



**SPEECHMATICS**

SaaS 22.01

## Table of Contents

- [Release Notes](#)
  - [2022-08-11](#)
    - [What's Changed?](#)
  - [2022-06-21](#)
    - [What's Changed?](#)
  - [2022-05-25](#)
    - [What's Changed?](#)
  - [2022-03-24](#)
    - [What's Changed?](#)
  - [2022-03-16](#)
    - [What's Changed?](#)
  - [2021-12-13](#)
    - [What's Changed?](#)
  - [2021-09-07](#)
    - [What's Changed?](#)
  - [2021-08-23](#)
    - [What's Changed?](#)
  - [2021-08-10](#)
    - [What's Changed?](#)
  - [2021-03-24](#)
    - [What's Changed?](#)
  - [Known Issues](#)
    - [GDPR notice](#)
- [Introduction](#)
  - [Supported languages](#)
  - [Supported Domain Language Pack](#)
  - [Supported file types](#)
  - [File Size Limits](#)
  - [Data Retention Limits](#)
  - [Connecting to the Speechmatics SaaS](#)
  - [Authorization Token](#)
  - [Trial tokens](#)
  - [Supported endpoints](#)
  - [IP whitelisting for notifications](#)
  - [Rate limiting and fair usage](#)
  - [Timestamps](#)
    - [A note on tools](#)
  - [API Specification](#)
- [How to use the Speechmatics SaaS](#)
  - [Basics](#)
  - [Important Note on Examples](#)
    - [Submitting a job](#)
    - [Requesting an enhanced model](#)
    - [Linux example](#)
    - [Windows example](#)
  - [Checking Job Status](#)
    - [Unix/Ubuntu](#)
    - [Windows](#)
    - [The response](#)
  - [Poll for more than one job](#)
    - [Unix/Ubuntu](#)
    - [Windows](#)

- [The response](#)
  - [Retrieving a transcript](#)
    - [Unix/Ubuntu](#)
    - [Windows](#)
    - [Unix/Ubuntu](#)
    - [Windows](#)
  - [Deleting a completed job](#)
    - [Unix/Ubuntu](#)
    - [Windows](#)
    - [The Response](#)
  - [Deleting a running job](#)
    - [Unix/Ubuntu](#)
    - [Windows](#)
    - [The Response](#)
  - [The transcript](#)
    - [Polling call flow](#)
- [Formatting Common Entities](#)
  - [Overview](#)
  - [Supported Languages](#)
  - [Using the enable\\_entities parameter](#)
  - [Configuration example](#)
  - [Different entity classes](#)
  - [Output locale styling](#)
  - [Example output](#)
- [Configuring the transcript](#)
  - [Important note on examples](#)
  - [Fetch URL](#)
    - [Configuration example](#)
    - [Fetch failure](#)
    - [Unix/Ubuntu example](#)
    - [Windows example](#)
  - [Speaker separation \(Diarization\)](#)
    - [Speaker diarization](#)
      - [Speaker diarization tuning](#)
      - [Speaker diarization post-processing](#)
      - [Speaker diarization timeout](#)
  - [Channel diarization](#)
  - [Speaker change detection](#)
  - [Speaker change detection With Channel diarization](#)
  - [Custom dictionary](#)
  - [Output locale](#)
  - [Advanced punctuation](#)
  - [Notifications](#)
    - [Notification call flow](#)
    - [Notification configuration example](#)
    - [How to configure a notification](#)
    - [Accepting the notification](#)
    - [Notification webserver configuration](#)
    - [Notification failure](#)
  - [Metadata and job tracking](#)
    - [Metadata and job tracking configuration example](#)
  - [Word Tagging](#)
    - [Profanity Tagging](#)
    - [Disfluency Tagging](#)
  - [Domain Language Packs](#)

- [Usage](#)
  - [Overview](#)
  - [Requesting Usage](#)
    - [Linux example](#)
    - [Windows example](#)
    - [The response](#)
- [Troubleshooting](#)
  - [Basics](#)
  - [Raising an incident](#)
  - [Rate limited requests](#)
- [Speechmatics ASR REST API](#)
  - [Jobs API](#)
    - [Version: 2.7.0](#)
    - [/jobs](#)
    - [POST](#)
    - [Summary:](#)
    - [Parameters](#)
    - [Responses](#)
    - [GET](#)
    - [Summary:](#)
      - [Parameters](#)
      - [Responses](#)
    - [/jobs/{jobid}](#)
    - [GET](#)
    - [Summary:](#)
    - [Parameters](#)
    - [Responses](#)
    - [DELETE](#)
    - [Summary:](#)
    - [Parameters](#)
    - [Responses](#)
    - [/jobs/{jobid}/data](#)
    - [GET](#)
    - [Summary:](#)
    - [Parameters](#)
    - [Responses](#)
    - [/jobs/{jobid}/transcript](#)
    - [GET](#)
    - [Summary:](#)
    - [Parameters](#)
    - [Responses](#)
    - [/jobs/{jobid}/log](#)
      - [GET](#)
        - [Summary:](#)
        - [Parameters](#)
        - [Responses](#)
  - [Usage API](#)
    - [/usage](#)
    - [GET](#)
    - [Summary:](#)
    - [Parameters](#)
    - [Responses](#)
  - [Object Models](#)
    - [ErrorResponse](#)
    - [TrackingData](#)
    - [DataFetchConfig](#)
    - [TranscriptionConfig](#)

- [PunctuationConfig](#)
- [SpeakerDiarizationConfig](#)
- [NotificationConfig](#)
- [OutputConfig](#)
- [JobConfig](#)
- [CreateJobResponse](#)
- [JobDetails](#)
- [RetrieveJobsResponse](#)
- [RetrieveJobResponse](#)
- [DeleteJobResponse](#)
- [JobInfo](#)
- [RecognitionMetadata](#)
- [RecognitionDisplay](#)
- [RecognitionAlternative](#)
- [RecognitionResult](#)
- [RetrieveTranscriptResponse](#)
- [UsageResponse](#)
- [UsageSummaryResult](#)
- [UsageDetailsResult](#)

# Release Notes

This page documents production updates to SaaS, including the trial environment. This page contains information about new or updated features, bug fixes, known issues, and deprecated functionality.

## 2022-08-11

### What's Changed?

- New SaaS environments EU2 (eu2.asr.api.speechmatics.com) and US2 (us2.asr.api.speechmatics.com) available in EU and US regions respectively. Refer to [documentation here](#) for more details
- New egress IP addresses to whitelist notifications for new EU2 and US2 environments. Refer to [documentation here](#) for more details

## 2022-06-21

### What's Changed?

- New English finance domain language pack. Provides accuracy improvements when specific financial jargon is spoken in your audio. Refer to [documentation here](#) for more details
- 16 Languages updated with additional punctuation marks for improved readability
  - The following languages now support (. ? , !): Bulgarian, Catalan, Czech, Greek, Finnish, Croatian, Hungarian, Lithuanian, Latvian, Norwegian, Polish, Romanian, Slovak, Slovenian, Ukrainian, Korean
- Improved accuracy for French, including more data for Canadian French (fr-ca)
- Improved accuracy for Portuguese, including more data for Brazilian Portuguese (pt-br)
- Standard operating point improved accuracy for Romanian, Hungarian, Danish, Slovakian, Croatian, Bulgarian, Finnish, Slovenian, Lithuanian
- Updated Danish, Norwegian and Swedish to remove undesired character sets
- Improved accuracy in localised spelling for English output locale feature
- Fixes for English and Italian written form numeric entities
- Improved accuracy of percentage symbol recognition in French

## 2022-05-25

### What's Changed?

- New parameter added for controlling Speaker Diarization sensitivity: `speaker_sensitivity`. Refer to our [documentation here](#) for more details
- New Ukrainian (uk) language pack
- Resolved an issue where a small number of files with multiple audio channels were mistakenly detected as containing inverted audio, which lead to no transcription being returned. The check for inverted audio is now more robust.

## 2022-03-24

### What's Changed?

- Resolves an issue where Profanity and Disfluency Tagging were not output when Speaker Diarization was enabled

## 2022-03-16

### What's Changed?

- Improved accuracy for all 31 language packs. Gains will be for both standard and enhanced operating points
  - Biggest gains: Danish, Dutch, Norwegian, Lithuanian and Turkish
- New Cantonese (yue) and Indonesian (id) language packs
- Improved formatting of numeric entities such as dates, currencies and large numbers for 10 languages (cmn, de, en, es, fr, hi, it, ja, pt, ru, yue). Additional metadata about these entities can be requested by using the new `enable_entities` config parameter. For more information please see [our documentation here](#)

- Improvements to speaker diarization functionality in scenarios where two speakers are labelled when it is only a single speaker
- Improvements to custom dictionary functionality. Custom dictionary entries should now have less false positives
- Languages updated with additional punctuation marks
  - Japanese (。 \、 )
  - Italian (. ? , !)
  - Portuguese (. ? , !)
  - Russian (. ? , !)
  - Mandarin (。 ? ! \、 )
  - Hindi (। ? , !)
- The JSON-v2 output version is now 2.7
- Non-breaking spaces are now possible in a single word
- Speaker Diarization sensitivity parameters (previously deprecated in March 2021) are now removed from the API
  - Jobs will now be rejected if these parameters are included in the job config
  - This includes `speaker_diarization_params` , `new_speaker_sensitivity` , `segment_boundary_sensitivity`

## 2021-12-13

### What's Changed?

- New [usage](#) resource allowing you to retrieve details of your SaaS usage
- Option to cancel and delete a running job
- Updated IP address whitelist

## 2021-09-07

### What's Changed?

- Enhanced model available for all 31 language packs
  - Please contact your account manager if you would like access to the enhanced model
- General improvements in pop culture terms recognition for the English language pack
- Removal of foreign characters from English and German language packs

## 2021-08-23

### What's Changed?

- New language packs for all 31 language model. By default a language pack will contain a standard and enhanced model for all 31 languages. The standard model is now available to use, with no user change required. The enhanced model will be released in September. Please see the API how-to guide for how to request the enhanced model to prepare your integration in advance
- Profanity tagging in Italian and Spanish
- The Chinese Mandarin language pack now supports Traditional as well as Simplified Mandarin. Please see 'Configuring the Job Request' for guidelines of how to do so.

## 2021-08-10

### What's Changed?

- Error information added in API response for Fetch URL and Notification failures

## 2021-03-24

### What's Changed?

- Improved speaker diarization
  - Speaker diarization has been completely re-designed internally and should now be significantly more accurate
  - Instead of gendered speaker labels (M1, F2) speaker labels will be now (S1, S2 etc.) in the `json-v2` and `txt` output. Speaker gender identification is no longer a supported feature

- o If requesting an output in `txt` format, and requesting no diarization, there will be no `Speaker:UU` at the start of a transcript
- o Users may still request speaker diarization as before via the configuration object
- o Beta sensitivity parameters will be removed. The parameters will remain within the API but will not have any effect
- o This update to speaker diarization feature can mean the turnaround time for your transcript will in some cases take longer
- Improved Swedish and Arabic language packs, both now have advanced punctuation enabled (Swedish supports . ? , ! and Arabic supports . ؟ ؛ !)
- For the English language pack only, a new tag, `[disfluency]` has been added to a pre-set list of words that imply hesitation or interjection in the JSON-v2 output only. Examples include 'hmm' and 'umm'. Customers may use this tag to carry out their own post-processing
- The json-v2 API schema has been updated to v2.6

## Known Issues

| Issue ID  | Summary  | Detailed Description and Possible Workarounds  |
|-----------|--|--|
| REQ-20261 | The Japanese language pack may output fewer punctuation marks in certain scenarios | In some cases, users may see a decreased output in punctuation marks when transcribing in Japanese. Please report this if this is the case |

## GDPR notice

To help customers comply with data protection obligations from GDPR and other regulations, we assume that all media, transcript, and configuration files processed by the Speechmatics SaaS may contain personal data. Media, transcript, and configuration data are only processed to perform automated speech transcription following customer instructions conveyed via the cloud API.

