Package ‘mscstts’

May 10, 2019

Version 0.5.2
Title R Client for the Microsoft Cognitive Services 'Text-to-Speech' REST API
Description R Client for the Microsoft Cognitive Services 'Text-to-Speech' REST API, including voice synthesis. A valid account must be registered at the Microsoft Cognitive Services website <https://www.microsoft.com/cognitive-services/> in order to obtain a (free) API key. Without an API key, this package will not work properly.
License GPL-3
Encoding UTF-8
LazyData true
Type Package
Suggests knitr, rmarkdown, covr, testthat
URL https://github.com/muschellij2/mscstts
BugReports https://github.com/muschellij2/mscstts/issues
RoxygenNote 6.1.1
Imports httr, jsonlite, tuneR
NeedsCompilation no
Author John Muschelli [aut, cre] (<https://orcid.org/0000-0001-6469-1750>)
Maintainer John Muschelli <muschellij2@gmail.com>
Repository CRAN
Date/Publication 2019-05-10 05:00:03 UTC

R topics documented:

ms_create_ssml .......................................................... 2
ms_get_tts_key .......................................................... 3
ms_get_tts_token ......................................................... 4
ms_language_to_ms_name .............................................. 4
ms_create_ssml

Description
Create SSML for Passing to Synthesize

Usage
ms_create_ssml(script, voice = NULL, gender = c("Female", "Male"),
language = "en-US", escape = FALSE)
ms_voice_info(voice)

Arguments
- script: A character vector of lines to be spoken
- voice: full voice name, usually from ms_language_to_ms_name. Will override language and gender.
- gender: Sex of the Speaker
- language: Language to be spoken, must be from ms_language_codes
- escape: Should non-standard characters be substituted? Should not be used if script has SSML tags

Value
A character string of the text and SSML markup

Examples
ms_create_ssml("hey I really like things & dogs", escape = TRUE)
ms_create_ssml("hey I really like things")
ms_create_ssml("hey I <emphasis level="strong">really like</emphasis> things")
ms_create_ssml("hey I <emphasis level="strong">really like</emphasis> things", escape = TRUE)
ms_get_tts_key

Find API Key for Microsoft Text To Speech (TTS) or Cognitive Services

Description

Determines if \texttt{option(ms\_tts\_key)} or \texttt{option(ms\_tts\_key)} is set. If not, it stops and returns an error. If so, returns the value.

Usage

\begin{verbatim}
ms_get_tts_key(api_key = NULL, error = TRUE)
ms_have_tts_key(api_key = NULL)
ms_set_tts_key(api_key)
ms_valid_tts_key(api_key = NULL, region = ms_regions())
\end{verbatim}

Arguments

- \texttt{api\_key}: Microsoft Cognitive Services API key
- \texttt{error}: Should the function error if \texttt{api\_key = NULL}?
- \texttt{region}: Subscription region for your key. See \url{https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/rest-apis#text-to-speech}

Value

API key

Note

You can either set the API key using \texttt{option(ms\_tts\_key)} or have it accessible by \texttt{api\_key = Sys.getenv('MS\_TTS\_API\_KEY')} or \texttt{api\_key = Sys.getenv('MS\_TTS\_API\_KEY1')} or \texttt{api\_key = Sys.getenv('MS\_TTS\_API\_KEY2')}.

Examples

\begin{verbatim}
res = ms_get_tts_key(error = FALSE)
\end{verbatim}
ms_get_tts_token  Get Microsoft Text To Speech (TTS) or Cognitive Services Token from API Key

Description
Get Microsoft Text To Speech (TTS) or Cognitive Services Token from API Key

Usage
ms_get_tts_token(api_key = NULL, region = ms_regions())
ms_auth_url(region = ms_regions())
ms_expired_token(token = NULL)

Arguments
api_key  Microsoft Cognitive Services API key
region  Subscription region for your key. See https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/rest-apis#text-to-speech
token  An authentication of class token, likely from ms_get_tts_token

Value
A list of the request, and token

Examples
if (ms_valid_tts_key()) {
  token = ms_get_tts_token()
}

ms_language_to_ms_name  Language and Gender to Microsoft Voice Name

Description
Language and Gender to Microsoft Voice Name
ms_list_voices

Usage

ms_language_to_ms_name(language = "en-US", gender = c("Female", "Male"))

ms_validate_language_gender(language = "en-US", gender = c("Female", "Male"))

Arguments

language A language code, see ms_language_codes

gender Either Male or Female, not all languages support both genders

Value

A character string of the name of the voice

Examples

ms_language_to_ms_name()
ms_validate_language_gender()

ms_list_voices List Voices

Description

List Voices

Usage

ms_list_voices(token = NULL, api_key = NULL, region = NULL, ...)

