demostic                  | 旅游   | travel | 国内   | demostic        |
| 12201000 | 旅游_国内_其他     | travel_demostic_other            | 旅游   | travel | 国内   | demostic        |
| 12201001 | 旅游_国内_北京     | travel_demostic_beijing          | 旅游   | travel | 国内   | demostic        |
| 12201002 | 旅游_国内_上海     | travel_demostic_shanghai         | 旅游   | travel | 国内   | demostic        |
| 12201003 | 旅游_国内_天津     | travel_demostic_tianjin          | 旅游   | travel | 国内   | demostic        |
| 12201004 | 旅游_国内_重庆     | travel_demostic_chongqing        | 旅游   | travel | 国内   | demostic        |
| 12201005 | 旅游_国内_河北     | travel_demostic_hebei            | 旅游   | travel | 国内   | demostic        |
| 12201006 | 旅游_国内_山西     | travel_demostic_jin              | 旅游   | travel | 国内   | demostic        |
| 12201007 | 旅游_国内_内蒙古    | travel_demostic_neimenggu        | 旅游   | travel | 国内   | demostic        |
| 12201008 | 旅游_国内_黑龙江    | travel_demostic_heilongjiang     | 旅游   | travel | 国内   | demostic        |
| 12201009 | 旅游_国内_吉林     | travel_demostic_jilin            | 旅游   | travel | 国内   | demostic        |

| 新分类ID    | 新分类中文名   | 新分类英文名                    | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名  |
|----------|----------|---------------------------|------|--------|------|----------|
| 12201010 | 旅游_国内_辽宁 | travel_demostic_liaojing  | 旅游   | travel | 国内   | demostic |
| 12201011 | 旅游_国内_陕西 | travel_demostic_shanxi    | 旅游   | travel | 国内   | demostic |
| 12201012 | 旅游_国内_甘肃 | travel_demostic_gansu     | 旅游   | travel | 国内   | demostic |
| 12201013 | 旅游_国内_青海 | travel_demostic_qinghai   | 旅游   | travel | 国内   | demostic |
| 12201014 | 旅游_国内_新疆 | travel_demostic_xinjiang  | 旅游   | travel | 国内   | demostic |
| 12201015 | 旅游_国内_宁夏 | travel_demostic_ningxia   | 旅游   | travel | 国内   | demostic |
| 12201016 | 旅游_国内_山东 | travel_demostic_shandong  | 旅游   | travel | 国内   | demostic |
| 12201017 | 旅游_国内_河南 | travel_demostic_henan     | 旅游   | travel | 国内   | demostic |
| 12201018 | 旅游_国内_江苏 | travel_demostic_jiangsu   | 旅游   | travel | 国内   | demostic |
| 12201019 | 旅游_国内_浙江 | travel_demostic_zhejiang  | 旅游   | travel | 国内   | demostic |
| 12201020 | 旅游_国内_安徽 | travel_demostic_anhui     | 旅游   | travel | 国内   | demostic |
| 12201021 | 旅游_国内_江西 | travel_demostic_jiangxi   | 旅游   | travel | 国内   | demostic |
| 12201022 | 旅游_国内_福建 | travel_demostic_fujian    | 旅游   | travel | 国内   | demostic |
| 12201023 | 旅游_国内_湖北 | travel_demostic_hubei     | 旅游   | travel | 国内   | demostic |
| 12201024 | 旅游_国内_湖南 | travel_demostic_hunan     | 旅游   | travel | 国内   | demostic |
| 12201025 | 旅游_国内_广东 | travel_demostic_guangdong | 旅游   | travel | 国内   | demostic |
| 12201026 | 旅游_国内_广西 | travel_demostic_guangxi   | 旅游   | travel | 国内   | demostic |
| 12201027 | 旅游_国内_海南 | travel_demostic_hainan    | 旅游   | travel | 国内   | demostic |
| 12201028 | 旅游_国内_四川 | travel_demostic_sichuan   | 旅游   | travel | 国内   | demostic |
| 12201029 | 旅游_国内_云南 | travel_demostic_yunnan    | 旅游   | travel | 国内   | demostic |
| 12201030 | 旅游_国内_贵州 | travel_demostic_guizhou   | 旅游   | travel | 国内   | demostic |
| 12201031 | 旅游_国内_西藏 | travel_demostic_xizang    | 旅游   | travel | 国内   | demostic |
| 12201032 | 旅游_国内_香港 | travel_demostic_hongkong  | 旅游   | travel | 国内   | demostic |
| 12201033 | 旅游_国内_澳门 | travel_demostic_macao     | 旅游   | travel | 国内   | demostic |

| 新分类ID    | 新分类中文名      | 新分类英文名                     | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名      |
|----------|-------------|----------------------------|------|--------|------|--------------|
|          | 门           |                            |      |        |      |              |
| 12201034 | 旅游_国内_台湾    | travel_demostic_taiwan     | 旅游   | travel | 国内   | demostic     |
| 12201035 | 旅游_国内_国内盘点  | travel_demostic_pandian    | 旅游   | travel | 国内   | demostic     |
| 12202    | 旅游_亚洲       | travel_asia                | 旅游   | travel | 亚洲   | asia         |
| 12202000 | 旅游_亚洲_其他    | travel_asia_other          | 旅游   | travel | 亚洲   | asia         |
| 12202001 | 旅游_亚洲_日本    | travel_asia_japan          | 旅游   | travel | 亚洲   | asia         |
| 12202002 | 旅游_亚洲_韩国    | travel_asia_korea          | 旅游   | travel | 亚洲   | asia         |
| 12202003 | 旅游_亚洲_东亚其他  | travel_asia_east           | 旅游   | travel | 亚洲   | asia         |
| 12202004 | 旅游_亚洲_泰国    | travel_asia_thailand       | 旅游   | travel | 亚洲   | asia         |
| 12202005 | 旅游_亚洲_马来西亚  | travel_asia_malaysia       | 旅游   | travel | 亚洲   | asia         |
| 12202006 | 旅游_亚洲_东南亚其他 | travel_asia_southeast      | 旅游   | travel | 亚洲   | asia         |
| 12202007 | 旅游_亚洲_印度    | travel_asia_india          | 旅游   | travel | 亚洲   | asia         |
| 12202008 | 旅游_亚洲_马尔代夫  | travel_asia_maldives       | 旅游   | travel | 亚洲   | asia         |
| 12202009 | 旅游_亚洲_南亚其他  | travel_asia_south          | 旅游   | travel | 亚洲   | asia         |
| 12202010 | 旅游_亚洲_西亚    | travel_asia_west           | 旅游   | travel | 亚洲   | asia         |
| 12202011 | 旅游_亚洲_中亚    | travel_asia_middle         | 旅游   | travel | 亚洲   | asia         |
| 12202012 | 旅游_亚洲_北亚    | travel_asia_north          | 旅游   | travel | 亚洲   | asia         |
| 12203    | 旅游_欧洲       | travel_europe              | 旅游   | travel | 欧洲   | europe       |
| 12203000 | 旅游_欧洲_其他    | travel_europe_other        | 旅游   | travel | 欧洲   | europe       |
| 12203001 | 旅游_欧洲_英国    | travel_europe_england      | 旅游   | travel | 欧洲   | europe       |
| 12203002 | 旅游_欧洲_法国    | travel_europe_france       | 旅游   | travel | 欧洲   | europe       |
| 12204    | 旅游_北美洲      | travel_northamerica        | 旅游   | travel | 北美洲  | northamerica |
| 12204000 | 旅游_北美洲_其他   | travel_northamerica_other  | 旅游   | travel | 北美洲  | northamerica |
| 12204001 | 旅游_北美洲_美国   | travel_northamerica_usa    | 旅游   | travel | 北美洲  | northamerica |
| 12204002 | 旅游_北美洲_加拿大  | travel_northamerica_canada | 旅游   | travel | 北美洲  | northamerica |