All media, transcript, and configuration data will not be stored any longer than 7 days, and after this period they are deleted. This process will occur unless a user has explicitly deleted them through the API before they are deleted automatically. GET & DELETE request for jobs and/or media files more than 7 days after their submission or that have already been deleted will return a 4xx response.

Beyond the 7 day window, logs will still be present for troubleshooting and support purposes identifying whether features such as Custom Dictionary have been used but no information of its contents will be available.

Any URLs provided by users within the job config relating to fetching media or for notifications on the job are not recorded by logs. However, the client IP addresses are recorded.

## Introduction

This document will walk you through how to use the RESTful API for the Speechmatics SaaS. This document will show you

- How to authenticate and connect to the Speechmatics SaaS endpoint
- How to successfully submit a file for processing
- How to check on the status of any transcription job
- How to retrieve a transcript in the format of your choice

This section shows:

- Supported languages, domains and file formats
- How to use the Authorization Token
- How to connect to a valid Speechmatics endpoint
- Rate limits

## Supported languages

The Speechmatics SaaS supports the following languages. Your ability to use any or all of the languages will depend on what languages you are contracted to use.



Speechmatics takes a global first approach to our languages. In a single language pack we aim to support many different accents and dialects. This simplifies your workflow when selecting which language to use, not requiring you to know which accent is being spoken in your audio up-front. With this approach we still achieve very high accuracy compared to accent specific language packs.

| Language   | Language Code |
|------------|---------------|
| Arabic     | (ar)          |
| Bulgarian  | (bg)          |
| Cantonese  | (yue)         |
| Catalan    | (ca)          |
| Croatian   | (hr)          |
| Czech      | (cs)          |
| Danish     | (da)          |
| Dutch      | (nl)          |
| English    | (en)          |
| Finnish    | (fi)          |
| French     | (fr)          |
| German     | (de)          |
| Greek      | (el)          |
| Hindi      | (hi)          |
| Hungarian  | (hu)          |
| Italian    | (it)          |
| Indonesian | (id)          |
| Japanese   | (ja)          |
| Korean     | (ko)          |
| Latvian    | (lv)          |
| Lithuanian | (lt)          |
| Malay      | (ms)          |
| Mandarin   | (cmn)         |
| Norwegian  | (no)          |
| Polish     | (pl)          |
| Portuguese | (pt)          |
| Romanian   | (ro)          |
| Russian    | (ru)          |
| Slovakian  | (sk)          |
| Slovenian  | (sl)          |
| Spanish    | (es)          |
| Swedish    | (sv)          |

|           |      |
|-----------|------|
| Turkish   | (tr) |
| Ukrainian | (uk) |

Please note any languages outside this list are not explicitly supported. Only one language can be processed within each request. Each language above has a two-letter ISO639-1 code that must be provided for any transcription request.

## Supported Domain Language Pack

The Speechmatics SaaS also supports specialized language packs that enhance the requested transcription language with optimization for a particular field. This is particularly useful for increasing the accuracy for domains that have specific terminology. The domain packs build on our global languages to give the best accuracy.

| Domain  | Supported languages | Description  |
|---------|---------------------|--|
| Finance | (en)                | Improve accuracy for audio containing financial terms such as those found in earnings calls or financial broadcast |

Refer to [documentation here](#) for more details on how to set the domain parameter.

## Supported file types

The Speechmatics SaaS supports the following file types for transcription:

- aac
- amr
- flac
- m4a
- mp3
- mp4
- mpeg
- ogg
- wav

The list above is exhaustive - any file format outside the list above is explicitly not supported.

## File Size Limits

The supported size limit for jobs is 2 hours of audio or 1 GB file size. Any larger or longer files may be rejected.

## Data Retention Limits

Audio files, transcripts, and configuration data are stored in the Speechmatics SaaS for 7 days. Any request to retrieve a transcript or file more than 7 days after it was processed will receive a HTTP 404 error message and a status of `expired`.

You can delete audio or transcripts in advance of this 7 day period - how to do so is documented in the How-To Guide

## Connecting to the Speechmatics SaaS

This section talks through the Pre-Requisites to authenticate to the Speechmatics SaaS

### Authorization Token

Speechmatics Support will provide you a unique Authorization Token. This must be used with any interaction with the Speechmatics SaaS to authenticate to the service. Any interaction without this token will receive a `HTTP 401 - Unauthorized` response.

The Authorization Token must be passed in the header of any request to the Speechmatics SaaS. The Token itself conforms to OAuth2 protocols.

It is your responsibility to store the Authorization Token securely and ensure it is not mishandled. It is recommended that you store and provide access to the Token on the principle of least privilege. If you believe that your Token has been compromised, please contact Speechmatics Support.

## Trial tokens

If you are using the trial Speechmatics endpoint, please note your authorization token may be time limited. If you try and use the token after the trial period has expired, you will receive a `HTTP 401 - Unauthorized` response.

## Supported endpoints

Speechmatics SaaS supports the following endpoints for production use:

| Region | Environment | Endpoints  |
|--------|-------------|--|
| EU     | EU1         | <b>eu.asr.api.speechmatics.com</b><br><b>eu1.asr.api.speechmatics.com</b><br><b>asr.api.speechmatics.com</b> |
| EU     | EU2         | <b>eu2.asr.api.speechmatics.com</b>  |
| US     | US1         | <b>us.asr.api.speechmatics.com</b><br><b>us1.asr.api.speechmatics.com</b>                                    |
| US     | US2         | <b>us2.asr.api.speechmatics.com</b>  |

Authorization Tokens are replicated between all environments in the same region. Therefore, you can use any environment in a region that you are entitled to access.

All production environments are active and highly available. Multiple environments can be used to balance requests or provide a failover in the event of disruption to one environment.

Note that jobs are created in the environment corresponding to the endpoint used. You must use the same endpoint for all requests relating to a specific job.

If you attempt to use an endpoint for a region you are not contracted to use, that request will be unsuccessful. If you want to use a different region, please contact [sales@speechmatics.com](mailto:sales@speechmatics.com).

The Trial endpoint is **trial.asr.api.speechmatics.com**

## IP whitelisting for notifications

If you want to receive notifications from the Speechmatics SaaS, you will need to whitelist the following egress IP addresses. You should only whitelist the IP addresses from the geo region you are contracted to use

The list is below:

| Regions | IP Addresses   |
|---------|--|
| EU1     | 40.74.41.91, 52.236.157.154, 40.74.37.0, 20.73.209.153, 20.73.142.44     |
| EU2     | 20.105.89.153, 20.105.89.173, 20.105.89.184, 20.105.89.98, 20.105.88.228 |
| US1     | 52.149.21.32, 52.149.21.10, 52.137.102.83, 40.64.107.92, 40.64.107.99    |
| US2     | 52.146.58.224, 52.146.58.159, 52.146.59.242, 52.146.59.213, 52.146.58.64 |
| Trial   | 20.101.4.47, 20.93.251.234   |

## Rate limiting and fair usage

Speechmatics SaaS applies rate limiting and fair queueing to provide a consistently high quality of service to all users.

If you make a large number of requests in a short period of time, some of these requests may fail with the response `HTTP 429 - Rate Limited`. To minimise the possibility of encountering rate limiting errors, we recommend that you do not exceed the following rates:

- 10 new jobs per second ( `POST` API calls).
- 50 job status requests per second ( `GET` API calls).

Aside from rate limiting, there is no limit to the number of jobs that you can submit. However, Speechmatics SaaS applies a fair queueing policy which means that if you have a large number of jobs in progress at one time, the most recently submitted jobs may take longer to complete.

If you require a service which exceeds the recommendations above, please contact [support@speechmatics.com](mailto:support@speechmatics.com)

## Timestamps

Please note: all timestamps from the Speechmatics SaaS are in ISO 8601 format, and use Coordinated Universal Time (UTC).

### A note on tools

Code samples in this guide use [curl](#) for making HTTP requests to the API, and the [jq](#) tool to parse and display JSON responses.

## API Specification

The API uses a standardised authentication method, with a wide range of supported output formats and has a structured JSON configuration that is easier to extend in the future.

For the full API specification please refer to the [API Reference](#) section.

# How to use the Speechmatics SaaS

## Basics

This section will show how to submit an audio file for batch transcription, check the status of a job, and to retrieve a transcript in a supported format once the job is done. It then shows how to configure a transcription request to take advantage of other Speechmatics features.

This section will show you

- How to submit a file to the Speechmatics SaaS
- How to poll on any job's status
- How to retrieve a transcript in a supported format
- How to delete a job and all associated data

In all requests your Authorization Token must always be passed in the header of any request

For these examples there are linked videos, and examples provided for Windows and for Linux (Ubuntu).

## Important Note on Examples

All examples below show, where relevant, an example job configuration, and how to submit, retrieve and delete job requests and their output via Linux or Windows Job configuration itself in these examples is enclosed within a file called `config.json` when submitted to the Speechmatics SaaS. These examples should work in Windows CME mode and Powershell

### Submitting a job

Speechmatics SaaS Demo: [Post Job Request](#)

!(<https://www.youtube.com/watch?v=8PgldlXl36M>)

The simplest configuration for a transcription job is simply to specify:

- The `type` of request you want. This is always `transcription`
- The `language` that you wish to use. This is a two digit language code following ISO639-1 format. It must be one of the language codes supported by Speechmatics for transcription:

Below is an example of a basic configuration

```
config='{
  "type": "transcription",
  "transcription_config": { "language": "en" }
}'
```

In examples below, job configuration is enclosed within a file called `config.json`. This file can be extended or modified as you wish when using Speechmatics features

### Requesting an enhanced model

Speechmatics supports two different models within each language pack; a standard or an enhanced model. The standard model is the faster of the two, whilst the enhanced model provides a higher accuracy, but a slower turnaround time.

The enhanced model is a premium model. Please contact your account manager or Speechmatics if you would like access to this feature.

Free trial users can access both enhanced and standard models without first speaking to a member of the Speechmatics team.

An example of requesting the enhanced model is below

```
{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "operating_point": "enhanced"
  }
}
```

Please note: `standard`, as well as being the default option, can also be explicitly requested with the `operating_point` parameter.

### Linux example

```
curl -L -X POST https://asr.api.speechmatics.com/v2/jobs/ -H "Authorization: Bearer NDFjOTE3NGEtOWVm" -F data_file=@example.wav -F config="$(cat config.json)" | jq
```

### Windows example

```
curl.exe -L -X POST https://asr.api.speechmatics.com/v2/jobs/ -H "Authorization: Bearer NDFjOTE3NGEtOWVm" -F data_file=@example.wav -F config="<config.json" | jq
```

Here `NDFjOTE3NGEtOWVm` is a sample authorization token. You would replace this with the token that you have been provided with.

A successful request will return a HTTP 201 response, and will contain a unique 10-digit alphanumeric Job ID, which will be returned as `id` in the HTTP response. For full details see the [Jobs API Reference](#).

An example response will look like this:

```
HTTP/1.1 201 Created
Content-Length: 20
Content-Type: application/json
Request-Id: 4d46aa73e1a4c5a6d4ba6c31369e7b2e
Strict-Transport-Security: max-age=15724800; includeSubDomains
```

```
X-Azure-Ref:
0XdIUxQAAAAD1flaR0qvRQpReZYf+q+FHTE9OMjFFRedFMDMwNwBhN2JjOWQ4MC02YjBiLTQ1NWEtYjE3MS01NGJkZmNiYWWE0YTk=

Date: Thu, 27 Jun 2019 14:27:45 GMT

{"id":"dlhsd8d69i"}
```

In addition to the Job ID The `Request-ID` uniquely identifies your API request. If you ever need to raise a support ticket we recommend that you include the `Request-ID` if possible, as it will help to identify your job. The `Strict-Transport-Security` response header indicates that only HTTPS access is allowed. The `X-Azure-Ref` response header identifies the load-balancer request.

