

# Package: speechmatics (via r-universe)

June 13, 2026

**Title** Client for the 'Speechmatics' Speech-to-Text API

**Version** 0.1.0

**Description** Transcribe audio files using the 'Speechmatics' speech-to-text API <<https://www.speechmatics.com/>>. Supports custom vocabulary, speaker diarization, punctuation control, and audio filtering.

**URL** <https://github.com/thisisnic/speechmatics>,  
<https://thisisnic.github.io/speechmatics/>

**BugReports** <https://github.com/thisisnic/speechmatics/issues>

**License** MIT + file LICENSE

**Imports** cli, curl, httr2 (>= 1.2.0), jsonlite, rlang

**Depends** R (>= 4.2.0)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 8.0.0

**Suggests** knitr, rmarkdown, testthat (>= 3.0.0), vcr (>= 2.0.0)

**VignetteBuilder** knitr

**Config/testthat/edition** 3

**Config/pak/sysreqs** libssl-dev

**Repository** <https://thisisnic.r-universe.dev>

**Date/Publication** 2026-06-06 18:46:35 UTC

**RemoteUrl** <https://github.com/thisisnic/speechmatics>

**RemoteRef** HEAD

**RemoteSha** f86d5200bb8c46b6dacedfee31c6945c9f60a154

## Contents

sm_delete_job . . . . .	2
sm_diarize_channel . . . . .	3
sm_diarize_speaker . . . . .	3
sm_get_transcript . . . . .	4
sm_list_jobs . . . . .	4
sm_transcribe . . . . .	5
sm_transcription_config . . . . .	6

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

---

sm_delete_job	<i>Delete a job</i>
---------------	---------------------

---

### Description

Delete a job

### Usage

```
sm_delete_job(job_id)
```

### Arguments

job_id	Job ID string.
--------	----------------

### Value

TRUE, invisibly.

### Examples

```
## Not run:
sm_delete_job("2om9psu1np")

## End(Not run)
```

---

sm\_diarize\_channel      *Configure channel diarization*

---

**Description**

Configure channel diarization

**Usage**

```
sm_diarize_channel(labels = NULL)
```

**Arguments**

labels                  Character vector of channel labels (e.g. c("Agent", "Caller")).

**Value**

A diarization config list.

**Examples**

```
sm_diarize_channel()  
  
# label channels  
sm_diarize_channel(labels = c("Nic", "Jess"))
```

---

sm\_diarize\_speaker      *Configure speaker diarization*

---

**Description**

Configure speaker diarization

**Usage**

```
sm_diarize_speaker(sensitivity = 0.5, prefer_current = FALSE)
```

**Arguments**

sensitivity      Numeric 0-1 (default 0.5). Higher values detect more speakers.  
prefer\_current   Logical. Reduces false switches between similar-sounding speakers.

**Value**

A diarization config list.

**Examples**

```
sm_diarize_speaker()
sm_diarize_speaker(sensitivity = 0.8, prefer_current = TRUE)
```

---

sm\_get\_transcript      *Get transcript for a completed job*

---

**Description**

Get transcript for a completed job

**Usage**

```
sm_get_transcript(job_id, format = c("txt", "srt", "json-v2"))
```

**Arguments**

job_id	Job ID string.
format	Output format: "txt" for plain text (default), "srt" for subtitles, or "json-v2" for structured data with timestamps, speaker labels, and confidence scores.

**Value**

Transcript as a character string ("txt" and "srt") or a list ("json-v2").

**Examples**

```
## Not run:
sm_get_transcript("2om9psu1np")
sm_get_transcript("2om9psu1np", format = "srt")
sm_get_transcript("2om9psu1np", format = "json-v2")

## End(Not run)
```

---

sm\_list\_jobs      *List all jobs*

---

**Description**

List all jobs

**Usage**

```
sm_list_jobs()
```

**Value**

A data frame of jobs.

**Examples**

```
## Not run:
sm_list_jobs()

## End(Not run)
```

---

sm_transcribe	<i>Transcribe an audio file</i>
---------------	---------------------------------

---

**Description**

Submits an audio file to the Speechmatics API, polls until complete, and writes the transcript to a file.

**Usage**

```
sm_transcribe(
  input,
  output = NULL,
  config = sm_transcription_config(),
  poll_interval = 5
)
```

**Arguments**

input	Path to the input audio file.
output	Path to the output transcript file. If NULL (the default), the output is written to the working directory with the same name as the input file but with a .txt extension.
config	Config from sm_transcription_config().
poll_interval	Seconds between status checks.

**Value**

The output path, invisibly.

**Examples**

```
## Not run:
audio <- system.file("extdata", "testrecording.mp3", package = "speechmatics")
sm_transcribe(audio)
sm_transcribe(audio, "output.txt")
sm_transcribe(audio, config = sm_transcription_config(quality = "enhanced"))

## End(Not run)
```

---

`sm_transcription_config`*Build transcription config*

---

**Description**

Build transcription config

**Usage**

```
sm_transcription_config(  
    language = "en",  
    quality = c("standard", "enhanced"),  
    diarization = NULL  
)
```

**Arguments**

language	Language code (default "en").
quality	"standard" or "enhanced".
diarization	A diarization object from <code>sm_diarize_speaker()</code> or <code>sm_diarize_channel()</code> , or NULL for no diarization.

**Value**

A config list suitable for passing to `sm_transcribe()`.

**Examples**

```
sm_transcription_config()  
sm_transcription_config(language = "fr", quality = "enhanced")  
sm_transcription_config(diarization = sm_diarize_speaker())
```

# Index

`sm_delete_job`, 2  
`sm_diarize_channel`, 3  
`sm_diarize_speaker`, 3  
`sm_get_transcript`, 4  
`sm_list_jobs`, 4  
`sm_transcribe`, 5  
`sm_transcribe()`, 6  
`sm_transcription_config`, 6