Package ‘aws.alexa’

June 25, 2019

Title  Client for the Amazon Alexa Web Information Services API
Version  0.1.7
Description  Use the Amazon Alexa Web Information Services API to find information about domains, including the kind of content that they carry, how popular are they—rank and traffic history, sites linking to them, among other things. See <https://aws.amazon.com/awis/> for more information.

Imports  httr, aws.signature, xml2, dplyr
Suggests  testthat, rmarkdown, knitr (>= 1.11), lintr
VignetteBuilder  knitr
License  MIT + file LICENSE
Encoding  UTF-8
LazyData  true
RoxygenNote  6.1.1
NeedsCompilation  no
Author  Gaurav Sood [aut, cre],
        Thomas Leeper [ctb]
Maintainer  Gaurav Sood <gsood07@gmail.com>
Repository  CRAN
Date/Publication  2019-06-25 05:30:07 UTC

R topics documented:

  aws.alexa-package .................................................. 2
  alexa_check ....................................................... 2
  alexa_GET .......................................................... 3
  alexa_PROCESS .................................................... 3
  browse_categories .................................................. 4
  category_listing ..................................................... 5
  in_links ............................................................. 6
  set_secret_key ....................................................... 6
  traffic_history ..................................................... 7
  url_info ............................................................. 8
aws.alexa-package  aws.alexa: R Client for the Alexa Web Information Services API

Description

Find information about domains, including the kind of content that they carry, how popular are
they, sites linking to them, among other things. The package provides access to the Alexa Web

To learn how to use aws.alexa, see this vignette: https://CRAN.R-project.org/package=aws.
alexa/vignettes/overview.html.

You need to get credentials (Access Key ID and Secret Access Key) to use this application. If you
haven’t already, get these at https://aws.amazon.com/. And set these using set_secret_key

Author(s)

Gaurav Sood

alexa_check  Request Response Verification

Description

Request Response Verification

Usage

alexa_check(req)

Arguments

req request

Value

in case of failure, a message
alexa_GET  Base POST AND GET functions. Not exported.

Description

GET

Usage

alexa_GET(query, key = Sys.getenv("AWS_ACCESS_KEY_ID"),
secret = Sys.getenv("AWS_SECRET_ACCESS_KEY"),
verbose = getOption("verbose", FALSE), session_token = NULL,
region = "us-west-1", headers = list(), ...)

Arguments

query  query list
key    A character string containing an AWS Access Key ID. The default is retrieved
       from Sys.getenv("AWS_ACCESS_KEY_ID").
secret A character string containing an AWS Secret Access Key. The default is re-
       trieved from Sys.getenv("AWS_SECRET_ACCESS_KEY").
verbose A logical indicating whether to be verbose. Default is given by options("verbose").
session_token Optionally, a character string containing an AWS temporary Session Token. If
       missing, defaults to value stored in environment variable AWS_SESSION_TOKEN.
region A character string containing the AWS region. If missing, defaults to “us-west-
       1”.
headers A list of request headers for the REST call.
...    Additional arguments passed to GET.

Value

list

alexa_PROCESS Postprocess the results a bit

Description

Postprocess the results a bit

Usage

alexa_PROCESS(res)
browse_categories

Arguments

res
result

Value
display request ID and Response Status and the first member of the list

---

browse_categories  Browse Categories

Description

Uses data from dmoz.org, which is no longer updated.

Usage

browse_categories(path = NULL, response_group = "Categories",
description = TRUE, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path</td>
<td>String; Required; valid category path</td>
</tr>
<tr>
<td>response_group</td>
<td>String; Required; One of the following: Categories, RelatedCategories, LanguageCategories, LetterBars</td>
</tr>
<tr>
<td>description</td>
<td>Boolean; Optional; Whether or not to return descriptions of categories; Default is TRUE</td>
</tr>
</tbody>
</table>

...  Additional arguments passed to alexa_GET.

Value
data.frame with 5 columns: path, title, sub_category_count, total_listing_count, description

References


Examples

```r
## Not run:
browse_categories(path="Top/Arts")

## End(Not run)
```
category_listing

Description

Uses data from dmoz.org, which is no longer updated. For any given category, it returns a list of site listings contained within that category.