If you do not specify an auth token in the request, or if the token provided is invalid, then you'll see a 401 status code like this:

```
HTTP/1.1 401 Authorization Required
```

## Checking Job Status

A `201 Created` status indicates that the job was accepted for processing. If you wish to retrieve a particular job, you can do so using the job id for up to 7 days, after which time it will be automatically deleted in accordance with our data retention policy. You can make a GET request to check the status of that job as follows:

### Unix/Ubuntu

```
curl -L -X GET https://asr.api.speechmatics.com/v2/jobs/dlhsd8d69i -H "Authorization: Bearer
NDFjOTE3NGEtOWVm" | jq
```

### Windows

```
curl.exe -L -X GET https://asr.api.speechmatics.com/v2/jobs/dlhsd8d69i -H "Authorization: Bearer
NDFjOTE3NGEtOWVm" | jq
```

### The response

The response is a JSON object containing details of the job, with the `status` field showing whether the job is still processing or not. A value of `done` means that the transcript is now ready. Jobs will typically take less than half the audio length to process; so an audio file that is 40 minutes in length should be ready within 20 minutes of submission. See [Jobs API Reference](#) for details. An example response looks like this:

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  "job": {
    "config": {
      "notification_config": null,
      "transcription_config": {
        "additional_vocab": null,
        "channel_diarization_labels": null,
        "language": "en"
      },
    },
    "type": "transcription"
  },
  "created_at": "2019-01-17T17:50:54.113Z",
  "data_name": "example.wav",
  "duration": 275,
  "id": "yjbmf9kqub",
  "status": "running"
}
```

```
}  
}
```

Possible job status types include:

- running
- done
- rejected

## Poll for more than one job

If you have submitted multiple jobs, you can retrieve a list of the 100 most recent jobs submitted in the past 7 days by making a GET request without a Job ID. If a job has been deleted it will not be included in the list.

### Unix/Ubuntu

```
curl -L -X GET https://asr.api.speechmatics.com/v2/jobs/ -H "Authorization: Bearer  
NDFjOTE3NGEtOWVm" | jq
```

### Windows

```
curl.exe -L -X GET https://asr.api.speechmatics.com/v2/jobs/ -H "Authorization: Bearer  
NDFjOTE3NGEtOWVm" | jq
```

## The response

Here is an example result:

```
{  
  "jobs": [  
    {  
      "config": {  
        "transcription_config": {  
          "language": "en"  
        },  
        "type": "transcription"  
      },  
      "created_at": "2021-02-03T17:16:05.500Z",  
      "data_name": "audio.mp3",  
      "duration": 143,  
      "id": "bhyy42zccb",  
      "status": "done"  
    },  
    {  
      "config": {  
        "transcription_config": {  
          "language": "en"  
        },  
        "type": "transcription"  
      },  
      "created_at": "2021-02-03T17:16:04.583Z",  
      "data_name": "audio.mp3",  
      "duration": 146,  
      "id": "zg4m8z4jtn",  
      "status": "done"  
    }  
  ]  
}
```

## Retrieving a transcript

Speechmatics SaaS Demo: [Retrieving Transcripts](#)

!([https://www.youtube.com/watch?v=-YqjJXDS\\_g](https://www.youtube.com/watch?v=-YqjJXDS_g))

Transcripts can be retrieved as follows:

### Unix/Ubuntu

```
curl -L -X GET https://asr.api.speechmatics.com/v2/jobs/yjbm9kqub/transcript -H "Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

### Windows

```
curl.exe -L -X GET https://asr.api.speechmatics.com/v2/jobs/yjbm9kqub/transcript -H "Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

The supported default transcription output is `json-v2`. Other formats supported are `srt` (SubRip subtitle format) and `txt` (plain text). The format is set using the `format` query string parameter. Below are examples for retrieving a transcript in TXT format:

### Unix/Ubuntu

```
curl -L -X GET "https://asr.api.speechmatics.com/v2/jobs/yjbm9kqub/transcript?format=txt" -H "Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

### Windows

```
curl.exe -L -X GET https://asr.api.speechmatics.com/v2/jobs/yjbm9kqub/transcript?format=txt -H "Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

See [Jobs API Reference](#) for details. If you require multiple transcripts simultaneously, this is supported via the notifications functionality in the configuration object and is shown [here](#)

## Deleting a completed job

Jobs will be automatically deleted after 7 days in accordance with our data retention policy. You can remove a completed job before it is automatically deleted as follows:

### Unix/Ubuntu

```
curl -L -X DELETE https://asr.api.speechmatics.com/v2/jobs/yjbm9kqub -H "Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

### Windows

```
curl.exe -L -X DELETE https://asr.api.speechmatics.com/v2/jobs/yjbm9kqub -H "Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

## The Response

If the DELETE request is successful, you will receive a response, showing the jobID and a status of `deleted`

```
{
  "job": {
    "config": {
      "transcription_config": {
```



```

        "language": "en"
      },
      "type": "transcription"
    },
    "created_at": "2021-02-19T09:47:09.561Z",
    "data_name": "bbcnews.mp3",
    "duration": 377,
    "id": "ovrdc3e13w",
    "status": "deleted"
  }
}

```

## Deleting a running job

A DELETE request with no query parameters will only remove completed jobs. To remove a job that may still be running, you can use DELETE with the `force=true` parameter as follows:

### Unix/Ubuntu

```
curl -L -X DELETE https://asr.api.speechmatics.com/v2/jobs/yjbmF9kqub?force=true -H "Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

### Windows

```
curl.exe -L -X DELETE https://asr.api.speechmatics.com/v2/jobs/yjbmF9kqub?force=true -H "Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

## The Response

If the DELETE request is successful, you will receive a response, showing the jobID and a status of `deleted`

```

{
  "job": {
    "config": {
      "transcription_config": {
        "language": "en"
      },
      "type": "transcription"
    },
    "created_at": "2021-02-19T09:47:09.561Z",
    "data_name": "bbcnews.mp3",
    "duration": 377,
    "id": "ovrdc3e13w",
    "status": "deleted"
  }
}

```

## The transcript

When retrieving the transcript, and requesting `json-v2` format, the following information is returned

- The format. The API format. This will be 2.7
- The `job` property. This contains information like:
  - `created_at` : The timestamp for when the job was created, in UTC
  - `data_name` : The file name of the submitted media
  - `duration` : The length of the file in seconds
  - `id` : The unique Job ID associated with the file

- The `metadata` property. This includes:
  - `created_at` : when the transcription output was created. Note: this will be a later time to the `created_at` result in `job`, which takes into account time taken to process the job and other system interactions
  - `transcription_config` : The configuration requested when the job was submitted
- The `results` section. This will contain:
  - `type` : One of 'word', 'entity', 'punctuation' or 'speaker\_change'
  - `start_time` : The start of the word or punctuation, marked in seconds
  - `end_time` : The end of the word or punctuation, marked in seconds
  - `is_eos` : Whether a punctuation mark is the end of the sentence or not. The value here is boolean (e.g. true or false). This value will not always be shown
  - `alternatives` : For 'word' and 'punctuation' results this contains a list of possible alternative options for the word/symbol
    - `content` : A word or punctuation mark. When `enable_entities` is requested this can be multiple words with spaces, for example "17th of January 2022".
    - `confidence` : This is a score between 0 and 1.
    - `speaker` : If speaker diarization is enabled, which speaker is talking. Speakers are numbered `s1`, `s2`, `s3` etc. If diarization is not chosen or fails to detect a speaker the results will be `UU`
    - `language` : the language spoken. This will always be in the language that you have requested
    - `tags` : any metadata tag about the type of word spoken. This will be `profanity` for a set list of words in English only
    - `channel` : Only shown if channel diarization is enabled. The default value is `channel_#`, where `#` is an integer that corresponds to the channel number in the audio file. Up to 6 channels in one file are supported. If channel diarization labels are requested in the job request (e.g. `caller`, `agent`) these would be shown in the `channel` value instead and override the default
  - `entity_class` : Only if `enable_entities` is requested and an entity is detected, `entity_class` will represent the type of entity the word(s) have been formatted as
  - `spoken_form` : Only if `enable_entities` is requested and an entity is detected, this is a list of words without formatting applied. This follows the results list API formatting.
  - `written_form` : Only if `enable_entities` is requested and an entity is detected, this is a list of formatted words that matches the entity content but with individual estimated timing and confidences. This follows the results list API formatting.

An example transcript using the JSON output format is shown:

```
HTTP/1.1 200 OK
Transfer-Encoding: chunked
Content-Type: application/vnd.speechmatics.v2+json
Request-Id: aad4eb68bca69f3f277d202456bb0b15
Strict-Transport-Security: max-age=15724800; includeSubDomains
X-Azure-Ref:
OztQUXQAAAAA4ifGtw2COQKyb52QoNgX4TE9OMjFFREdFMDMyMABhN2JjOWQ4MCO2YjBiLTQ1NWEtYjE3MS01NGJkZmNiYWEOYTk=

Date: Thu, 27 Jun 2019 14:38:05 GMT

{
  "format": "2.7",
  "job": {
    "created_at": "2019-01-17T17:50:54.113Z",
    "data_name": "example.wav",
    "duration": 275,
    "id": "yjbmf9kqub"
  },
  "metadata": {
    "created_at": "2019-01-17T17:52:26.222Z",
    "transcription_config": {
      "diarization": "none",
```