| 新分类ID    | 新分类中文名       | 新分类英文名                     | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名      |
|----------|--------------|----------------------------|------|--------|------|--------------|
| 12205    | 旅游_南美洲       | travel_southamerica        | 旅游   | travel | 南美洲  | southamerica |
| 12205000 | 旅游_南美洲_其他    | travel_southamerica_other  | 旅游   | travel | 南美洲  | southamerica |
| 12205001 | 旅游_南美洲_巴西    | travel_southamerica_brazil | 旅游   | travel | 南美洲  | southamerica |
| 12206    | 旅游_大洋洲       | travel_oceania             | 旅游   | travel | 大洋洲  | oceania      |
| 12206000 | 旅游_大洋洲_其他    | travel_oceania_other       | 旅游   | travel | 大洋洲  | oceania      |
| 12206001 | 旅游_大洋洲_澳大利亚  | travel_oceania_australia   | 旅游   | travel | 大洋洲  | oceania      |
| 12206002 | 旅游_大洋洲_新西兰   | travel_oceania_newzealand  | 旅游   | travel | 大洋洲  | oceania      |
| 12207    | 旅游_非洲        | travel_africa              | 旅游   | travel | 非洲   | africa       |
| 12207000 | 旅游_非洲_其他     | travel_africa_other        | 旅游   | travel | 非洲   | africa       |
| 12207001 | 旅游_非洲_埃及     | travel_africa_egypt        | 旅游   | travel | 非洲   | africa       |
| 12207002 | 旅游_非洲_毛里求斯   | travel_africa_mauritius    | 旅游   | travel | 非洲   | africa       |
| 12208    | 旅游_南极洲       | travel_antarctica          | 旅游   | travel | 南极洲  | antarctica   |
| 12209    | 旅游_旅游盘点      | travel_pandian             | 旅游   | travel | 旅游盘点 | pandian      |
| 12210    | 旅游_旅游产业      | travel_industry            | 旅游   | travel | 旅游产业 | industry     |
| 123      | 汽车           | auto                       | 汽车   | auto   |      |              |
| 12300    | 汽车_其他        | auto_other                 | 汽车   | auto   | 其他   | other        |
| 12301    | 汽车_汽车产业      | auto_chanye                | 汽车   | auto   | 汽车产业 | chanye       |
| 12302    | 汽车_新车资讯      | auto_xinche                | 汽车   | auto   | 新车资讯 | xinche       |
| 12302000 | 汽车_新车资讯_其他   | auto_xinche_other          | 汽车   | auto   | 新车资讯 | xinche       |
| 12302001 | 汽车_新车资讯_微型车  | auto_xinche_mini           | 汽车   | auto   | 新车资讯 | xinche       |
| 12302002 | 汽车_新车资讯_小型车  | auto_xinche_small          | 汽车   | auto   | 新车资讯 | xinche       |
| 12302003 | 汽车_新车资讯_紧凑型车 | auto_xinche_compact        | 汽车   | auto   | 新车资讯 | xinche       |
| 12302004 | 汽车_新车资讯_中型车  | auto_xinche_medium         | 汽车   | auto   | 新车资讯 | xinche       |
| 12302005 | 汽车_新车资讯_豪华车  | auto_xinche_luxury         | 汽车   | auto   | 新车资讯 | xinche       |
| 12302006 | 汽车_新车资讯_SUV  | auto_xinche_suv            | 汽车   | auto   | 新车资讯 | xinche       |
| 12302007 | 汽车_新车资       | auto_xinche_mpv            | 汽车   | auto   | 新车资讯 | xinche       |



| 新分类ID    | 新分类中文名        | 新分类英文名                 | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名 |
|----------|---------------|------------------------|------|--------|------|---------|
|          | 讯_MPV         |                        |      |        |      |         |
| 12302008 | 汽车_新车资讯_超跑    | auto_xinche_super      | 汽车   | auto   | 新车资讯 | xinche  |
| 12302009 | 汽车_新车资讯_皮卡    | auto_xinche_pickup     | 汽车   | auto   | 新车资讯 | xinche  |
| 12302010 | 汽车_新车资讯_摩托车   | auto_xinche_motorcycle | 汽车   | auto   | 新车资讯 | xinche  |
| 12302011 | 汽车_新车资讯_新能源汽车 | auto_xinche_newenergy  | 汽车   | auto   | 新车资讯 | xinche  |
| 12302012 | 汽车_新车资讯_中大型车  | auto_xinche_large      | 汽车   | auto   | 新车资讯 | xinche  |
| 12302013 | 汽车_新车资讯_跑车    | auto_xinche_sports     | 汽车   | auto   | 新车资讯 | xinche  |
| 12303    | 汽车_汽车评测       | auto_ceping            | 汽车   | auto   | 汽车评测 | ceping  |
| 12303000 | 汽车_汽车评测_其他    | auto_ceping_other      | 汽车   | auto   | 汽车评测 | ceping  |
| 12303001 | 汽车_汽车评测_微型车   | auto_ceping_mini       | 汽车   | auto   | 汽车评测 | ceping  |
| 12303002 | 汽车_汽车评测_小型车   | auto_ceping_small      | 汽车   | auto   | 汽车评测 | ceping  |
| 12303003 | 汽车_汽车评测_紧凑型车  | auto_ceping_compact    | 汽车   | auto   | 汽车评测 | ceping  |
| 12303004 | 汽车_汽车评测_中型车   | auto_ceping_medium     | 汽车   | auto   | 汽车评测 | ceping  |
| 12303005 | 汽车_汽车评测_豪华车   | auto_ceping_luxury     | 汽车   | auto   | 汽车评测 | ceping  |
| 12303006 | 汽车_汽车评测_SUV   | auto_ceping_suv        | 汽车   | auto   | 汽车评测 | ceping  |
| 12303007 | 汽车_汽车评测_MPV   | auto_ceping_mpv        | 汽车   | auto   | 汽车评测 | ceping  |
| 12303008 | 汽车_汽车评测_超跑    | auto_ceping_super      | 汽车   | auto   | 汽车评测 | ceping  |
| 12303009 | 汽车_汽车评测_皮卡    | auto_ceping_pickup     | 汽车   | auto   | 汽车评测 | ceping  |
| 12303010 | 汽车_汽车评测_摩托车   | auto_ceping_motorcycle | 汽车   | auto   | 汽车评测 | ceping  |
| 12303011 | 汽车_汽车评测_新能源汽车 | auto_ceping_newenergy  | 汽车   | auto   | 汽车评测 | ceping  |
| 12303012 | 汽车_汽车评测_中大型车  | auto_ceping_large      | 汽车   | auto   | 汽车评测 | ceping  |
| 12303013 | 汽车_汽车评测_跑车    | auto_ceping_sports     | 汽车   | auto   | 汽车评测 | ceping  |
| 12304    | 汽车_汽车导购       | auto_daogou            | 汽车   | auto   | 汽车导购 | daogou  |

