

# Package ‘aws.transcribe’

March 11, 2020

**Type** Package

**Title** Client for 'AWS Transcribe'

**Version** 0.1.3

**Date** 2020-03-10

**Description** Client for 'AWS Transcribe' <<https://aws.amazon.com/documentation/transcribe>>, a cloud transcription service that can convert an audio media file in English and other languages into a text transcript.

**License** GPL (>= 2)

**URL** <https://github.com/cloudyr/aws.transcribe>

**BugReports** <https://github.com/cloudyr/aws.transcribe/issues>

**Imports** tools, httr, jsonlite, aws.signature (>= 0.3.4)

**Suggests** testthat

**Encoding** UTF-8

**RoxygenNote** 7.0.2

**NeedsCompilation** no

**Author** Thomas J. Leeper [aut] (<<https://orcid.org/0000-0003-4097-6326>>),  
Antoine Sachet [cre]

**Maintainer** Antoine Sachet <[antoine.sac@gmail.com](mailto:antoine.sac@gmail.com)>

**Repository** CRAN

**Date/Publication** 2020-03-11 13:20:06 UTC

## R topics documented:

aws.transcribe-package . . . . .	2
get_transcription . . . . .	2
list_transcriptions . . . . .	3
start_transcription . . . . .	4
transcribeHTTP . . . . .	5

<b>Index</b>	<b>7</b>
--------------	----------

---

aws.transcribe-package

*aws.transcribe*

---

### Description

A Cloudyr Project Package

### Details

This is a template package for the cloudyr project

### Author(s)

Thomas J. Leeper <thosjleeper@gmail.com>

### See Also

[start\\_transcription](#), [list\\_transcriptions](#), [get\\_transcription](#)

---

get\_transcription

*Get AWS Transcribe Job*

---

### Description

Retrieve a specific AWS Transcribe job

### Usage

```
get_transcription(job, download = TRUE, ...)
```

### Arguments

job	A character string specifying the name of a job, possibly returned by <a href="#">list_transcriptions</a> .
download	A logical indicating whether to download the transcription(s).
...	Additional arguments passed to <a href="#">transcribeHTTP</a> .

### Value

A list.

### See Also

[start\\_transcription](#), [list\\_transcriptions](#)

## Examples

```
## Not run:
# start a transcription
## upload a file to S3
library("aws.s3")
put_object(file = "recording.mp3", bucket = "my-bucket", object = "recording.mp3")

## start transcription
start_transcription("first-example", "https://my-bucket.us-east-1.amazonaws.com/recording.mp3")

## wait
Sys.sleep(5)

## retrieve transcription
transcript <- get_transcription("first-example")
transcript$Transcriptions

## End(Not run)
```

---

list\_transcriptions    *List AWS Transcribe Jobs*

---

## Description

List AWS Transcribe jobs, by status

## Usage

```
list_transcriptions(
  status = c("COMPLETED", "IN_PROGRESS", "FAILED"),
  n = NULL,
  token = NULL,
  ...
)
```

## Arguments

status	A character string specifying the status of jobs to retrieve. Use <a href="#">get_transcription</a> to retrieve a specific transcription.
n	Optionally, a numeric value indicating the maximum number of results to return (for pagination).
token	Optionally, a “NextToken” indicating the next result to retrieve (for pagination).
...	Additional arguments passed to <a href="#">transcribeHTTP</a> .

## Value

A list.

## Examples

```
## Not run:  
list_transcriptions("COMPLETED")  
  
## End(Not run)
```

---

start\_transcription    *Start AWS Transcribe Job*

---

## Description

Start an AWS Transcribe job

## Usage

```
start_transcription(  
  name,  
  url,  
  format = tools::file_ext(url),  
  language = "en-US",  
  hertz = NULL,  
  ...  
)
```

## Arguments

name	A character string specifying a unique name for the transcription job.
url	A character string specifying a URL for the media file to be transcribed.
format	A character string specifying the file format. One of: "mp3", "mp4", "wav", "flac".
language	A character string specifying a language code. Currently defaults to "en-US".
hertz	Optionally, a numeric value specifying sample rate in Hertz.
...	Additional arguments passed to <a href="#">transcribeHTTP</a> .

## Value

A list containing details of the job. The transcript can be retrieved with [get\\_transcription](#).

## See Also

[get\\_transcription](#)

**Examples**

```
## Not run:
# start a transcription
## upload a file to S3
library("aws.s3")
put_object(file = "recording.mp3", bucket = "my-bucket", object = "recording.mp3")

## start trancription
start_transcription("first-example", "https://my-bucket.us-east-1.amazonaws.com/recording.mp3")

## End(Not run)
```

---

transcribeHTTP

*Execute AWS Transcribe API Request*


---

**Description**

This is the workhorse function to execute calls to the Transcribe API.

**Usage**

```
transcribeHTTP(
  action,
  headers = list(),
  query = list(),
  body = NULL,
  version = "v1",
  verbose = getOption("verbose", FALSE),
  region = Sys.getenv("AWS_DEFAULT_REGION", "us-east-1"),
  key = NULL,
  secret = NULL,
  session_token = NULL,
  ...
)
```

**Arguments**

action	A character string specifying an API endpoint.
headers	A list of headers to pass to the HTTP request.
query	An optional named list containing query string parameters and their character values.
body	A request body
version	A character string specifying the API version.
verbose	A logical indicating whether to be verbose. Default is given by options("verbose").
region	A character string specifying an AWS region. See <a href="#">locate_credentials</a> .

key	A character string specifying an AWS Access Key. See <a href="#">locate_credentials</a> .
secret	A character string specifying an AWS Secret Key. See <a href="#">locate_credentials</a> .
session_token	Optionally, a character string specifying an AWS temporary Session Token to use in signing a request. See <a href="#">locate_credentials</a> .
...	Additional arguments passed to <a href="#">GET</a> .

**Details**

This function constructs and signs an Transcribe API request and returns the results thereof, or relevant debugging information in the case of error.

**Value**

If successful, a named list. Otherwise, a data structure of class “aws-error” containing any error message(s) from AWS and information about the request attempt.

**Author(s)**

Thomas J. Leeper

# Index

## \*Topic **package**

aws.transcribe-package, [2](#)

aws.transcribe

(aws.transcribe-package), [2](#)

aws.transcribe-package, [2](#)

GET, [6](#)

get\_transcription, [2](#), [2](#), [3](#), [4](#)

list\_transcriptions, [2](#), [3](#)

locate\_credentials, [5](#), [6](#)

start\_transcription, [2](#), [4](#)

transcribeHTTP, [2-4](#), [5](#)