Usage

category_listing(path = NULL, sort_by = "Popularity", recursive = TRUE, start = 0, count = 20, description = TRUE, ...)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>path</td>
<td>String</td>
<td>Required; valid category path</td>
</tr>
<tr>
<td>sort_by</td>
<td>sort results by Popularity, Title, or AverageReview</td>
<td></td>
</tr>
<tr>
<td>recursive</td>
<td>Boolean</td>
<td>Whether to return listings for the current category only, or for the current category plus all subcategories, Default is TRUE</td>
</tr>
<tr>
<td>start</td>
<td>index of result at which to start; default is 0</td>
<td></td>
</tr>
<tr>
<td>count</td>
<td>Number of results to return for this request; Max = 20; Default = 20</td>
<td></td>
</tr>
<tr>
<td>description</td>
<td>Boolean</td>
<td>Optional; Whether or not to return descriptions of categories; Default is TRUE</td>
</tr>
</tbody>
</table>

Value
data.frame

References

http://docs.aws.amazon.com/AlexaWebInfoService/latest/ApiReference_CategoryListingsAction.html

Examples

```r
## Not run:
category_listing(path="Top/Arts")

## End(Not run)
```
in_links  

Sites linking to the site

Description

Sites linking to the site

Usage

in_links(url = NULL, start = 0, count = 20, ...)

Arguments

url  String; Required; valid url
start  index of result at which to start; default = 0
count  Number of results to return for this request; Max = 20; Default = 20
...  Additional arguments passed to alexa_GET.

Value

data.frame with two columns: title (site hostname) and url (specific url)

References

http://docs.aws.amazon.com/AlexaWebInfoService/latest/ApiReference_SitesLinkingInAction.html

Examples

```r
## Not run:
in_links(url = "google.com")
```

set_secret_key  

Set up Key and Secret

Description

Get the Access Key ID and Secret Access Key by logging into https://console.aws.amazon.com/, clicking on the username followed by security credentials. The function sets two environmental variables AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY. These environment variables persist within a R session. The function looks for these variables
**traffic_history**

**Usage**

```
set_secret_key(key = NULL, secret = NULL, force = FALSE)
```

**Arguments**

- **key**: String; Required; Access Key ID
- **secret**: String; Required; Secret Access Key
- **force**: String; Required; Force change the AWS_ACCESS_KEY_ID and AWS_SECRET_ACCESS_KEY stored in the environment

**References**

[https://aws.amazon.com/](https://aws.amazon.com/)

**Examples**

```r
## Not run:
set_secret_key(key = "key", secret = "secret")
```

```r
## End(Not run)
```

---

**traffic_history**

Get Traffic History of a URL

**Description**

Get Traffic History of a URL

**Usage**

```
traffic_history(url = NULL, range = 31, start = NULL, ...)
```

**Arguments**

- **url**: String; Required; valid url
- **range**: Integer; Required; Default is 31, Maximum is 31. Pick an integer between 1 and 31.
- **start**: String; Optional; A date within the last 4 years in format YYYYMMDD.
- **...**: Additional arguments passed to `alexa_GET`.

**Value**

data.frame with the following columns: site, start, range, date, page_views_per_million, page_views_per_user
References
http://docs.aws.amazon.com/AlexaWebInfoService/latest/ApiReference_TrafficHistoryAction.html

Examples

## Not run:
traffic_history(url = "http://www.google.com", start = "20160505")

## End(Not run)

---

url_info

*Get Information about a URL*

**Description**
Get Information about a URL

**Usage**
url_info(url = NULL, response_group = "SiteData", ...)

**Arguments**

- **url**
  String; Required; valid url

- **response_group**
  String; Required; One of the following: RelatedLinks, Categories, Rank, RankBy-Country, UsageStats, AdultContent, Speed, Language, OwnedDomains, LinksIn-Count, SiteData Default is 'SiteData'. Multiple fields can be passed. They must be separated by comma.

- ... Additional arguments passed to alexa_GET.

**Value**
named list

**References**

**Examples**

## Not run:
url_info(url = "http://www.google.com")

## End(Not run)
Index

alexa_check, 2
alexa_GET, 3, 4–8
alexa_PROCESS, 3
aws.alexa (aws.alexa-package), 2
aws.alexa-package, 2

browse_categories, 4

category_listing, 5

GET, 3

in_links, 6

set_secret_key, 2, 6

traffic_history, 7

url_info, 8