| 新分类ID    | 新分类中文名        | 新分类英文名                 | 一级分类 | 一级分类英文  | 二级分类 | 二级分类英文名   |
|----------|---------------|------------------------|------|---------|------|-----------|
| 12304000 | 汽车_汽车导购_其他    | auto_daogou_other      | 汽车   | auto    | 汽车导购 | daogou    |
| 12304001 | 汽车_汽车导购_微型车   | auto_daogou_mini       | 汽车   | auto    | 汽车导购 | daogou    |
| 12304002 | 汽车_汽车导购_小型车   | auto_daogou_small      | 汽车   | auto    | 汽车导购 | daogou    |
| 12304003 | 汽车_汽车导购_紧凑型车  | auto_daogou_compact    | 汽车   | auto    | 汽车导购 | daogou    |
| 12304004 | 汽车_汽车导购_中型车   | auto_daogou_medium     | 汽车   | auto    | 汽车导购 | daogou    |
| 12304005 | 汽车_汽车导购_豪华车   | auto_daogou_luxury     | 汽车   | auto    | 汽车导购 | daogou    |
| 12304006 | 汽车_汽车导购_SUV   | auto_daogou_suv        | 汽车   | auto    | 汽车导购 | daogou    |
| 12304007 | 汽车_汽车导购_MPV   | auto_daogou_mpv        | 汽车   | auto    | 汽车导购 | daogou    |
| 12304008 | 汽车_汽车导购_超跑    | auto_daogou_super      | 汽车   | auto    | 汽车导购 | daogou    |
| 12304009 | 汽车_汽车导购_皮卡    | auto_daogou_pickup     | 汽车   | auto    | 汽车导购 | daogou    |
| 12304010 | 汽车_汽车导购_摩托车   | auto_daogou_motorcycle | 汽车   | auto    | 汽车导购 | daogou    |
| 12304011 | 汽车_汽车导购_新能源汽车 | auto_daogou_newenergy  | 汽车   | auto    | 汽车导购 | daogou    |
| 12304012 | 汽车_汽车导购_中大型车  | auto_daogou_large      | 汽车   | auto    | 汽车导购 | daogou    |
| 12304013 | 汽车_汽车导购_跑车    | auto_daogou_sports     | 汽车   | auto    | 汽车导购 | daogou    |
| 12305    | 汽车_用车保养       | auto_takecare          | 汽车   | auto    | 用车保养 | takecare  |
| 12306    | 汽车_开车技巧       | auto_skills            | 汽车   | auto    | 开车技巧 | skills    |
| 12307    | 汽车_汽车改装       | auto_refit             | 汽车   | auto    | 汽车改装 | refit     |
| 12308    | 汽车_汽车模型       | auto_diymodel          | 汽车   | auto    | 汽车模型 | diymodel  |
| 12309    | 汽车_二手车        | auto_secondcar         | 汽车   | auto    | 二手车  | secondcar |
| 12310    | 汽车_汽车趣闻       | auto_anecdote          | 汽车   | auto    | 汽车趣闻 | anecdote  |
| 12311    | 汽车_汽车广告       | auto_ad                | 汽车   | auto    | 汽车广告 | ad        |
| 12312    | 汽车_车模         | auto_model             | 汽车   | auto    | 车模   | model     |
| 12313    | 汽车_车展资讯       | auto_news              | 汽车   | auto    | 车展资讯 | news      |
| 124      | 财经            | finance                | 财经   | finance |      |           |

| 新分类ID    | 新分类中文名         | 新分类英文名                          | 一级分类 | 一级分类英文  | 二级分类  | 二级分类英文名         |
|----------|----------------|---------------------------------|------|---------|-------|-----------------|
| 12400    | 财经_其他          | finance_other                   | 财经   | finance | 其他    | other           |
| 12401    | 财经_银行          | finance_band                    | 财经   | finance | 银行    | band            |
| 12402    | 财经_基金          | finance_fund                    | 财经   | finance | 基金    | fund            |
| 12402000 | 财经_基金_其他       | finance_fund_other              | 财经   | finance | 基金    | fund            |
| 12402001 | 财经_基金_基金投资技巧   | finance_fund_skills             | 财经   | finance | 基金    | fund            |
| 12403    | 财经_保险          | finance_insurance               | 财经   | finance | 保险    | insurance       |
| 12403000 | 财经_保险_其他       | finance_insurance_other         | 财经   | finance | 保险    | insurance       |
| 12403001 | 财经_保险_投保技巧     | finance_insurance_skills        | 财经   | finance | 保险    | insurance       |
| 12405    | 财经_外汇          | finance_frgexchange             | 财经   | finance | 外汇    | frgexchange     |
| 12405000 | 财经_外汇_其他       | finance_frgexchange_other       | 财经   | finance | 外汇    | frgexchange     |
| 12405001 | 财经_外汇_外汇投资技巧   | finance_frgexchange_skills      | 财经   | finance | 外汇    | frgexchange     |
| 12406    | 财经_债券          | finance_bond                    | 财经   | finance | 债券    | bond            |
| 12406000 | 财经_债券_其他       | finance_bond_other              | 财经   | finance | 债券    | bond            |
| 12406001 | 财经_债券_债券投资技巧   | finance_bond_skills             | 财经   | finance | 债券    | bond            |
| 12407    | 财经_信托          | finance_xintuo                  | 财经   | finance | 信托    | xintuo          |
| 12408    | 财经_贵金属         | finance_metal                   | 财经   | finance | 贵金属   | metal           |
| 12408000 | 财经_贵金属_其他      | finance_metal_other             | 财经   | finance | 贵金属   | metal           |
| 12408001 | 财经_贵金属_贵金属投资技巧 | finance_metal_skills            | 财经   | finance | 贵金属   | metal           |
| 12409    | 财经_能源          | finance_energy                  | 财经   | finance | 能源    | energy          |
| 12410    | 财经_数字货币        | finance_digitalcurrency         | 财经   | finance | 数字货币  | digitalcurrency |
| 12410000 | 财经_数字货币_其他     | finance_digitalcurrency_other   | 财经   | finance | 数字货币  | digitalcurrency |
| 12410001 | 财经_数字货币_比特币    | finance_digitalcurrency_bitcoin | 财经   | finance | 数字货币  | digitalcurrency |
| 12411    | 财经_互联网金融       | finance_interfinance            | 财经   | finance | 互联网金融 | interfinance    |
| 12411000 | 财经_互联网金融_其他    | finance_interfinance_other      | 财经   | finance | 互联网金融 | interfinance    |
| 12411001 | 财经_互联网金融_P2P   | finance_interfinance_p2p        | 财经   | finance | 互联网金融 | interfinance    |
| 12412    | 财经_期货          | finance_futures                 | 财经   | finance | 期货    | futures         |
| 12412000 | 财经_期货_其他       | finance_futures_other           | 财经   | finance | 期货    | futures         |