```

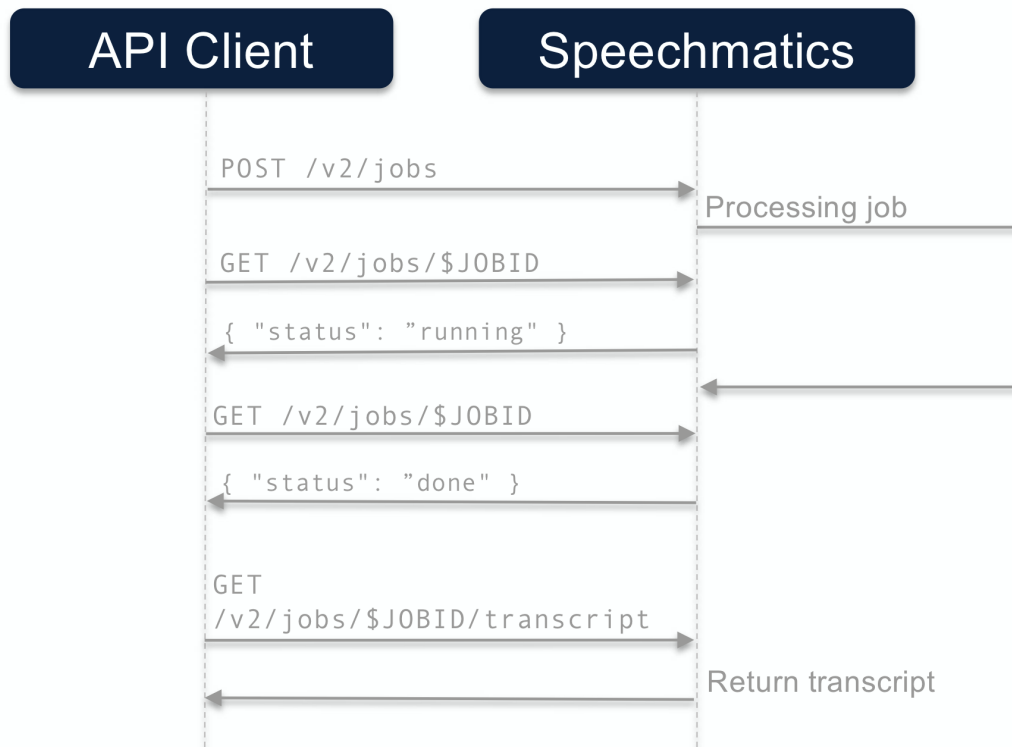
    "language":"en"
  },
  "type":"transcription"
},
"results":[
  {
    "alternatives":[
      {
        "confidence":0.9,
        "content":"Just",
        "language":"en",
        "speaker":"UU"
      }
    ],
    "end_time":1.07,
    "start_time":0.9,
    "type":"word"
  },
  {
    "alternatives":[
      {
        "confidence":1,
        "content":"this",
        "language":"en",
        "speaker":"UU"
      }
    ],
    "end_time":1.44,
    "start_time":1.11,
    "type":"word"
  },
  {
    "alternatives":[
      {
        "confidence":1,
        "content":".",
        "language":"en",
        "speaker":"UU"
      }
    ],
    "end_time":273.64,
    "start_time":273.64,
    "type":"punctuation"
  }
]
}

```

:::warning use of non-ASCII output You should be aware that for most non-English languages, you will be working with characters outside the ASCII range. Ensure that the programming language or client framework you are using is able to output the human-readable or machine-readable format that you require for your use case. Some client bindings will assume that non-ASCII characters are escaped, others do not. In most cases there will be a parameter that enables you to decide which you want to output. :::

### Polling call flow

The call flow for the polling method looks like this:



## Formatting Common Entities

### Overview

Entities are commonly recognisable classes of information that appear in languages, for example numbers and dates. Formatting these entities is commonly referred to as Inverse Text Normalisation (ITN). Speechmatics will output entities in a predictable, consistent written form, reducing post-processing work required aiming to make the transcript more readable.

The language pack will use these formatted entities by default in the transcription for all outputs (JSON, text and srt). Additional metadata about these entities can be requested via the API including the spoken words without formatting and the entity class that was used to format it.

### Supported Languages

Entities are supported in the following languages:

- Cantonese
- Chinese Mandarin (Simplified and Traditional)
- English
- French
- German
- Hindi
- Italian
- Japanese
- Portuguese
- Russian
- Spanish

### Using the `enable_entities` parameter

Speechmatics now includes an `enable_entities` parameter. This can be requested via the API. By default this is `false`.

Changing `enable_entities` to `true` will enable a richer set of metadata in the JSON output only. Customers can choose between the default written form, spoken form, or a mixture, for their own workflows.

The changes are as following:

- A new `type - entity` in the JSON output in addition to `word` and `punctuation`. For example: "1.99" would have a `type of entity` and a corresponding `entity_class` of `decimal`
- The `entity` will contain the formatted text in the `content` section, like other words and punctuation
  - The `content` can include spaces, non-breaking spaces, and symbols (e.g. `$/£/%`)
- A new output element, `entity_class` has been introduced. This provides more detail about how the entity has been formatted. A full list of entity classes is provided below.
- The start and end time of the entity will span all the words that make up that entity
- The entity also contains two ways that the content will be output:
  - `spoken_form` - Each individual `word` within the entity, written out in words as it was spoken. Each individual word has its own start time, end time, and confidence score. For example: "one", "million", "dollars"
  - `written_form` - The same output as within `entity` content, with a `type of word` instead. If there are spaces in the content it will be split into individual words. For example: "\$1", "million"

## Configuration example

Please see an example configuration file that would request entities:

```
{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "enable_entities": true
  }
}
```

## Different entity classes

The following `entity_classes` can be returned. Entity classes indicate how the numerals are formatted. In some cases, the choice of class can be contextual and the class may not be what was expected (for example "2001" may be a "cardinal" instead of "date"). The number of `entity_classes` may grow or shrink in the future.

N.B. Please note existing behaviour for English where numbers from zero to 10 (excluding where they are output as a decimal/money/percentage) are output as **words** is unchanged.

| Entity Class | Formatting Behaviour   | Spoken Word Form Example  | Written Form Example |
|--------------|--|---|----------------------|
| alphanum     | A series of three or more alphanumerics, where an alphanumeric is a digit less than 10, a character or symbol        | triple seven five four  | 77754                |
| cardinal     | Any number greater than ten is converted to numbers. Numbers ten or below remain as words. Includes negative numbers | nineteen  | 19                   |
| credit card  | A long series of spoken digits less than 10 are converted to numbers. Support for common credit cards                | one one one one two two two two three three three three four four four four | 1111222233334444     |
| date         | Day, month and year, or a year on its own. Any words spoken in the date are maintained (including "the" and "of")    | fifteenth of January twenty twenty two                                      | 15th of January 2022 |

|            |   |   |                     |
|------------|---|---|---------------------|
| decimal    | A series of numbers divided by a separator  | eighteen point one two                    | 18.12               |
| fraction   | Small fractions are kept as words ("half"), complex fractions are converted to numbers separated by "/" | three sixteenths                          | 3/16                |
| money      | Currency words are converted to symbols before or after the number (depending on the language)          | twenty dollars                            | \$20                |
| ordinal    | Ordinals greater than 10 are output as numbers  | forty second                              | 42nd                |
| percentage | Numbers with a per cent have the per cent converted to a % symbol                                       | duecento percento                         | 200%                |
| span       | A range expressed as "x to y" where x and y correspond to another entity class                          | one hundred to two hundred million pounds | 100 to £200 million |
| time       | Times are converted to numbers  | eleven forty a m                          | 11:40 a.m.          |
| word       | Entities that do not match a specific class   | hundreds                                  | hundreds            |

## Output locale styling

Each language has a specific style applied to it for thousands, decimals and where the symbol is positioned for money or percentages.

For example

- English contains commas as separators for numbers above 9999 (example: "20,000"), the money symbol at the start (example: "\$10") and full stops for decimals (example: "10.5")
- German contains full stops as separators for numbers above 9999 (example: "20.000"), the money symbol comes after with a non-breaking space (example: "10 \$") and commas for decimals (example: "10,5")
- French contains non-breaking spaces as separators for numbers above 9999 (example: "20 000"), the money symbol comes after with a non-breaking space (example: "10 \$") and commas for decimals (example: "10,5")

## Example output

Here is an example of a transcript requested with `enable_entities` set to true:

- An `entity` that is "17th of January 2022", including spaces
  - The start and end times span the entire entity
  - An `entity_class` of `date`
  - The `spoken_form` is split into the following individual words: "seventeenth", "of", "January", "twenty", "twenty", "two". Each word has its own start and end time
  - the `written_form` split into the following individual words: "17th", "of", "January", "2022". Each word has its own start and end time

Note:

- By default and when speaker diarization is enabled, `speaker` parameter is added per word within the entity, spoken and written form
- When channel diarization is enabled, `channel` parameter is only added on the `results` parent within the entity and not included in spoken and written form

```
"results": [
  {
    "alternatives": [
      {
        "confidence": 0.99,
        "content": "17th of January 2022",
```

```

    "language": "en",
    "speaker": "UU"
  }
],
"end_time": 3.14,
"entity_class": "date",
"spoken_form": [
  {
    "alternatives": [
      {
        "confidence": 1.0,
        "content": "seventeenth",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 1.41,
    "start_time": 0.72,
    "type": "word"
  },
  {
    "alternatives": [
      {
        "confidence": 1.0,
        "content": "of",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 1.53,
    "start_time": 1.41,
    "type": "word"
  },
  {
    "alternatives": [
      {
        "confidence": 1.0,
        "content": "January",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 2.04,
    "start_time": 1.53,
    "type": "word"
  },
  {
    "alternatives": [
      {
        "confidence": 1.0,
        "content": "twenty",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 2.46,
    "start_time": 2.04,
    "type": "word"
  },

```

```

{
  "alternatives": [
    {
      "confidence": 1.0,
      "content": "twenty",
      "language": "en",
      "speaker": "UU"
    }
  ],
  "end_time": 2.79,
  "start_time": 2.46,
  "type": "word"
},
{
  "alternatives": [
    {
      "confidence": 0.97,
      "content": "two",
      "language": "en",
      "speaker": "UU"
    }
  ],
  "end_time": 3.14,
  "start_time": 2.79,
  "type": "word"
}
],
"start_time": 0.72,
"type": "entity",
"written_form": [
  {
    "alternatives": [
      {
        "confidence": 0.99,
        "content": "17th",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 1.33,
    "start_time": 0.72,
    "type": "word"
  },
  {
    "alternatives": [
      {
        "confidence": 0.99,
        "content": "of",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 1.93,
    "start_time": 1.33,
    "type": "word"
  },
  {
    "alternatives": [
      {

```



```

        "confidence": 0.99,
        "content": "January",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 2.54,
    "start_time": 1.93,
    "type": "word"
  },
  {
    "alternatives": [
      {
        "confidence": 0.99,
        "content": "2022",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 3.14,
    "start_time": 2.54,
    "type": "word"
  }
]
}
]

```

If `enable_entities` is set to `false`, the output is as below:

```

"results": [
  {
    "alternatives": [
      {
        "confidence": 0.99,
        "content": "17th",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 1.33,
    "start_time": 0.72,
    "type": "word"
  },
  {
    "alternatives": [
      {
        "confidence": 0.99,
        "content": "of",
        "language": "en",
        "speaker": "UU"
      }
    ],
    "end_time": 1.93,
    "start_time": 1.33,
    "type": "word"
  },
  {
    "alternatives": [
      {

```

```

        "confidence": 0.99,
        "content": "January",
        "language": "en",
        "speaker": "UU"
    }
],
"end_time": 2.54,
"start_time": 1.93,
"type": "word"
},
{
    "alternatives": [
        {
            "confidence": 0.99,
            "content": "2022",
            "language": "en",
            "speaker": "UU"
        }
    ],
    "end_time": 3.14,
    "start_time": 2.54,
    "type": "word"
}
]
}

```

## Configuring the transcript

So far you should know

- How to connect and authenticate to the Speechmatics SaaS
- How to successfully submit a media file for processing
- How to check the status of one or multiple jobs
- How to retrieve the transcript, and interpret the output

The following section will show you how to configure your requests to take advantage of Speechmatics features.

### Important note on examples

All examples below show the configuration separately to the actual HTTP Submission request. This configuration can be enclosed within a file (e.g. `config.json`). How to submit this configuration file as part of a successful job request is shown for both Linux and Windows. Examples for Windows should work in both CMD mode and Powershell.

### Fetch URL

Speechmatics SaaS Demo: [Fetch URL Callback](#)

!([https://www.youtube.com/watch?v=5htoThz5\\_W8](https://www.youtube.com/watch?v=5htoThz5_W8))

Examples so far have shown to submit a job where a media file is locally uploaded. If you store your digital media in cloud storage (for example AWS S3 or Azure Blob Storage) you can also submit a job by providing the URL of the audio file. The configuration uses a `fetch_data` section, which looks like this:

### Configuration example

```

{
  "type": "transcription",
  "transcription_config": {
    "language": "en",

```

```

    "diarization": "speaker"
  },
  "fetch_data": {
    "url": "${URL}/{FILENAME}"
  }
}

```

## Fetch failure

If the Speechmatics SaaS is unable to retrieve audio from the specified online location, the job will fail, with a `status` of `rejected`, and no transcript will be generated. Users can now retrieve failure information by making a GET `/jobs/$JOBID` request, and use that to carry out diagnostic information.

If the job has failed, there will be an additional `errors` element, which will show all failure messages Speechmatics SaaS encountered when carrying out the fetch request. Please note, there can be multiple failure attempts associated with one submitted job, as there is a retry mechanism in place.