Arguments

token An authentication token, base-64 encoded usually from ms_get_tts_token. If not provided, will be created from ms_get_tts_token

api_key Microsoft Cognitive Services API key, if token is not provided.

region Subscription region for your key. See https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/rest-apis#text-to-speech

... Additional arguments to send to GET

Value

A data.frame of the names and their long names.
## ms_locales

### Description
Locales for Microsoft Translate

### Usage
- `ms_locales()`
- `ms_locale_df()`
- `ms_languages()`
- `ms_language_codes()`

### Value
A list of Locales and the Speech Voices

### Examples
```r
ms_locales()
```

## ms_read_synthesis

### Description
Read Synthesized output

### Usage
```r
ms_read_synthesis(output)
```
**Arguments**

output

List from `ms_synthesize` with elements `output_format` and `content`

**Value**

A Wave Object

**Note**

The `tuneR` package cannot read all different types of the output here.

---

**Description**

Get Microsoft Text To Speech (TTS) or Cognitive Services Token from API Key

**Usage**

```r
ms_synthesize(script, token = NULL, api_key = NULL, 
gender = c("Female", "Male"), language = "en-US", voice = NULL, 
output_format = c("raw-16khz-16bit-mono-pcm", 
"raw-8khz-8bit-mono-mulaw", "riff-8khz-8bit-mono-alaw", 
"riff-8khz-8bit-mono-mulaw", "riff-16khz-16bit-mono-pcm", 
"audio-16khz-128kb bitrate-mono-mp3", "audio-16khz-64kb bitrate-mono-mp3", 
"audio-16khz-32kb bitrate-mono-mp3", "raw-24khz-16bit-mono-pcm", 
"riff-24khz-16bit-mono-pcm", "audio-24khz-160kb bitrate-mono-mp3", 
"audio-24khz-96kb bitrate-mono-mp3", "audio-24khz-48kb bitrate-mono-mp3"), 
escape = FALSE, region = NULL, api = c("tts", "bing"), ...)

ms_region(region = ms_regions())

ms_regions()

ms_set_region(region = ms_regions())

ms_synthesize_api_url(api = c("tts", "bing"), region = NULL)
```

**Arguments**

- `script`: A character vector of lines to be spoken
- `token`: An authentication token, base-64 encoded usually from `ms_get_tts_token`. If not provided, will be created from `ms_get_tts_token`
- `api_key`: Microsoft Cognitive Services API key, if token is not provided.
gender  Sex of the Speaker
language Language to be spoken, must be from ms_language_codes
voice full voice name, usually from ms_language_to_ms_name. Will override language and gender.
output_format Format of the output, see https://docs.microsoft.com/en-us/azure/cognitive-services/speech/api-reference-rest/bingvoiceoutput for more information
escape Should non-standard characters be substituted? Should not be used if script has SSML tags. See ms_create_ssml
region Subscription region for your key. See https://docs.microsoft.com/en-us/azure/cognitive-services/speech-service/rest-apis#text-to-speech
api which API to authorize on, either tts for text to speech or bing for Bing text to speech API
... Additional arguments to send to POST

Value
A list of the request, content, token, and ‘SSML’.

Note
The content is likely in a binary format and the output depends on the ‘output_format’ chosen. For example, if the ‘output_format’ is an ‘MP3’, then see below example

Examples

```r
## Not run:
if (ms_have_tts_key()) {
  res = ms_synthesize(
    script = "hey, how are you doing? I'm doing pretty good",
    output_format = "audio-16khz-128kbitrate-mono-mp3")
  tmp <- tempfile(fileext = ".mp3")
  writeBin(res$content, con = tmp)
  mp3 = tuner::readMP3(tmp)
}
```

## End(Not run)
ms_regions()

---

**print.token**

*Print method for token*

**Description**

Print method for token
print.token

Usage

```r
## S3 method for class 'token'
print(x, reveal = FALSE, ...)

reveal(x, ...)
```

Arguments

- `x`:
  - an object used to select a method.
- `reveal`:
  - Should the token be revealed
- `...`:
  - further arguments passed to or from other methods

Examples

```r
x <- BasdfB
class(x) <- BtokenB
print(x)
attr(x, "timestamp") <- Sys.time()
print(x)

print(x, reveal = TRUE)
```
Index

GET, 5
ms_auth_url (ms_get_tts_token), 4
ms_create_ssml, 2, 8
ms_expired_token (ms_get_tts_token), 4
ms_get_tts_key, 3
ms_get_tts_token, 4, 4, 5, 7
ms_have_tts_key (ms_get_tts_key), 3
ms_language_codes, 2, 5, 8
ms_language_codes (ms_locales), 6
ms_language_to_ms_name, 2, 4, 8
ms_languages (ms_locales), 6
ms_list_voices, 5
ms_locale_df (ms_locales), 6
ms_locales, 6
ms_read_synthesis, 6
ms_region (ms_synthesize), 7
ms_regions (ms_synthesize), 7
ms_set_region (ms_synthesize), 7
ms_set_tts_key (ms_get_tts_key), 3
ms_synthesize, 7, 7
ms_synthesize_api_url (ms_synthesize), 7
ms_valid_tts_key (ms_get_tts_key), 3
ms_validate_language_gender (ms_language_to_ms_name), 4
ms_voice_info (ms_create_ssml), 2

POST, 8
print.token, 8
reveal (print.token), 8