| 新分类ID    | 新分类中文名       | 新分类英文名                       | 一级分类 | 一级分类英文  | 二级分类 | 二级分类英文名  |
|----------|--------------|------------------------------|------|---------|------|----------|
| 12412001 | 财经_期货_化工品期货  | finance_futures_chemical     | 财经   | finance | 期货   | futures  |
| 12412002 | 财经_期货_能源期货   | finance_futures_energy       | 财经   | finance | 期货   | futures  |
| 12412003 | 财经_期货_农产品期货  | finance_futures_agricultural | 财经   | finance | 期货   | futures  |
| 12412004 | 财经_期货_金属期货   | finance_futures_metal        | 财经   | finance | 期货   | futures  |
| 12412005 | 财经_期货_股指期货   | finance_futures_stock        | 财经   | finance | 期货   | futures  |
| 12412006 | 财经_期货_期货投资技巧 | finance_futures_skills       | 财经   | finance | 期货   | futures  |
| 12413    | 财经_股票        | finance_stock                | 财经   | finance | 股票   | stock    |
| 12413000 | 财经_股票_其他     | finance_stock_other          | 财经   | finance | 股票   | stock    |
| 12413001 | 财经_股票_A股     | finance_stock_ashare         | 财经   | finance | 股票   | stock    |
| 12413002 | 财经_股票_港股     | finance_stock_hk             | 财经   | finance | 股票   | stock    |
| 12413003 | 财经_股票_美股     | finance_stock_us             | 财经   | finance | 股票   | stock    |
| 12413004 | 财经_股票_全球股市   | finance_stock_global         | 财经   | finance | 股票   | stock    |
| 12413005 | 财经_股票_股票投资技巧 | finance_stock_skills         | 财经   | finance | 股票   | stock    |
| 12414    | 财经_宏观经济      | finance_macroeco             | 财经   | finance | 宏观经济 | macroeco |
| 12414000 | 财经_宏观经济_其他   | finance_macroeco_other       | 财经   | finance | 宏观经济 | macroeco |
| 12414001 | 财经_宏观经济_国内经济 | finance_macroeco_domestic    | 财经   | finance | 宏观经济 | macroeco |
| 12414002 | 财经_宏观经济_国外经济 | finance_macroeco_internation | 财经   | finance | 宏观经济 | macroeco |
| 12415    | 财经_产业经济      | finance_industry             | 财经   | finance | 产业经济 | industry |
| 12415000 | 财经_产业经济_其他   | finance_industry_other       | 财经   | finance | 产业经济 | industry |
| 12415001 | 财经_产业经济_房地产业 | finance_industry_house       | 财经   | finance | 产业经济 | industry |
| 12415002 | 财经_产业经济_金融业  | finance_industry_venture     | 财经   | finance | 产业经济 | industry |
| 12415003 | 财经_产业经济_商业消费 | finance_industry_business    | 财经   | finance | 产业经济 | industry |
| 12416    | 财经_公司新闻      | finance_corpnews             | 财经   | finance | 公司新闻 | corpnews |
| 12417    | 财经_财经人物      | finance_famous               | 财经   | finance | 财经人物 | famous   |

| 新分类ID    | 新分类中文名    | 新分类英文名               | 一级分类 | 一级分类英文  | 二级分类   | 二级分类英文名    |
|----------|-----------|----------------------|------|---------|--------|------------|
| 12418    | 财经_投资     | finance_inverstment  | 财经   | finance | 投资     | investment |
| 12419    | 财经_创业     | finance_venture      | 财经   | finance | 创业     | venture    |
| 12420    | 财经_收藏     | finance_collect      | 财经   | finance | 收藏     | collect    |
| 12421    | 财经_理财投资技巧 | finance_mngmoney     | 财经   | finance | 理财投资技巧 | mngmoney   |
| 125      | 房产        | house                | 房产   | house   |        |            |
| 12500    | 房产_其他     | house_other          | 房产   | house   | 其他     | other      |
| 12501    | 房产_房产政策   | house_policy         | 房产   | house   | 房产政策   | policy     |
| 12502    | 房产_房价走势   | house_pricetrend     | 房产   | house   | 房价走势   | pricetrend |
| 12503    | 房产_买房相关   | house_buying         | 房产   | house   | 买房相关   | buying     |
| 12504    | 房产_卖房相关   | house_selling        | 房产   | house   | 卖房相关   | selling    |
| 12505    | 房产_租房相关   | house_renting        | 房产   | house   | 租房相关   | renting    |
| 12506    | 房产_海外房产   | house_oversea        | 房产   | house   | 海外房产   | oversea    |
| 126      | 文化        | cul                  | 文化   | cul     |        |            |
| 12600    | 文化_其他     | cul_other            | 文化   | cul     | 其他     | other      |
| 12601    | 文化_文学     | cul_wenxue           | 文化   | cul     | 文学     | wenxue     |
| 12601000 | 文化_文学_其他  | cul_wenxue_other     | 文化   | cul     | 文学     | wenxue     |
| 12601001 | 文化_文学_诗歌  | cul_wenxue_shige     | 文化   | cul     | 文学     | wenxue     |
| 12602    | 文化_绘画     | cul_painting         | 文化   | cul     | 绘画     | painting   |
| 12603    | 文化_舞蹈     | cul_dance            | 文化   | cul     | 舞蹈     | dance      |
| 12604    | 文化_音乐     | cul_music            | 文化   | cul     | 音乐     | music      |
| 12604000 | 文化_音乐_其他  | cul_music_other      | 文化   | cul     | 音乐     | music      |
| 12604001 | 文化_音乐_乐器  | cul_music_instrument | 文化   | cul     | 音乐     | music      |
| 12605    | 文化_雕塑     | cul_sculpture        | 文化   | cul     | 雕塑     | sculpture  |
| 12606    | 文化_建筑     | cul_building         | 文化   | cul     | 建筑     | building   |
| 12607    | 文化_戏剧     | cul_drama            | 文化   | cul     | 戏剧     | drama      |
| 12608    | 文化_工艺     | cul_shougongyi       | 文化   | cul     | 工艺     | shougongyi |
| 12609    | 文化_书法     | cul_shufa            | 文化   | cul     | 书法     | shufa      |
| 12610    | 文化_民俗     | cul_minsu            | 文化   | cul     | 民俗     | minsu      |
| 12611    | 文化_哲学     | cul_philosophy       | 文化   | cul     | 哲学     | philosophy |
| 12612    | 文化_国学     | cul_guoxue           | 文化   | cul     | 国学     | guoxue     |

| 新分类ID    | 新分类中文名        | 新分类英文名                     | 一级分类 | 一级分类英文  | 二级分类  | 二级分类英文名       |
|----------|---------------|----------------------------|------|---------|-------|---------------|
| 12612000 | 文化_国学_其他      | cul_guoxue_other           | 文化   | cul     | 国学    | guoxue        |
| 12612001 | 文化_国学_儒学      | cul_guoxue_ru              | 文化   | cul     | 国学    | guoxue        |
| 12613    | 文化_宗教         | cul_religion               | 文化   | cul     | 宗教    | religion      |
| 12613000 | 文化_宗教_其他      | cul_religion_other         | 文化   | cul     | 宗教    | religion      |
| 12613001 | 文化_宗教_佛教      | cul_religion_fo            | 文化   | cul     | 宗教    | religion      |
| 12613002 | 文化_宗教_道教      | cul_religion_dao           | 文化   | cul     | 宗教    | religion      |
| 12613003 | 文化_宗教_基督教     | cul_religion_christ        | 文化   | cul     | 宗教    | religion      |
| 12614    | 文化_收藏         | cul_collect                | 文化   | cul     | 收藏    | collect       |
| 12615    | 文化_文物         | cul_wenwu                  | 文化   | cul     | 文物    | wenwu         |
| 12616    | 文化_读书         | cul_dushu                  | 文化   | cul     | 读书    | dushu         |
| 12616000 | 文化_读书_其他      | cul_dushu_other            | 文化   | cul     | 读书    | dushu         |
| 12616001 | 文化_读书_书文朗读    | cul_dushu_read             | 文化   | cul     | 读书    | dushu         |
| 12616002 | 文化_读书_书单推荐    | cul_dushu_list             | 文化   | cul     | 读书    | dushu         |
| 12616003 | 文化_读书_书评析     | cul_dushu_pingxi           | 文化   | cul     | 读书    | dushu         |
| 12617    | 文化_综合文化       | cul_comprehensive          | 文化   | cul     | 综合文化  | comprehensive |
| 12617000 | 文化_综合文化_其他    | cul_comprehensive_other    | 文化   | cul     | 综合文化  | comprehensive |
| 12617001 | 文化_综合文化_文化沙龙  | cul_comprehensive_salon    | 文化   | cul     | 综合文化  | comprehensive |
| 12617002 | 文化_综合文化_对话访谈  | cul_comprehensive_convers  | 文化   | cul     | 综合文化  | comprehensive |
| 12617003 | 文化_综合文化_文化脱口秀 | cul_comprehensive_talkshow | 文化   | cul     | 综合文化  | comprehensive |
| 12617004 | 文化_综合文化_文化演讲  | cul_comprehensive_speech   | 文化   | cul     | 综合文化  | comprehensive |
| 127      | 历史            | history                    | 历史   | history |       |               |
| 12700    | 历史_其他         | history_other              | 历史   | history | 其他    | other         |
| 12701    | 历史_中国古代史      | history_ancient            | 历史   | history | 中国古代史 | ancient       |
| 12701000 | 历史_中国古代史_其他   | history_ancient_other      | 历史   | history | 中国古代史 | ancient       |
| 12701001 | 历史_中国古代史_明清史  | history_ancient_mingqing   | 历史   | history | 中国古代史 | ancient       |