```

{
  "job": {
    "config": {
      "fetch_data": {
        "url": "https://example.com/average-files/punctuation1.mp3"
      },
      "notification_config": [
        {
          "contents": [
            "jobinfo"
          ],
          "url": "https://example.com/"
        }
      ],
      "transcription_config": {
        "language": "de"
      },
      "type": "transcription"
    },
    "created_at": "2021-07-19T12:55:03.754Z",
    "data_name": "",
    "duration": 0,
    "errors": [
      {
        "message": "unable to fetch audio: http status code 404",
        "timestamp": "2021-07-19T12:55:05.425Z"
      },
      {
        "message": "unable to fetch audio: http status code 404",
        "timestamp": "2021-07-19T12:55:07.649Z"
      },
      {
        "message": "unable to fetch audio: http status code 404",
        "timestamp": "2021-07-19T12:55:17.665Z"
      },
      {
        "message": "unable to fetch audio: http status code 404",
        "timestamp": "2021-07-19T12:55:37.643Z"
      }
    ],
    "id": "a81ko4eqj1",
    "status": "rejected"
  }
}

```

```
}  
}
```

## Unix/Ubuntu example

```
curl -L -X POST https://asr.api.speechmatics.com/v2/jobs/ -H "Authorization: Bearer  
NDFjOTE3NGEtOWVm" -F config="$(cat config.json)"
```

## Windows example

```
curl.exe -L -X POST https://asr.api.speechmatics.com/v2/jobs/ -H "Authorization: Bearer  
NDFjOTE3NGEtOWVm" -F config="<config.json"
```

Some cloud storage solutions may require authentication. You can use the `auth_headers` property in the `fetch_data` section to provide the headers necessary to access the resource.

Ensure that any URLs you generate - for example using [AWS pre-signed URLs](#) or Microsoft Azure [Shared Access Signatures](#) - have not expired, or the job will be rejected.

## Speaker separation (Diarization)

Speechmatics offers four different modes for separating out different speakers in the audio:

| Type                                 | Description  | Use Case  |
|--------------------------------------|--|---|
| speaker diarization                  | Aggregates all audio channels into a single stream for processing and picks out unique speakers based on acoustic matching.      | Used in cases where there are multiple speakers embedded in the same audio recording and it's required to understand what each unique speaker said. |
| channel diarization                  | Transcribes each audio channel separately and treats each channel as a unique speaker.   | Used when it's possible to record each speaker on separate audio channels.  |
| speaker change (beta)                | Provides the point in transcription when there is believed to be a new speaker.  | Used for when you just need to know the speaker has changed usually in a real-time application.   |
| channel diarization & speaker change | Transcribes each audio channel separately and within each channel provides the point when there is believed to be a new speaker. | Used when it's possible to record some speakers on a separate audio channel, but some channels there are multiple speakers.                         |

Each of these modes can be enabled by using the `diarization` config. The following are valid values:

The default value is `none` - e.g. the transcript will not be diarized.

| Type                                 | Config Value               |
|--------------------------------------|----------------------------|
| speaker diarization                  | speaker                    |
| channel diarization                  | channel                    |
| speaker change                       | speaker_change             |
| channel diarization & speaker change | channel_and_speaker_change |

## Speaker diarization

Speaker diarization aggregates all audio channels into a single stream for processing, and picks out different speakers based on acoustic matching.

By default the feature is disabled. To enable speaker diarization the following must be set when you are using the config object:

```
{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "diarization": "speaker"
  }
}
```

When enabled, every `word` and `punctuation` object in the output results will be a given "speaker" property which is a label indicating who said that word. There are two kinds of labels you will see:

- `S#` - S stands for speaker and the # will be an incrementing integer identifying an individual speaker. S1 will appear first in the results, followed by S2 and S3 etc.
- `UU` - Diarization is disabled or individual speakers cannot be identified. `UU` can appear for example if some background noise is transcribed as speech, but the diarization system does not recognise it as a speaker.

**Note:** Enabling diarization increases the amount of time taken to transcribe an audio file. In general we expect diarization to take roughly the same amount of time as transcription does, therefore expect the use of diarization to roughly double the overall processing time.

The example below shows relevant parts of a transcript with 3 speakers. The output shows the configuration information passed in the `config.json` object and relevant segments with the different speakers in the JSON output. Only part of the transcript is shown here to highlight how different speakers are displayed in the output.

```
"format": "2.7",
"metadata": {
  "created_at": "2020-07-01T13:26:48.467Z",
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "diarization": "speaker"
  }
},
"results": [
  {
    "alternatives": [
      {
        "confidence": 0.93,
        "content": "hello",
        "language": "en",
        "speaker": "S1"
      }
    ],
    "end_time": 0.51,
    "start_time": 0.36,
    "type": "word"
  },
  {
    "alternatives": [
      {
        "confidence": 1.0,
        "content": "hi",
        "language": "en",
        "speaker": "S2"
      }
    ],
    "end_time": 12.6,
```

```

    "start_time": 12.27,
    "type": "word"
  },
  {
    "alternatives": [
      {
        "confidence": 1.0,
        "content": "good",
        "language": "en",
        "speaker": "S3"
      }
    ],
    "end_time": 80.63,
    "start_time": 80.48,
    "type": "word"
  }
}

```

In our JSON output, `start_time` identifies when a person starts speaking each utterance and `end_time` identifies when they finish speaking.

### Speaker diarization tuning

The sensitivity of the speaker detection is set to a sensible default that gives the optimum performance under most circumstances. However, you can change this value based on your specific requirements by using the `speaker_sensitivity` setting in the `speaker_diarization_config` section of the job config object, which takes a value between 0 and 1 (the default is 0.5). A higher sensitivity will increase the likelihood of more unique speakers returning. For example, if you see fewer speakers returned than expected, you can try increasing the sensitivity value, or if too many speakers are returned try reducing this value. It's not guaranteed to change since several factors can affect the number of speakers detected. Here's an example of how to set the value:

```

{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "diarization": "speaker",
    "speaker_diarization_config": {
      "speaker_sensitivity": 0.6
    }
  }
}

```

### Speaker diarization post-processing

To enhance the accuracy of our speaker diarization, we make small corrections to the speaker labels based on the punctuation in the transcript. For example if our system originally thought that 9 words in a sentence were spoken by speaker S1, and only 1 word by speaker S2, we will correct the incongruous S2 label to be S1. This only works if punctuation is enabled in the transcript.

Therefore if you disable punctuation, for example by removing all `permitted_marks` in the `punctuation_overrides` section of the `config.json` then expect the accuracy of speaker diarization to vary slightly.

### Speaker diarization timeout

Speaker diarization will timeout if it takes too long to run for a particular audio file. Currently the timeout is set to 5 minutes or  $0.5 * \text{the audio duration}$ ; whichever is longer. For example, with a 2 hour audio file the timeout is 1 hour. If a timeout happens the transcript will still be returned but without the speaker labels set.

If a timeout occurs then all speaker labels in the output will be labelled as UU.

Under normal operation we do not expect diarization to timeout, but diarization can be affected by a number of factors including audio quality and the number of speakers. If you do encounter timeouts frequently then please get in contact with Speechmatics support.

## Channel diarization

The V2 API also supports *Channel diarization* which can be used to add your own speaker or channel labels to the transcript. With Channel diarization, multiple input channels are processed individually and collated into a single transcript. In order to use this method of diarization your input audio must have been transcoded into multiple channels or streams.

In order to use this feature you set the `diarization` property to `channel`. You optionally name these channels by using the `channel_diarization_labels` in the configuration:

```
{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "diarization": "channel",
    "channel_diarization_labels": [
      "Presenter",
      "Questions"
    ]
  }
}
```

If you do not specify any labels then defaults will be used (e.g. Channel 1). The number of labels you use should be the same as the number of channels in your audio. Additional labels are ignored. When the transcript is returned a `channel` property for each word will indicate the speaker, for example:

```
"results": [
  {
    "type": "word",
    "end_time": 1.8,
    "start_time": 1.45,
    "channel": "Presenter",
    "alternatives": [
      {
        "display": {
          "direction": "ltr"
        },
        "language": "en",
        "content": "world",
        "confidence": 0.76
      }
    ]
  }
]
```

## Speaker change detection

This feature introduces markers into the JSON transcript only that indicate when a speaker change has been detected in the audio. For example, if the audio contains two people speaking to each other and you want the transcript to show when there is a change of speaker, specify `speaker_change` as the diarization setting:

```
{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "diarization": "speaker_change"
  }
}
```

The transcript will have special json elements in the `results` array between two words where a different person started talking. For example, if one person says "Hello James" and the other responds with "Hi", there will a `speaker_change` json element between "James" and "Hi".

```
"results": [
  {
    "start_time": 0.1,
    "end_time": 0.22,
    "type": "word",
    "alternatives": [
      {
        "confidence": 0.71,
        "content": "Hello",
        "language": "en",
        "speaker": "UU"
      }
    ]
  },
  {
    "start_time": 0.22,
    "end_time": 0.55,
    "type": "word",
    "alternatives": [
      {
        "confidence": 0.71,
        "content": "James",
        "language": "en",
        "speaker": "UU"
      }
    ]
  },
  {
    "start_time": 0.55,
    "end_time": 0.55,
    "type": "speaker_change",
    "alternatives": []
  },
  {
    "start_time": 0.56,
    "end_time": 0.61,
    "type": "word",
    "alternatives": [
      {
        "confidence": 0.71,
        "content": "Hi",
        "language": "en",
        "speaker": "UU"
      }
    ]
  }
]
```

The sensitivity of the speaker change detection is set to a sensible default that gives the optimum performance under most circumstances. You can however change this if you wish using the `speaker_change_sensitivity` setting, which takes a value between 0 and 1 (the default is 0.4). The higher the sensitivity setting, the more likelihood of a speaker change being indicated. We've found through our own experimentation that values outside the range 0.3-0.6 produce too few speaker change events, or too many false positives. Here's an example of how to set the value:



```

{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "diarization": "speaker_change",
    "speaker_change_sensitivity": 0.55
  }
}

```

## Speaker change detection With Channel diarization

The speaker change feature can be used in conjunction with channel diarization. It will process the channels separately and indicate in the output both the channels and the speaker changes. For example, if a two-channel audio contains two people greeting each other (both recorded over the same channel), the config submitted with the audio can request speaker change detection like this:

```

{
  "type": "transcription",
  "transcription_config": {
    "diarization": "channel_and_speaker_change"
  }
}

```

Again, the `speaker_change_sensitivity` setting may be used to tune the likelihood of speaker change being identified.

## Custom dictionary

The Custom Dictionary feature allows a list of custom words to be added for each transcription job. This helps when a specific word is not recognised during transcription. It could be that it's not in the vocabulary for that language, for example a company or person's name. Adding custom words can improve the likelihood they will be output.

The `sounds_like` feature is an extension to this to allow alternative pronunciations to be specified to aid recognition when the pronunciation is not obvious.

The Custom Dictionary feature can be accessed through the `additional_vocab` property.

Prior to using this feature, consider the following:

- `sounds_like` is an optional setting recommended when the pronunciation is not obvious for the word or it can be pronounced in multiple ways; it is valid just to provide the `content` value
- `sounds_like` only works with the main script for that language
  - Japanese (ja) `sounds_like` only supports full width Hiragana or Katakana
- You can specify up to 1000 words or phrases (per job) in your custom dictionary

```

"transcription_config": {
  "language": "en",
  "additional_vocab": [
    {
      "content": "gnocchi",
      "sounds_like": [
        "nyohki",
        "nokey",
        "nochi"
      ]
    },
    {
      "content": "CEO",
      "sounds_like": [
        "C.E.O."
      ]
    }
  ]
}

```

```

    ]
  },
  {
    "content": "financial crisis"
  }
]
}

```

In the above example, the words *gnocchi* and *CEO* have pronunciations applied to them; the phrase *financial crisis* does not require a pronunciation. The `content` property represents how you want the word to be output in the transcript.

## Output locale

For the English language pack only, it is possible to specify the spelling rules to be used when generating the transcription, based on the `output_locale` configuration setting.

The three locales that are available are:

- British English (en-GB)
- US English (en-US)
- Australian English (en-AU)

If no locale is specified then the ASR engine will use whatever spelling it has learnt as part of our language model training (in other words it will be based on the training data used).

An example configuration request is below:

