

Package ‘chorrds’

January 28, 2019

Title Music Chords Extraction

Type Package

Version 0.1.8

Author Bruna Wundervald

Maintainer Bruna Wundervald <brunadaviesw@gmail.com>

Description Extracts music chords from the 'CifraClub' website <<https://www.cifraclub.com.br/>>. The package also has functions for cleaning the extracted data and feature extraction.

Depends R (>= 3.2.4)

Suggests ggplot2, knitr, network

URL <https://github.com/r-music/chorrds>

BugReports <https://github.com/r-music/chorrds/issues>

License GPL-2

Encoding UTF-8

LazyData true

RoxygenNote 6.1.1

Imports stringr, dplyr, XML, magrittr, purrr

NeedsCompilation no

Repository CRAN

Date/Publication 2019-01-28 10:40:03 UTC

R topics documented:

| | |
|-------------------|---|
| all | 2 |
| clean | 2 |
| deg_maj | 3 |
| deg_min | 3 |
| dist | 4 |
| eqv | 4 |

| | |
|-----------------------|---|
| genre | 5 |
| get_chords | 5 |
| get_songs | 6 |
| search_data | 6 |

| | |
|--------------|----------|
| Index | 8 |
|--------------|----------|

| | |
|-----|------------|
| all | <i>all</i> |
|-----|------------|

Description

All data available.

Usage

all

Format

A data frame with 6 variables:

date integer. The date of the album which contains the music.
 music factor. The name of the music.
 popul integer. The popularity of the music.
 chord factor. The chord names of each music, by order of occurrence in the music.
 key factor. The key for each music.
 artist factor. The name of the artist

| | |
|-------|--------------|
| clean | <i>clean</i> |
|-------|--------------|

Description

Clean data when there is some excessive long text on a column.

Usage

```
clean(da, column = "chord", long = 15, message = TRUE)
```

Arguments

| | |
|---------|---|
| da | a data.frame. |
| column | string. The column by which we want to make the cleaning. |
| long | numeric. The longest string we wish exists on our |
| message | logical. Should the function print how many lines were removed? |

Value

A database, with the text cleaning done.

Examples

```
{
## Not run:
data("caetano")
  clean(da = caetano, column = "chord", long = 15, message = TRUE)

## End(Not run)
}
```

 deg_maj

deg_maj

Description

Accessory data with the chords present in each scale, with its respective degrees, for the minor cases.

Usage

```
deg_maj
```

Format

An object of class `data.frame` with 7 rows and 18 columns.

 deg_min

deg_min

Description

Accessory data with the chords present in each scale, with its respective degrees, for the minor cases.

Usage

```
deg_min
```

Format

An object of class `data.frame` with 7 rows and 16 columns.

 dist
*dist***Description**

A simple measure of the chords distances in the circle of fifths.

Usage

dist

Format

A data frame with 3 variables:

prox factor. The chord.

dist numeric. The distance from C in the circle of fifths.

order integer. The order in the circle of fifths.

 eqv
*eqv***Description**

Accessory data for the recognition of equivalent keys, including major and minor relatives.

Usage

eqv

Format

A data frame with 3 variables:

key factor. Keys ordered by the circle of fifths.

minor.rel factor. Relative minors of the key in the previous column.

rep num. A number indicating if the key scale is equivalent to some other; repeated numbers indicate equivalent keys.

| | |
|-------|--------------|
| genre | <i>genre</i> |
|-------|--------------|

Description

Accessory data with the genre for each artist in the package.

Usage

```
genre
```

Format

An object of class `data.frame` with 106 rows and 2 columns.

| | |
|------------|-------------------|
| get_chords | <i>get_chords</i> |
|------------|-------------------|

Description

Extracts music chords from an artist.

Usage

```
get_chords(songs, nf = FALSE)
```

Arguments

| | |
|-------|--|
| songs | character. The song url. |
| nf | TRUE or FALSE. If the chords of a song are not found, should we return this information on the database? |

Value

A database with the chords of the input songs, if found.

Examples

```
{
  songs <- chorrrds::get_songs("tim-maia")
  get_chords(songs$url[4])
}
```

get_songs *get_songs*

Description

Get songs names from an artist.

Usage

```
get_songs(artist)
```

Arguments

artist character. The artist's name.

Value

If the artist (or band) is found, it is returned a database with the song names and URLs.

Examples

```
{  
  get_songs("jorge")  
  get_songs("los-hermanos")  
}
```

search_data *search_data*

Description

Search artists in the available package database.

Usage

```
search_data(name)
```

Arguments

name character. The searched artist's name.

Value

If a database with the corresponding searched name is found, it's name is returned. If not, nothing is returned.

search_data

7

Examples

```
{  
  search_data("chico")  
}
```

Index

*Topic **datasets**

- all, [2](#)
- deg_maj, [3](#)
- deg_min, [3](#)
- dist, [4](#)
- eqv, [4](#)
- genre, [5](#)

all, [2](#)

clean, [2](#)

deg_maj, [3](#)

deg_min, [3](#)

dist, [4](#)

eqv, [4](#)

genre, [5](#)

get_chords, [5](#)

get_songs, [6](#)

search_data, [6](#)