| 新分类ID    | 新分类中文名         | 新分类英文名                       | 一级分类 | 一级分类英文  | 二级分类  | 二级分类英文名     |
|----------|----------------|------------------------------|------|---------|-------|-------------|
| 12702    | 历史_中国近代史       | history_jindai               | 历史   | history | 中国近代史 | jindai      |
| 12702000 | 历史_中国近代史_其他    | history_jindai_other         | 历史   | history | 中国近代史 | jindai      |
| 12702001 | 历史_中国近代史_民国史   | history_jindai_minguo        | 历史   | history | 中国近代史 | jindai      |
| 12703    | 历史_中国当代史       | history_dangdai              | 历史   | history | 中国当代史 | dangdai     |
| 12703000 | 历史_中国当代史_其他    | history_dangdai_other        | 历史   | history | 中国当代史 | dangdai     |
| 12703001 | 历史_中国当代史_建国初期史 | history_dangdai_jianguochuqi | 历史   | history | 中国当代史 | dangdai     |
| 12703002 | 历史_中国当代史_三反五反  | history_dangdai_sanfanwufan  | 历史   | history | 中国当代史 | dangdai     |
| 12703003 | 历史_中国当代史_文革史   | history_dangdai_wenge        | 历史   | history | 中国当代史 | dangdai     |
| 12703004 | 历史_中国当代史_知青史   | history_dangdai_zhiqing      | 历史   | history | 中国当代史 | dangdai     |
| 12703005 | 历史_中国当代史_改革开放史 | history_dangdai_gaigekaifang | 历史   | history | 中国当代史 | dangdai     |
| 12704    | 历史_党史          | history_party                | 历史   | history | 党史    | party       |
| 12705    | 历史_世界史         | history_world                | 历史   | history | 世界史   | world       |
| 12706    | 历史_野史八卦        | history_unofficial           | 历史   | history | 野史八卦  | unofficial  |
| 12707    | 历史_历史考古        | history_archaeology          | 历史   | history | 历史考古  | archaeology |
| 128      | 职场             | career                       | 职场   | career  |       |             |
| 12800    | 职场_其他          | career_other                 | 职场   | career  | 其他    | other       |
| 12801    | 职场_职场经验        | career_experience            | 职场   | career  | 职场经验  | experience  |
| 12802    | 职场_求职面试        | career_interview             | 职场   | career  | 求职面试  | interview   |
| 12803    | 职场_商业管理        | career_business              | 职场   | career  | 商业管理  | business    |
| 12803000 | 职场_商业管理_其他     | career_business_other        | 职场   | career  | 商业管理  | business    |
| 12803001 | 职场_商业管理_企业管理   | career_business_management   | 职场   | career  | 商业管理  | business    |
| 12803002 | 职场_商业管理_人力资源   | career_business_hr           | 职场   | career  | 商业管理  | business    |
| 12803003 | 职场_商业管理_市场营销   | career_business_market       | 职场   | career  | 商业管理  | business    |
| 12804    | 职场_创业指导        | career_venture               | 职场   | career  | 创业指导  | venture     |

| 新分类ID    | 新分类中文名      | 新分类英文名                              | 一级分类 | 一级分类英文      | 二级分类 | 二级分类英文名     |
|----------|-------------|-------------------------------------|------|-------------|------|-------------|
| 12805    | 职场_网店经营     | career_shop                         | 职场   | career      | 网店经营 | shop        |
| 129      | 摄影          | photograph                          | 摄影   | photograph  |      |             |
| 12900    | 摄影_其他       | photograph_other                    | 摄影   | photograph  | 其他   | other       |
| 12901    | 摄影_摄影作品     | photograph_works                    | 摄影   | photograph  | 摄影作品 | works       |
| 12902    | 摄影_摄影技术     | photograph_skills                   | 摄影   | photograph  | 摄影技术 | skills      |
| 12903    | 摄影_摄影器材     | photograph_tool                     | 摄影   | photograph  | 摄影器材 | tool        |
| 12904    | 摄影_后期处理     | photograph_postprocess              | 摄影   | photograph  | 后期处理 | postprocess |
| 130      | 三农          | agriculture                         | 三农   | agriculture |      |             |
| 13000    | 三农_其他       | agriculture_other                   | 三农   | agriculture | 其他   | other       |
| 13001    | 三农_农村       | agriculture_countryside             | 三农   | agriculture | 农村   | countryside |
| 13001000 | 三农_农村_其他    | agriculture_countryside_other       | 三农   | agriculture | 农村   | countryside |
| 13001001 | 三农_农村_农村政策  | agriculture_countryside_policy      | 三农   | agriculture | 农村   | countryside |
| 13001002 | 三农_农村_农村发展  | agriculture_countryside_development | 三农   | agriculture | 农村   | countryside |
| 13002    | 三农_农业       | agriculture_industry                | 三农   | agriculture | 农业   | industry    |
| 13002000 | 三农_农业_其他    | agriculture_industry_other          | 三农   | agriculture | 农业   | industry    |
| 13002001 | 三农_农业_种养殖经验 | agriculture_industry_zhongyangzhi   | 三农   | agriculture | 农业   | industry    |
| 13002002 | 三农_农业_农业市场  | agriculture_industry_market         | 三农   | agriculture | 农业   | industry    |
| 13002003 | 三农_农业_农产品   | agriculture_industry_products       | 三农   | agriculture | 农业   | industry    |
| 13003    | 三农_农民       | agriculture_farmers                 | 三农   | agriculture | 农民   | farmers     |
| 13003000 | 三农_农民_其他    | agriculture_farmers_other           | 三农   | agriculture | 农民   | farmers     |
| 13003001 | 三农_农民_农民生活  | agriculture_farmers_life            | 三农   | agriculture | 农民   | farmers     |
| 13003002 | 三农_农民_致富故事  | agriculture_farmers_zhifu           | 三农   | agriculture | 农民   | farmers     |
| 131      | 宠物          | pet                                 | 宠物   | pet         |      |             |
| 13100    | 宠物_其他       | pet_other                           | 宠物   | pet         | 其他   | other       |
| 13101    | 宠物_猫        | pet_cat                             | 宠物   | pet         | 猫    | cat         |
| 13102    | 宠物_狗        | pet_dog                             | 宠物   | pet         | 狗    | dog         |
| 13103    | 宠物_鸟        | pet_bird                            | 宠物   | pet         | 鸟    | bird        |
| 13104    | 宠物_龙猫仓鼠     | pet_mouse                           | 宠物   | pet         | 龙猫仓鼠 | mouse       |