```

{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "output_locale": "en-GB"
  }
}

```

The following locales are supported for Chinese Mandarin:

- Simplified Mandarin (cmn-Hans)
- Traditional Mandarin (cmn-Hant)

The default is Simplified Mandarin.

## Advanced punctuation

All Speechmatics language packs support Advanced Punctuation. This uses machine learning techniques to add in more naturalistic punctuation, improving the readability of your transcripts.

The following punctuation marks are supported for each language:

| Language(s)         | Supported Punctuation | Comment                          |
|---------------------|-----------------------|----------------------------------|
| Cantonese, Mandarin | , 。 ? ! 、             | Full-width punctuation supported |
| Japanese            | 。 、                   | Full-width punctuation supported |
| Hindi               | ? , !                 |                                  |
| All other languages | . , ! ?               |                                  |

If you do not want to see any of the supported punctuation marks in the output, then you can explicitly control this through the `punctuation_overrides` settings, for example:

```

"transcription_config": {
  "language": "en",
  "punctuation_overrides": {
    "permitted_marks": [ ".", ", " ]
  }
}

```

This will exclude exclamation and question marks from the returned transcript.

All Speechmatics output formats support Advanced Punctuation. JSON output places punctuation marks in the results list marked with a `type` of `"punctuation"`.

**Note:** Disabling punctuation may slightly harm the accuracy of speaker diarization. Please see the ["Speaker diarization post-processing"](#) section in these docs for more information.

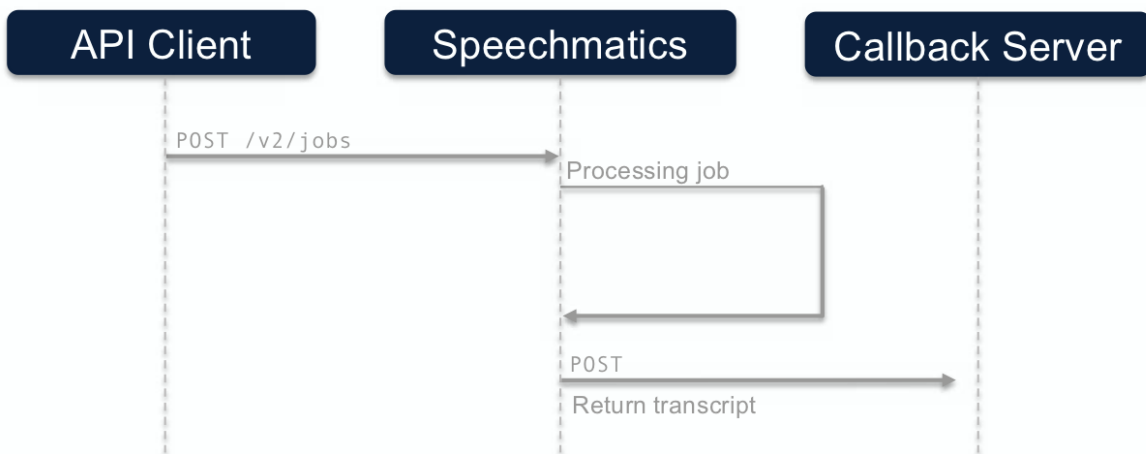
## Notifications

Notifications allow users to be informed both as to the status of a job and to receive the output, without having to continuously poll the Speechmatics SaaS to check for the status of a job.

Speechmatics sends a notification to a web service that you control and specify in the configuration request. This is done once the job is done transcript is available.

### Notification call flow

The call flow for the notification method looks like this:



Below are examples of a configuration that requests a `JSON-v2` transcript and the media file upon job completion:

### Notification configuration example

```

{
  "type": "transcription",
  "transcription_config": {
    "language": "en"
  },
  "notification_config": [
    {
      "url": "https://collector.example.org/callback",
      "contents": [
        "transcript",
        "data"
      ]
    }
  ]
}

```

```

    ],
    "auth_headers": [
      "Authorization: Bearer eyJ0eXAiOiJKV1QiLCJhb"
    ]
  }
]
}

```

This configuration example assumes you have implemented a `/callback` endpoint on host `collector.example.org` that listens for POST requests containing Speechmatics transcripts. In this example requests are only accepted if the auth token `eyJ0eXAiOiJKV1QiLCJhb` is used (note this is the auth token that *your* service accepts, *not* the Speechmatics auth token).

## How to configure a notification

Here is the relevant information you can pass in a notification request:

| Option       | Mandatory | Description   |
|--------------|-----------|---|
| URL          | Yes       | The web address where the notification will be sent. Up to 3xURLs are supported   |
| auth_headers | no        | If required to successfully authenticate to your specified notification location  |
| Content      | No        | <p>There are multiple types of content that can be sent. Multiple attachments can be sent in multiple callbacks:</p> <p><code>transcript</code>: The transcript in <code>json-v2</code> format</p> <p><code>transcript.json-v2</code>: The transcript in <code>json-v2</code> format</p> <p><code>transcript.txt</code>: The Transcript in <code>txt</code> format</p> <p><code>transcript.srt</code>: The transcript in <code>srt</code> format</p> <p><code>data</code>: The media file provided in the job submission request</p> <p><code>jobinfo</code>: The configuration information used to submit the job. Also present in the transcript in <code>json-v2</code> format</p> |
| Method       | No        | POST and PUT HTTP Methods are supported. If none is specified, then POST is used  |

Further Notes:

- If no content is requested as part of the notification, it will simply be a HTTP message containing the job ID and the status of the job
- Up to 3 URLs are supported in any request
- Multiple pieces of content can be sent as multiple attachments in one request, allowing any combination of the input(s) and output(s) of the job to be forwarded to another processing stage.
- You can setup multiple notifications to different endpoints: for instance you can send a `jobinfo` notification to one service, and the transcript notification to another.
- Callbacks with a single attachment will send the content item as the HTTP request body, rather than using multipart mode. This allows writing an individual item to an object store like Amazon S3.
- `auth_headers` should be specified:
  - To satisfy authentication / authorization requirements for systems that do not support auth tokens in query parameters.
  - To control behaviour of an object store or another existing service endpoint.
- Multiple callbacks can be specified per job.
  - This allows sending individual pieces of content to different URLs, eg. to allow uploading the audio and transcript to an object store as distinct objects for a downstream workflow.
  - It allows sending arbitrary combinations of the inputs/outputs to multiple destinations, to support a fanout workflow.

Callbacks will be invoked in parallel and so may complete in any order. If a downstream workflow depends on getting several items of content delivered as separate callbacks (eg. uploaded as separate items to S3), then the downstream processing logic will need to be robust to the ordering of uploads and the possibility that only some might succeed.

To ensure that the callbacks you receive come from Speechmatics you can [apply a whitelist](#).

## Accepting the notification

You need to ensure that the service that you implement to receive the callback notification is capable of processing the Speechmatics transcript using the format that has been specified in the config JSON. When testing your integration you should check the error logs on your web service to ensure that notifications are being accepted and processed correctly.

The callback appends the job ID as a query string parameter with name `id`, as well as the status of the job. As an example, if the job ID is `r6sr3j1zjj`, you'd see the following POST request:

```
POST /callback?id=r6sr3j1zjj&status=success HTTP/1.1
Host: collector.example.org
```

The user agent is `Speechmatics-API/2.0`.

## Notification webserver configuration

Once the submitted media file is transcribed, and the transcript file is available, the Speechmatics cloud service will send the transcript file in a HTTP POST request to the client web server (customers webserver) specified in the `notification_config` config object. If the Speechmatics cloud service does not receive a 2xx response (that the request is successfully received, understood, or accepted) it will keep trying to send the file until it reaches the set timeout threshold.

If the clients webserver that has been set as the callback endpoint is not configured with a large enough size limit to receive the transcript file and original media file it will generate a 413 (Request Entity Too Large) response to the Speechmatics service. The Speechmatics cloud service app has not receive a 2xx response it will continue to retry sending the file.

Users are recommended to check their webserver size limits to ensure they are adequate for the files that will be sent.

## Notification failure

If the Speechmatics cloud service is unable to send a notification to the specified online location, the transcript is still available to directly retrieve via an API request. A user can retrieve failure information by making a GET `/jobs/$JOBID` request.

If the job has failed, there will be an additional `errors` element, which will show all failure messages Speechmatics cloud service encountered when attempting to send notifications. Please note, there can be multiple failure attempts associated with one submitted job.

```
{
  "job": {
    "config": {
      "fetch_data": {
        "url": "https://example.com/average-files/punctuation1.mp3"
      },
      "notification_config": [
        {
          "contents": [
            "jobjnfo"
          ],
          "url": "https://example.com"
        }
      ],
      "transcription_config": {
        "language": "de"
      },
      "type": "transcription"
    }
  }
}
```

```

},
"created_at": "2021-07-19T09:02:17.283Z",
"data_name": "",
"duration": 4,
"errors": [
  {
    "message": "Error in sending notification: unable to send notification: HTTPError: Post
\"https://example.com/500?id=1uyo82blbv&status=success\": context deadline exceeded (Client.Timeout
exceeded while awaiting headers), retrying",
    "timestamp": "2021-07-19T09:04:11.080Z"
  },
  {
    "message": "Error in sending notification: unable to send notification: Response status: 500,
retrying",
    "timestamp": "2021-07-19T09:04:26.619Z"
  },
  {
    "message": "Error in sending notification: unable to send notification: Response status: 500,
retrying",
    "timestamp": "2021-07-19T09:04:47.090Z"
  },
  {
    "message": "unable to send notification: unable to send notification: Response status: 500",
    "timestamp": "2021-07-19T09:05:17.347Z"
  },
  {
    "message": "Sending notification failed",
    "timestamp": "2021-07-19T09:05:17.570Z"
  }
],
"id": "1uyo82blbv",
"status": "done"
}
}

```

## Metadata and job tracking

It is now possible to attach richer metadata to a job using the `tracking` configuration. The tracking object contains the following properties:

| Name             | Type             | Description                      | Notes      |
|------------------|------------------|----------------------------------|------------|
| <b>title</b>     | <b>str</b>       | The title of the job.            | [optional] |
| <b>reference</b> | <b>str</b>       | External system reference.       | [optional] |
| <b>tags</b>      | <b>list[str]</b> |                                  | [optional] |
| <b>details</b>   | <b>object</b>    | Customer-defined JSON structure. | [optional] |

This allows you to track the job through your own management workflow using whatever information is relevant to you.

### Metadata and job tracking configuration example

```

{
  "type": "transcription",
  "transcription_config": {
    "language": "en"
  },
  "tracking": {

```

```

"title": "ACME Q12018 Statement",
"reference": "/data/clients/ACME/statements/segs/2018Q1-seg8",
"tags": [
  "quick-review",
  "segment"
],
"details": {
  "client": "ACME Corp",
  "segment": 8,
  "seg_start": 963.201,
  "seg_end": 1091.481
}
}
}

```

## Word Tagging

### Profanity Tagging

Speechmatics now outputs in JSON transcript only a metadata tag to indicate whether a word is a profanity or not. This is for the following languages:

- English (EN)
- Italian (IT)
- Spanish (ES)

For each language pack, the list of profanities is not alterable. Users do not have to take any action to access this - it is provided in our JSON output as standard. Customers can use this tag for their own post-processing in order to identify, redact, or obfuscate profanities and integrate this data into their own workflows. An example of how this looks is below.