| 新分类ID    | 新分类中文名      | 新分类英文名                           | 一级分类 | 一级分类英文      | 二级分类 | 二级分类英文名       |
|----------|-------------|----------------------------------|------|-------------|------|---------------|
| 132      | 情感          | emotion                          | 情感   | emotion     |      |               |
| 13200    | 情感_其他       | emotion_other                    | 情感   | emotion     | 其他   | other         |
| 13201    | 情感_家庭关系     | emotion_family                   | 情感   | emotion     | 家庭关系 | family        |
| 13201000 | 情感_家庭关系_其他  | emotion_family_other             | 情感   | emotion     | 家庭关系 | family        |
| 13201001 | 情感_家庭关系_夫妻  | emotion_family_marriage          | 情感   | emotion     | 家庭关系 | family        |
| 13201002 | 情感_家庭关系_婆媳  | emotion_family_poxi              | 情感   | emotion     | 家庭关系 | family        |
| 13201003 | 情感_家庭关系_亲情  | emotion_family_qinqing           | 情感   | emotion     | 家庭关系 | family        |
| 13201004 | 情感_家庭关系_亲戚  | emotion_family_qinqi             | 情感   | emotion     | 家庭关系 | family        |
| 13201005 | 情感_家庭关系_婚外恋 | emotion_family_affair            | 情感   | emotion     | 家庭关系 | family        |
| 13201006 | 情感_家庭关系_相亲  | emotion_family_xiangqin          | 情感   | emotion     | 家庭关系 | family        |
| 13202    | 情感_恋爱关系     | emotion_love                     | 情感   | emotion     | 恋爱关系 | love          |
| 13202000 | 情感_恋爱关系_其他  | emotion_love_other               | 情感   | emotion     | 恋爱关系 | love          |
| 13202001 | 情感_恋爱关系_暗恋  | emotion_love_anlian              | 情感   | emotion     | 恋爱关系 | love          |
| 13202002 | 情感_恋爱关系_表白  | emotion_love_biaobai             | 情感   | emotion     | 恋爱关系 | love          |
| 13202003 | 情感_恋爱关系_异地恋 | emotion_love_yidi                | 情感   | emotion     | 恋爱关系 | love          |
| 13202004 | 情感_恋爱关系_恋爱中 | emotion_love_lianai              | 情感   | emotion     | 恋爱关系 | love          |
| 13202005 | 情感_恋爱关系_失恋  | emotion_love_shilian             | 情感   | emotion     | 恋爱关系 | love          |
| 13202006 | 情感_恋爱关系_同性恋 | emotion_love_gay                 | 情感   | emotion     | 恋爱关系 | love          |
| 13203    | 情感_单身       | emotion_single                   | 情感   | emotion     | 单身   | single        |
| 13204    | 情感_人际关系     | emotion_interpersonal            | 情感   | emotion     | 人际关系 | interpersonal |
| 13204000 | 情感_人际关系_其他  | emotion_interpersonal_other      | 情感   | emotion     | 人际关系 | interpersonal |
| 13204001 | 情感_人际关系_社交  | emotion_interpersonal_social     | 情感   | emotion     | 人际关系 | interpersonal |
| 13204002 | 情感_人际关系_友情  | emotion_interpersonal_friendship | 情感   | emotion     | 人际关系 | interpersonal |
| 133      | 心灵鸡汤        | inspiration                      | 心灵鸡汤 | inspiration |      |               |
| 13300    | 心灵鸡汤_其他     | inspiration_other                | 心灵鸡汤 | inspiration | 其他   | other         |

| 新分类ID    | 新分类中文名       | 新分类英文名                   | 一级分类 | 一级分类英文      | 二级分类   | 二级分类英文名     |
|----------|--------------|--------------------------|------|-------------|--------|-------------|
| 13301    | 心灵鸡汤_人生感悟    | inspiration_lifelessons  | 心灵鸡汤 | inspiration | 人生感悟   | lifelessons |
| 13302    | 心灵鸡汤_名人名言    | inspiration_aphorisms    | 心灵鸡汤 | inspiration | 名人名言   | aphorisms   |
| 134      | 星座命理         | astro                    | 星座命理 | astro       |        |             |
| 13400    | 星座命理_其他      | astro_other              | 星座命理 | astro       | 其他     | other       |
| 13401    | 星座命理_星座      | astro_xingzuo            | 星座命理 | astro       | 星座     | xingzuo     |
| 13402    | 星座命理_生肖      | astro_shengxiao          | 星座命理 | astro       | 生肖     | shengxiao   |
| 13403    | 星座命理_命理      | astro_mingli             | 星座命理 | astro       | 命理     | mingli      |
| 13404    | 星座命理_风水      | astro_fengshui           | 星座命理 | astro       | 风水     | fengshui    |
| 13405    | 星座命理_相术      | astro_xiangshu           | 星座命理 | astro       | 相术     | xiangshu    |
| 13406    | 星座命理_血型      | astro_blood              | 星座命理 | astro       | 血型     | blood       |
| 13407    | 星座命理_塔罗牌     | astro_tarot              | 星座命理 | astro       | 塔罗牌    | tarot       |
| 13408    | 星座命理_测试      | astro_test               | 星座命理 | astro       | 测试     | test        |
| 135      | 广告创意         | creativity               | 广告创意 | creativity  |        |             |
| 13500    | 广告创意_其他      | creativity_other         | 广告创意 | creativity  | 其他     | other       |
| 13501    | 广告创意_创意广告    | creativity_idea          | 广告创意 | creativity  | 创意广告   | idea        |
| 13502    | 广告创意_公益广告    | creativity_gongyi        | 广告创意 | creativity  | 公益广告   | gongyi      |
| 13503    | 广告创意_品牌广告    | creativity_brand         | 广告创意 | creativity  | 品牌广告   | brand       |
| 13504    | 广告创意_宣传短片    | creativity_xuanchuan     | 广告创意 | creativity  | 宣传短片   | xuanchuan   |
| 136      | 纪录片          | documentary              | 纪录片  | documentary |        |             |
| 13600    | 纪录片_其他       | documentary_other        | 纪录片  | documentary | 其他     | other       |
| 13601    | 纪录片_主旋律纪录片   | documentary_mainmelody   | 纪录片  | documentary | 主旋律纪录片 | mainmelody  |
| 13602    | 纪录片_工业纪录片    | documentary_industry     | 纪录片  | documentary | 工业纪录片  | industry    |
| 13603    | 纪录片_美食纪录片    | documentary_food         | 纪录片  | documentary | 美食纪录片  | food        |
| 13604    | 纪录片_自然纪录片    | documentary_nature       | 纪录片  | documentary | 自然纪录片  | nature      |
| 13604000 | 纪录片_自然纪录片_其他 | documentary_nature_other | 纪录片  | documentary | 自然纪录片  | nature      |

| 新分类ID    | 新分类中文名                  | 新分类英文名                     | 一级分类 | 一级分类英文      | 二级分类          | 二级分类英文名     |
|----------|-------------------------|----------------------------|------|-------------|---------------|-------------|
| 13604001 | 纪录片_自然<br>纪录片_地理<br>纪录片 | documentary_nature_geo     | 纪录片  | documentary | 自然纪录片         | nature      |
| 13604002 | 纪录片_自然<br>纪录片_动物<br>纪录片 | documentary_nature_anim    | 纪录片  | documentary | 自然纪录片         | nature      |
| 13604003 | 纪录片_自然<br>纪录片_风景<br>纪录片 | documentary_nature_scen    | 纪录片  | documentary | 自然纪录片         | nature      |
| 13605    | 纪录片_科学<br>纪录片           | documentary_science        | 纪录片  | documentary | 科学纪录片         | science     |
| 13606    | 纪录片_历史<br>纪录片           | documentary_history        | 纪录片  | documentary | 历史纪录片         | history     |
| 13607    | 纪录片_社会<br>现实题材纪录<br>片   | documentary_social         | 纪录片  | documentary | 社会现实题<br>材纪录片 | social      |
| 13608    | 纪录片_旅游<br>纪录片           | documentary_travel         | 纪录片  | documentary | 旅游纪录片         | travel      |
| 13609    | 纪录片_文化<br>纪录片           | documentary_culture        | 纪录片  | documentary | 文化纪录片         | culture     |
| 13610    | 纪录片_人物<br>纪录片           | documentary_character      | 纪录片  | documentary | 人物纪录片         | character   |
| 13611    | 纪录片_军事<br>纪录片           | documentary_military       | 纪录片  | documentary | 军事纪录片         | military    |
| 137      | 小说                      | novel                      | 小说   | novel       |               |             |
| 13700    | 小说_其他                   | novel_other                | 小说   | novel       | 其他            | other       |
| 13701    | 小说_出版                   | novel_publication          | 小说   | novel       | 出版            | publication |
| 13701001 | 小说_出版_传<br>记            | novel_publication_zhuanji  | 小说   | novel       | 出版            | publication |
| 13701002 | 小说_出版_都<br>市            | novel_publication_dushi    | 小说   | novel       | 出版            | publication |
| 13701003 | 小说_出版_短<br>篇            | novel_publication_duanpian | 小说   | novel       | 出版            | publication |
| 13701004 | 小说_出版_法<br>律            | novel_publication_falv     | 小说   | novel       | 出版            | publication |
| 13701005 | 小说_出版_风<br>水            | novel_publication_fengshui | 小说   | novel       | 出版            | publication |
| 13701006 | 小说_出版_工<br>具            | novel_publication_gongju   | 小说   | novel       | 出版            | publication |
| 13701007 | 小说_出版_工<br>业            | novel_publication_gongye   | 小说   | novel       | 出版            | publication |
| 13701008 | 小说_出版_公<br>版            | novel_publication_gongban  | 小说   | novel       | 出版            | publication |
| 13701009 | 小说_出版_古<br>典            | novel_publication_gudian   | 小说   | novel       | 出版            | publication |
| 13701010 | 小说_出版_古<br>籍            | novel_publication_guji     | 小说   | novel       | 出版            | publication |

| 新分类ID    | 新分类中文名     | 新分类英文名                          | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名     |
|----------|------------|---------------------------------|------|--------|------|-------------|
| 13701011 | 小说_出版_管理   | novel_publication_guanli        | 小说   | novel  | 出版   | publication |
| 13701012 | 小说_出版_计算机  | novel_publication_jisuanji      | 小说   | novel  | 出版   | publication |
| 13701013 | 小说_出版_家居   | novel_publication_jiaju         | 小说   | novel  | 出版   | publication |
| 13701014 | 小说_出版_建筑   | novel_publication_jianzhu       | 小说   | novel  | 出版   | publication |
| 13701015 | 小说_出版_教材教辅 | novel_publication_jiaocaijiaofu | 小说   | novel  | 出版   | publication |
| 13701016 | 小说_出版_教育   | novel_publication_jiaoyu        | 小说   | novel  | 出版   | publication |
| 13701017 | 小说_出版_经管   | novel_publication_jingguan      | 小说   | novel  | 出版   | publication |
| 13701018 | 小说_出版_经济   | novel_publication_jingji        | 小说   | novel  | 出版   | publication |
| 13701019 | 小说_出版_剧本   | novel_publication_juben         | 小说   | novel  | 出版   | publication |
| 13701020 | 小说_出版_军事   | novel_publication_junshi        | 小说   | novel  | 出版   | publication |
| 13701021 | 小说_出版_考试   | novel_publication_kaoshi        | 小说   | novel  | 出版   | publication |
| 13701022 | 小说_出版_科幻   | novel_publication_kehuan        | 小说   | novel  | 出版   | publication |
| 13701023 | 小说_出版_科普   | novel_publication_kepu          | 小说   | novel  | 出版   | publication |
| 13701024 | 小说_出版_历史   | novel_publication_lishi         | 小说   | novel  | 出版   | publication |
| 13701025 | 小说_出版_励志   | novel_publication_lizhi         | 小说   | novel  | 出版   | publication |
| 13701026 | 小说_出版_两性   | novel_publication_liangxing     | 小说   | novel  | 出版   | publication |
| 13701027 | 小说_出版_旅游   | novel_publication_lvyou         | 小说   | novel  | 出版   | publication |
| 13701028 | 小说_出版_漫画   | novel_publication_manhua        | 小说   | novel  | 出版   | publication |
| 13701029 | 小说_出版_美食   | novel_publication_meishi        | 小说   | novel  | 出版   | publication |
| 13701030 | 小说_出版_名著   | novel_publication_mingzhu       | 小说   | novel  | 出版   | publication |
| 13701031 | 小说_出版_农林   | novel_publication_nonglin       | 小说   | novel  | 出版   | publication |
| 13701032 | 小说_出版_其他   | novel_publication_qita          | 小说   | novel  | 出版   | publication |
| 13701033 | 小说_出版_奇侠   | novel_publication_qixia         | 小说   | novel  | 出版   | publication |
| 13701034 | 小说_出版_亲    | novel_publication_qinzi         | 小说   | novel  | 出版   | publication |

| 新分类ID    | 新分类中文名     | 新分类英文名                         | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名     |
|----------|------------|--------------------------------|------|--------|------|-------------|
|          | 子          |                                |      |        |      |             |
| 13701035 | 小说_出版_青春   | novel_publication_qingchun     | 小说   | novel  | 出版   | publication |
| 13701036 | 小说_出版_散文   | novel_publication_sanwen       | 小说   | novel  | 出版   | publication |
| 13701037 | 小说_出版_少儿   | novel_publication_shaoer       | 小说   | novel  | 出版   | publication |
| 13701038 | 小说_出版_社会   | novel_publication_shehui       | 小说   | novel  | 出版   | publication |
| 13701039 | 小说_出版_社科   | novel_publication_sheke        | 小说   | novel  | 出版   | publication |
| 13701040 | 小说_出版_时尚   | novel_publication_shishang     | 小说   | novel  | 出版   | publication |
| 13701041 | 小说_出版_手工   | novel_publication_shougong     | 小说   | novel  | 出版   | publication |
| 13701042 | 小说_出版_体育   | novel_publication_tiyu         | 小说   | novel  | 出版   | publication |
| 13701043 | 小说_出版_童书   | novel_publication_tongshu      | 小说   | novel  | 出版   | publication |
| 13701044 | 小说_出版_投资   | novel_publication_touzi        | 小说   | novel  | 出版   | publication |
| 13701045 | 小说_出版_外语   | novel_publication_waiyu        | 小说   | novel  | 出版   | publication |
| 13701046 | 小说_出版_文化   | novel_publication_wenhua       | 小说   | novel  | 出版   | publication |
| 13701047 | 小说_出版_文学   | novel_publication_wenxue       | 小说   | novel  | 出版   | publication |
| 13701048 | 小说_出版_文艺   | novel_publication_wenyi        | 小说   | novel  | 出版   | publication |
| 13701049 | 小说_出版_武侠   | novel_publication_wuxia        | 小说   | novel  | 出版   | publication |
| 13701050 | 小说_出版_小说   | novel_publication_xiaoshuo     | 小说   | novel  | 出版   | publication |
| 13701051 | 小说_出版_心理   | novel_publication_xinli        | 小说   | novel  | 出版   | publication |
| 13701052 | 小说_出版_休闲   | novel_publication_xiuxian      | 小说   | novel  | 出版   | publication |
| 13701053 | 小说_出版_悬疑   | novel_publication_xuanyi       | 小说   | novel  | 出版   | publication |
| 13701054 | 小说_出版_养生   | novel_publication_yangsheng    | 小说   | novel  | 出版   | publication |
| 13701055 | 小说_出版_医学   | novel_publication_yixue        | 小说   | novel  | 出版   | publication |
| 13701056 | 小说_出版_艺术   | novel_publication_yishu        | 小说   | novel  | 出版   | publication |
| 13701057 | 小说_出版_英文图书 | novel_publication_yingwentushu | 小说   | novel  | 出版   | publication |