```

"results": [
  {
    "alternatives": [
      {
        "confidence": 1.0,
        "content": "$PROFANITY",
        "language": "en",
        "speaker": "UU",
        "tags": [
          "profanity"
        ]
      }
    ]
  },
  "end_time": 18.03,
  "start_time": 17.61,
  "type": "word"
}
]

```

### Disfluency Tagging

Speechmatics now outputs in JSON transcript only a metadata tag to indicate whether a word is a disfluency or not in the English language only. A disfluency here refers to a set list of words in English that imply hesitation or indecision. Please note while disfluency can cover a range of items like stuttering and interjections, here it is only used to tag words such as 'hmm' or 'umm'. Users do not have to take any action to access this - it is provided in our JSON output as standard. Customers can use this tag for their own post-processing workflows. An example of how this looks is below:

```

"results": [
  {
    "alternatives": [

```

```

    {
      "confidence": 1.0,
      "content": "hmm",
      "language": "en",
      "speaker": "UU",
      "tags": [
        "disfluency"
      ]
    }
  ],
  "end_time": 18.03,
  "start_time": 17.61,
  "type": "word"
}
]

```

## Domain Language Packs

Some Speechmatics language packs are optimized for specific domains where high accuracy for specific vocabulary and terminology is required. Using the domain parameter provides additional transcription accuracy, and must be used in conjunction with a standard language pack (this is currently limited to the "finance" domain and supports the "en" language pack). An example of how this looks is below:

```

{
  "type": "transcription",
  "transcription_config": {
    "language": "en",
    "domain": "finance"
  }
}

```

These domain language packs are built on top of our global language packs so give the highest accuracy in different acoustic environments that our customers have come to expect.

It is expected that whilst there will be improvements for the specific domain there can be degradation in accuracy for other outside domains.

## Usage

### Overview

This section describes how to make requests to Speechmatics SaaS to understand your usage of the service. Usage is reported as the number of jobs successfully processed in a time period, along with the total duration of the corresponding audio files.

As for other requests to Speechmatics SaaS, usage is queried using an Authorization Token. Usage is returned for the account associated with the Authorization Token. When using the examples below, you should replace the Authorization Token shown with your own.

All usage is reported on a Coordinated Universal Time (UTC) calendar day basis. Note that usage for the current day (in UTC) is **not** included in the results.

Note that deleting a completed job will have no effect on reported usage.

### Requesting Usage

The following request retrieves usage for all jobs from the beginning of 15 September 2020 until the end of 31 December 2020. More precisely, this includes all jobs completed successfully at or after 2020-09-15T00:00Z and before 2021-01-01T00:00Z.



The `since` and `until` parameters are both optional:

- If `since` is omitted, then usage is reported from the day on which the first job was successfully processed for the account.
- If `until` is omitted, then usage is reported until the end of the previous UTC day.

### Linux example

```
curl -L -X GET "https://asr.api.speechmatics.com/v2/usage?since=2020-09-15&until=2020-12-31" -H
"Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

### Windows example

```
curl.exe -L -X GET "https://asr.api.speechmatics.com/v2/usage?since=2020-09-15&until=2020-12-31" -H
"Authorization: Bearer NDFjOTE3NGEtOWVm" | jq
```

### The response

The response is a JSON object containing usage information. See [API Reference](#) for full details.

The response JSON contains `since` and `until` properties which should be interpreted in the same way as the `since` and `until` query parameters. In general, the `since` value is returned as the first day in the specified time period on which a job was successfully processed. If the `since` query parameter is not provided and there is no usage during the specified time period, then the `since` value is returned as `1970-01-01`.

The response also contains `summary` and `details` properties. The `summary` value lists the total usage for each SaaS mode and job type. Currently, `mode` will always be `batch` and `type` will be one of `transcription` or `alignment`. The `details` value lists a more granular breakdown of usage by mode, job type, language and operating point. Each object in `summary` and `details` contains `count` and `duration_hrs` fields which report the total number of successfully completed jobs and the total audio duration (in hours) of these jobs. If there is no usage in the specified time period, then the `summary` and `details` values will be `null`.

Here is an example response:

```
HTTP/1.1 200 OK
Content-Type: application/json

{
  'since': '2020-09-15',
  'until': '2020-12-31',
  'summary': [
    {
      'mode': 'batch',
      'type': 'transcription',
      'count': 55386,
      'duration_hrs': 8432.16
    },
    {
      'mode': 'batch',
      'type': 'alignment',
      'count': 489,
      'duration_hrs': 70.53
    }
  ],
  'details': [
    {
      'mode': 'batch',
      'type': 'transcription',
      'language': 'sv',
```

```

    'operating_point': 'standard',
    'count': 27830,
    'duration_hrs': 5204.55
  },
  {
    'mode': 'batch',
    'type': 'transcription',
    'language': 'de',
    'operating_point': 'enhanced',
    'count': 27556,
    'duration_hrs': 3227.61
  },
  {
    'mode': 'batch',
    'type': 'alignment',
    'language': 'en',
    'count': 489,
    'duration_hrs': 70.53
  }
]
}

```

## Troubleshooting

### Basics

If you see errors when you submit a job or performing any other actions it might be worth checking few simple elements to potentially help avoid having to raise a support ticket.

Its worth checking few specific things:

1. The endpoint. Ensure that you are using the correct endpoint. This is especially important for trial users. Endpoints can be found in the overview section of this documentation portal in the API endpoints subsection.
2. The language code. Ensure that the language code is correct. These can be checked against the language list in the language support subsection of the overview sections.
3. Token expiration. This is especially relevant for trial users. Trial tokens are valid for 14 days, to check the status of your trial ticket please reach out to your account point of contact at Speechmatics.

### Raising an incident

If you need to raise a support ticket, then please open a ticket by emailing [support@speechmatics.com](mailto:support@speechmatics.com).

As part of good practice always provide:

- A thorough description of the error you encountered
- Any relevant Job IDs
- The configuration information you used to submit the job request

Please provide a Job ID where possible. A response for a successful job should look like this:

```

HTTP/1.1 201 Created
Content-Length: 20
Content-Type: application/json
Request-Id: df6ec500191bf2c02ee42d519fba34f
Strict-Transport-Security: max-age=15724800; includeSubDomains
X-Azure-Ref:
01mDmXAAAAABUtz62gS7QRI2F3f5B+19wTE9OMjFFREdFMDMxMgBhN2JjOWQ4MCO2YjBiLTQ1NWETyJE3MS01NGJkZmNiYWEOYTk=

Date: Thu, 23 May 2019 08:59:08 GMT

```

```
{"id": "z2jfp1jwu1"}
```

Here, the Job ID is `z2jfp1jwu1` . The Request ID is `df6ec500191bf2c02ee42d519fba34f`

If you submitted a job, but did not receive a Job ID as expected please provide any Request-Id, and X-Azure-Ref headers that may have been returned to you in the HTTP response. An error will be reported as a 4XX or 5XX status code; instead of the job ID an error message will be returned. The Request-Id can be included in the ticket to support to help support engineers troubleshoot the problem. Here's an example:

```
HTTP/1.1 401 Unauthorized
Content-Length: 179
Content-Type: text/html
Request-Id: 00adae505fa8cb7c7f24c6bf52c60048
Strict-Transport-Security: max-age=15724800; includeSubDomains
X-Azure-Ref:
OLWLMXAAAAAB2mDIIdAu7kSZ/kKszi7eH4TE9OMjFFREdFMDMwOQBhN2JjOWQ4MC02YjBiLTQ1NWETyJE3MS01NGJkZmNiYWEOYTk=

Date: Thu, 23 May 2019 09:04:49 GMT

<html>
<head><title>401 Authorization Required</title></head>
<body>
<center><h1>401 Authorization Required</h1></center>
<hr><center>nginx/1.15.8</center>
</body>
</html>
```

Please note that in the future the response for HTTP 401 and 429 responses will be content-type `application/json` rather than `text/html` . This is how a HTTP 401 request will now look:

```
HTTP/1.1 401 Unauthorized
Content-Length: 179
Content-Type: application/json
Request-Id: 00adae505fa8cb7c7f24c6bf52c60048
Strict-Transport-Security: max-age=15724800; includeSubDomains
X-Azure-Ref:
OLWLMXAAAAAB2mDIIdAu7kSZ/kKszi7eH4TE9OMjFFREdFMDMwOQBhN2JjOWQ4MC02YjBiLTQ1NWETyJE3MS01NGJkZmNiYWEOYTk=

Date: Thu, 23 May 2019 09:04:49 GMT

{"code": 401, "error": "Unauthorized"}
```

Here, the Request-Id is `00adae505fa8cb7c7f24c6bf52c60048` . Please note the response here is in JSON format and is no longer a HTTP response.

## Rate limited requests

A 429 error means that the request has been rate limited by our API. This situation could be avoided by adding 1 second delay between requests of the same type, e.g., adding 1 second delay between `GET /v2/jobs/{job_id}` requests.

## Speechmatics ASR REST API

The Speechmatics' Automatic Speech Recognition (ASR) REST API is used to submit ASR jobs, receive job status and results, and retrieve usage.

- [Jobs API](#)
- [Usage API](#)
- [Object Models](#)

## Contact information:

In case of system issues, requests, or unavailability, please contact [support@speechmatics.com](mailto:support@speechmatics.com)

## Jobs API

Version: 2.7.0

/jobs

POST

### Summary:

Create a new job.

### Parameters

| Name          | Located in | Description  | Required | Schema |
|---------------|------------|--|----------|--------|
| Authorization | header     | Customer API token   | Yes      | string |
| config        | formData   | JSON containing a <code>JobConfig</code> model indicating the type and parameters for the recognition job.                 | Yes      | string |
| data_file     | formData   | The data file to be processed. Alternatively the data file can be fetched from a url specified in <code>JobConfig</code> . | No       | file   |

### Responses

| Code | Description           | Schema                            |
|------|-----------------------|-----------------------------------|
| 201  | OK                    | <a href="#">CreateJobResponse</a> |
| 400  | Bad request           | <a href="#">ErrorResponse</a>     |
| 401  | Unauthorized          | <a href="#">ErrorResponse</a>     |
| 403  | Forbidden             | <a href="#">ErrorResponse</a>     |
| 429  | Rate Limited          |                                   |
| 500  | Internal Server Error | <a href="#">ErrorResponse</a>     |

GET

### Summary:

List all jobs.

### Parameters

| Name          | Located in | Description        | Required | Schema |
|---------------|------------|--------------------|----------|--------|
| Authorization | header     | Customer API token | Yes      | string |

### Responses

| Code | Description           | Schema                               |
|------|-----------------------|--------------------------------------|
| 200  | OK                    | <a href="#">RetrieveJobsResponse</a> |
| 401  | Unauthorized          | <a href="#">ErrorResponse</a>        |
| 429  | Rate Limited          |                                      |
| 500  | Internal Server Error | <a href="#">ErrorResponse</a>        |

## /jobs/{jobid}

### GET

#### Summary:

Get job details for a specific job, including progress and any error reports.

#### Parameters

| Name          | Located in | Description        | Required | Schema |
|---------------|------------|--------------------|----------|--------|
| Authorization | header     | Customer API token | Yes      | string |
| jobid         | path       | ID of the job.     | Yes      | string |

#### Responses

| Code | Description           | Schema                              |
|------|-----------------------|-------------------------------------|
| 200  | OK                    | <a href="#">RetrieveJobResponse</a> |
| 401  | Unauthorized          | <a href="#">ErrorResponse</a>       |
| 404  | Not found             | <a href="#">ErrorResponse</a>       |
| 410  | Job Expired           | <a href="#">ErrorResponse</a>       |
| 429  | Rate Limited          |                                     |
| 500  | Internal Server Error | <a href="#">ErrorResponse</a>       |

### DELETE

#### Summary:

Delete a job and remove all associated resources.