| 新分类ID    | 新分类中文名    | 新分类英文名                     | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名     |
|----------|-----------|----------------------------|------|--------|------|-------------|
| 13701058 | 小说_出版_游戏  | novel_publication_youxi    | 小说   | novel  | 出版   | publication |
| 13701059 | 小说_出版_育儿  | novel_publication_yuer     | 小说   | novel  | 出版   | publication |
| 13701060 | 小说_出版_杂志  | novel_publication_zazhi    | 小说   | novel  | 出版   | publication |
| 13701061 | 小说_出版_哲学  | novel_publication_zhexue   | 小说   | novel  | 出版   | publication |
| 13701062 | 小说_出版_政治  | novel_publication_zhengzhi | 小说   | novel  | 出版   | publication |
| 13701063 | 小说_出版_自然  | novel_publication_ziran    | 小说   | novel  | 出版   | publication |
| 13701064 | 小说_出版_宗教  | novel_publication_zongjiao | 小说   | novel  | 出版   | publication |
| 13702    | 小说_男频     | novel_man                  | 小说   | novel  | 男频   | man         |
| 13702000 | 小说_男频_其他  | novel_man_other            | 小说   | novel  | 男频   | man         |
| 13702001 | 小说_男频_都市  | novel_man_dushi            | 小说   | novel  | 男频   | man         |
| 13702002 | 小说_男频_短篇  | novel_man_duanpian         | 小说   | novel  | 男频   | man         |
| 13702003 | 小说_男频_二次元 | novel_man_erciyuan         | 小说   | novel  | 男频   | man         |
| 13702004 | 小说_男频_竞技  | novel_man_jingji           | 小说   | novel  | 男频   | man         |
| 13702005 | 小说_男频_军事  | novel_man_junshi           | 小说   | novel  | 男频   | man         |
| 13702006 | 小说_男频_科幻  | novel_man_kehuan           | 小说   | novel  | 男频   | man         |
| 13702007 | 小说_男频_历史  | novel_man_lishi            | 小说   | novel  | 男频   | man         |
| 13702008 | 小说_男频_灵异  | novel_man_lingyi           | 小说   | novel  | 男频   | man         |
| 13702009 | 小说_男频_奇幻  | novel_man_qihuan           | 小说   | novel  | 男频   | man         |
| 13702010 | 小说_男频_体育  | novel_man_tiyu             | 小说   | novel  | 男频   | man         |
| 13702011 | 小说_男频_武侠  | novel_man_wuxia            | 小说   | novel  | 男频   | man         |
| 13702012 | 小说_男频_仙侠  | novel_man_xianxia          | 小说   | novel  | 男频   | man         |
| 13702013 | 小说_男频_现实  | novel_man_xianshi          | 小说   | novel  | 男频   | man         |
| 13702014 | 小说_男频_玄幻  | novel_man_xuanhuan         | 小说   | novel  | 男频   | man         |
| 13702015 | 小说_男频_游戏  | novel_man_youxi            | 小说   | novel  | 男频   | man         |

| 新分类ID    | 新分类中文名    | 新分类英文名               | 一级分类 | 一级分类英文  | 二级分类 | 二级分类英文名   |
|----------|-----------|----------------------|------|---------|------|-----------|
| 13702016 | 小说_男频_职场  | novel_man_zhichang   | 小说   | novel   | 男频   | man       |
| 13703    | 小说_女频     | novel_woman          | 小说   | novel   | 女频   | woman     |
| 13703000 | 小说_女频_其他  | novel_woman_other    | 小说   | novel   | 女频   | woman     |
| 13703001 | 小说_女频_N次元 | novel_woman_nciyuan  | 小说   | novel   | 女频   | woman     |
| 13703002 | 小说_女频_耽美  | novel_woman_danmei   | 小说   | novel   | 女频   | woman     |
| 13703003 | 小说_女频_短篇  | novel_woman_duanpian | 小说   | novel   | 女频   | woman     |
| 13703004 | 小说_女频_古言  | novel_woman_guyan    | 小说   | novel   | 女频   | woman     |
| 13703005 | 小说_女频_幻情  | novel_woman_huanqing | 小说   | novel   | 女频   | woman     |
| 13703006 | 小说_女频_科幻  | novel_woman_kehuan   | 小说   | novel   | 女频   | woman     |
| 13703007 | 小说_女频_青春  | novel_woman_qingchun | 小说   | novel   | 女频   | woman     |
| 13703008 | 小说_女频_同人  | novel_woman_tongren  | 小说   | novel   | 女频   | woman     |
| 13703009 | 小说_女频_仙侠  | novel_woman_xianxia  | 小说   | novel   | 女频   | woman     |
| 13703010 | 小说_女频_现言  | novel_woman_xianyan  | 小说   | novel   | 女频   | woman     |
| 13703011 | 小说_女频_玄幻  | novel_woman_xuanhuan | 小说   | novel   | 女频   | woman     |
| 13703012 | 小说_女频_悬疑  | novel_woman_xuanyi   | 小说   | novel   | 女频   | woman     |
| 13703013 | 小说_女频_游戏  | novel_woman_youxi    | 小说   | novel   | 女频   | woman     |
| 138      | 天气        | weather              | 天气   | weather |      |           |
| 139      | 其他        | other                | 其他   | other   |      |           |
| 140      | 壁纸头像      | pics                 | 壁纸头像 | pics    |      |           |
| 14000    | 壁纸头像_其他   | pics_other           | 壁纸头像 | pics    | 其他   | other     |
| 14001    | 壁纸头像_壁纸   | pics_wallpaper       | 壁纸头像 | pics    | 壁纸   | wallpaper |
| 14002    | 壁纸头像_头像   | pics_portrait        | 壁纸头像 | pics    | 头像   | portrait  |
| 14003    | 壁纸头像_文字图  | pics_textpic         | 壁纸头像 | pics    | 文字图  | textpic   |
| 14004    | 壁纸头像_表情包  | pics_emoji           | 壁纸头像 | pics    | 表情包  | emoji     |
| 14005    | 壁纸头像_套图   | pics_atlas           | 壁纸头像 | pics    | 套图   | atlas     |

| 新分类ID | 新分类中文名 | 新分类英文名         | 一级分类 | 一级分类英文 | 二级分类 | 二级分类英文名 |
|-------|--------|----------------|------|--------|------|---------|
| 141   | 美女     | beauty         | 美女   | beauty |      |         |
| 14100 | 美女_其他  | beauty_other   | 美女   | beauty | 其他   | other   |
| 14101 | 美女_写真  | beauty_xiezhen | 美女   | beauty | 写真   | xiezhen |
| 14102 | 美女_街拍  | beauty_jiepai  | 美女   | beauty | 街拍   | jiepai  |