#### Parameters

| Name          | Located in | Description  | Required | Schema |
|---------------|------------|--|----------|--------|
| Authorization | header     | Customer API token   | Yes      | string |
| jobid         | path       | ID of the job to delete.   | Yes      | string |
| force         | query      | Use <code>true</code> to force delete a job that may still be running. Default is <code>false</code> . | No       | string |

#### Responses

| Code | Description               | Schema                            |
|------|---------------------------|-----------------------------------|
| 200  | The job that was deleted. | <a href="#">DeleteJobResponse</a> |
| 401  | Unauthorized              | <a href="#">ErrorResponse</a>     |
| 404  | Not found                 | <a href="#">ErrorResponse</a>     |
| 410  | Job Expired               | <a href="#">ErrorResponse</a>     |
| 423  | Resource Locked           | <a href="#">ErrorResponse</a>     |
| 429  | Rate Limited              |                                   |

|     |                       |                               |
|-----|-----------------------|-------------------------------|
| 500 | Internal Server Error | <a href="#">ErrorResponse</a> |
|-----|-----------------------|-------------------------------|

## /jobs/{jobid}/data

GET

### Summary:

Get the data file used as input to a job.

### Parameters

| Name          | Located in | Description        | Required | Schema |
|---------------|------------|--------------------|----------|--------|
| Authorization | header     | Customer API token | Yes      | string |
| jobid         | path       | ID of the job.     | Yes      | string |

### Responses

| Code | Description           | Schema                        |
|------|-----------------------|-------------------------------|
| 200  | OK                    | file                          |
| 401  | Unauthorized          | <a href="#">ErrorResponse</a> |
| 404  | Not found             | <a href="#">ErrorResponse</a> |
| 410  | Gone                  | <a href="#">ErrorResponse</a> |
| 429  | Rate Limited          |                               |
| 500  | Internal Server Error | <a href="#">ErrorResponse</a> |

## /jobs/{jobid}/transcript

GET

### Summary:

Get the transcript for a transcription job.

### Parameters

| Name          | Located in | Description  | Required | Schema |
|---------------|------------|--|----------|--------|
| Authorization | header     | Customer API token   | Yes      | string |
| jobid         | path       | ID of the job.   | Yes      | string |
| format        | query      | The transcription format (by default the <code>json-v2</code> format is returned). | No       | string |

### Responses

| Code | Description  | Schema                                     |
|------|--------------|--|
| 200  | OK           | <a href="#">RetrieveTranscriptResponse</a> |
| 401  | Unauthorized | <a href="#">ErrorResponse</a>              |
| 404  | Not found    | <a href="#">ErrorResponse</a>              |
| 410  | Job Expired  | <a href="#">ErrorResponse</a>              |
|      |              |  |

|     |                       |                               |
|-----|-----------------------|-------------------------------|
| 429 | Rate Limited          |                               |
| 500 | Internal Server Error | <a href="#">ErrorResponse</a> |

## Usage API

**/usage**

**GET**

### Summary:

Get usage information for an account.

### Parameters

| Name          | Located in | Description   | Required | Schema |
|---------------|------------|---|----------|--------|
| Authorization | header     | Customer API token                                  | Yes      | string |
| since         | query      | Start date for usage information in ISO 8601 format | No       | string |
| until         | query      | End date for usage information in ISO 8601 format   | No       | string |

### Responses

| Code | Description           | Schema                        |
|------|-----------------------|-------------------------------|
| 200  | OK                    | <a href="#">UsageResponse</a> |
| 401  | Unauthorized          | <a href="#">ErrorResponse</a> |
| 429  | Rate Limited          |                               |
| 500  | Internal Server Error | <a href="#">ErrorResponse</a> |

## Object Models

### ErrorResponse

| Name   | Type    | Description               | Required |
|--------|---------|---------------------------|----------|
| code   | integer | The HTTP status code.     | Yes      |
| error  | string  | The error message.        | Yes      |
| detail | string  | The details of the error. | No       |

### TrackingData

| Name      | Type       | Description                      | Required |
|-----------|------------|----------------------------------|----------|
| title     | string     | The title of the job.            | No       |
| reference | string     | External system reference.       | No       |
| tags      | [ string ] |                                  | No       |
| details   | object     | Customer-defined JSON structure. | No       |

### DataFetchConfig

| Name | Type   | Description | Required |
|------|--------|-------------|----------|
| url  | string |             | Yes      |

|              |                  |   |    |
|--------------|------------------|---|----|
| auth_headers | [<br>string<br>] | A list of additional headers to be added to the input fetch request when using http or https. This is intended to support authentication or authorization, for example by supplying an OAuth2 bearer token. | No |
|--------------|------------------|---|----|

### TranscriptionConfig

| Name                       | Type                                     | Description   | Required |
|----------------------------|--|---|----------|
| language                   | string                                   | Language pack to process the audio input, normally specified as an ISO language code  | Yes      |
| domain                     | string                                   | Request a specialized language pack optimized for a particular domain, e.g. "finance". Domain is only supported for selected languages.   | No       |
| output_locale              | string                                   | Language locale to be used when generating the transcription output, normally specified as an ISO language code   | No       |
| additional_vocab           | [ object ]                               | List of custom words or phrases that should be recognized. Alternative pronunciations can be specified to aid recognition.  | No       |
| punctuation_overrides      | <a href="#">PunctuationConfig</a>        | Configuration for punctuation settings. <code>permitted_marks</code> defines the punctuation marks which the client is prepared to accept in transcription output, or the special value 'all' (the default). Unsupported marks are ignored. This value is used to guide the transcription process. <code>sensitivity</code> ranges between zero and one. Higher values will produce more punctuation. The default is 0.5.   | No       |
| diarization                | string                                   | Specify whether speaker or channel labels are added to the transcript. The default is <code>none</code> . - <b>none</b> : no speaker or channel labels are added. - <b>speaker</b> : speaker attribution is performed based on acoustic matching; all input channels are mixed into a single stream for processing. - <b>channel</b> : multiple input channels are processed individually and collated into a single transcript. - <b>speaker_change</b> : the output indicates when the speaker in the audio changes. No speaker attribution is performed. This is a faster method than speaker. The reported speaker changes may not agree with speaker. - <b>channel_and_speaker_change</b> : both channel and speaker_change are switched on. The speaker change is indicated if more than one speaker are recorded in one channel. | No       |
| speaker_diarization_config | <a href="#">SpeakerDiarizationConfig</a> | Configuration for speaker diarization. Includes <code>speaker_sensitivity</code> : Range between 0 and 1. A higher sensitivity will increase the likelihood of more unique speakers returning. For example, if you see fewer speakers returned than expected, you   | No       |



|                            |            |   |    |
|----------------------------|------------|---|----|
|                            |            | can try increasing the sensitivity value or if too many speakers are returned try reducing this value. The default is 0.5.  |    |
| speaker_change_sensitivity | float      | Used for the <code>speaker_change</code> feature. Range between 0 and 1. Controls how responsive the system is for potential speaker changes. High value indicates high sensitivity. Defaults to 0.4. | No |
| channel_diarization_labels | [ string ] | Transcript labels to use when using collating separate input channels.  | No |
| operating_point            | string     | Specify whether to use a <code>standard</code> or <code>enhanced</code> model for transcription. By default the model used is <code>standard</code>   | No |
| enable_entities            | Boolean    | Specify whether to enable <code>entity</code> types within JSON output, as well as additional <code>spoken_form</code> and <code>written_form</code> metadata. By default <code>false</code>          | No |

### PunctuationConfig

Additional configuration for the Advanced Punctuation feature.

| Name            | Type   | Description   | Required |
|-----------------|--------|---|----------|
| permitted_marks | string | Defines the punctuation marks which the client is prepared to accept in transcription output, or the special value 'all' (the default). Unsupported marks are ignored. This value is used to guide the transcription process. | No       |
| sensitivity     | float  | Ranges between zero and one. Higher values will produce more punctuation. The default is 0.5.   | No       |

### SpeakerDiarizationConfig

Additional configuration for the Speaker Diarization feature.

| Name                | Type  | Description   | Required |
|---------------------|-------|---|----------|
| speaker_sensitivity | float | Used for <code>speaker_diarization</code> feature. Range between 0 and 1. A higher sensitivity will increase the likelihood of more unique speakers returning. For example, if you see fewer speakers returned than expected, you can try increasing the sensitivity value, or if too many speakers are returned try reducing this value. The default is 0.5. | No       |

### NotificationConfig

| Name | Type   | Description   | Required |
|------|--------|---|----------|
| url  | string | The url to which a notification message will be sent upon completion of the job. The <code>job_id</code> and <code>status</code> are added as query parameters, and any combination of the job inputs and outputs can be included by listing them in <code>contents</code> . If <code>contents</code> is empty, the body of the request will be empty. If only one item is listed, it will be sent as the body of the request with <code>Content-Type</code> set to an appropriate value such as <code>application/octet-stream</code> or <code>application/json</code> . If multiple items are listed they will be sent as named file attachments using the multipart content type. If <code>contents</code> is not specified, the <code>transcript</code> item will be sent as a file attachment named <code>data_file</code> , for backwards compatibility. If the job was rejected or failed during processing, | Yes      |

|              |                  |  |    |
|--------------|------------------|--|----|
|              |                  | that will be indicated by the status, and any output items that are not available as a result will be omitted. The body formatting rules will still be followed as if all items were available. The user-agent header is set to <code>Speechmatics-API/2.0</code> , or <code>Speechmatics API V2</code> in older API versions. |    |
| contents     | [<br>string<br>] | Specifies a list of items to be attached to the notification message. When multiple items are requested, they are included as named file attachments.  | No |
| method       | string           | The method to be used with http and https urls. The default is post.   | No |
| auth_headers | [<br>string<br>] | A list of additional headers to be added to the notification request when using http or https. This is intended to support authentication or authorization, for example by supplying an OAuth2 bearer token.   | No |

### OutputConfig

| Name          | Type   | Description   | Required |
|---------------|--------|---|----------|
| srt_overrides | object | Parameters that override default values of srt conversion. <code>max_line_length</code> : sets maximum count of characters per subtitle line including white space. <code>max_lines</code> : sets maximum count of lines in a subtitle section. | No       |

### JobConfig

JSON object that contains various groups of job configuration parameters. Based on the value of `type`, a type-specific object such as `transcription_config` is required to be present to specify all configuration settings or parameters needed to process the job inputs as expected.

If the results of the job are to be forwarded on completion, `notification_config` can be provided with a list of callbacks to be made; no assumptions should be made about the order in which they will occur.

Customer specific job details or metadata can be supplied in `tracking`, and this information will be available where possible in the job results and in callbacks.

| Name                 | Type                                | Description | Required |
|----------------------|-------------------------------------|-------------|----------|
| type                 | string                              |             | Yes      |
| fetch_data           | <a href="#">DataFetchConfig</a>     |             | No       |
| transcription_config | <a href="#">TranscriptionConfig</a> |             | No       |
| notification_config  | <a href="#">NotificationConfig</a>  |             | No       |
| tracking             | <a href="#">TrackingData</a>        |             | No       |
| output_config        | <a href="#">OutputConfig</a>        |             | No       |

### CreateJobResponse

| Name | Type   | Description   | Required |
|------|--------|---|----------|
| id   | string | The unique ID assigned to the job. Keep a record of this for later retrieval of your completed job. | Yes      |

### JobDetails

Document describing a job. JobConfig will be present in JobDetails returned for GET jobs/ request in the SaaS and in Batch Appliance, but it will not be present in JobDetails returned as item in RetrieveJobsResponse for the Batch Appliance.

| Name       | Type     | Description                            | Required |
|------------|----------|--|----------|
| created_at | dateTime | The UTC date time the job was created. | Yes      |

|           |                           |   |     |
|-----------|---------------------------|---|-----|
| data_name | string                    | Name of the data file submitted for job.  | Yes |
| duration  | integer                   | The file duration (in seconds). May be missing for fetch URL jobs.  | No  |
| errors    | array                     | errors encountered only when either fetching audio or sending notifications to a customer-specified endpoint.   | No  |
| id        | string                    | The unique id assigned to the job.  | Yes |
| status    | string                    | The status of the job. * <i>running</i> - The job is actively running. * <i>done</i> - The job completed successfully. * <i>rejected</i> - The job was accepted at first, but later could not be processed by the transcriber. * <i>deleted</i> - The user deleted the job. * <i>expired</i> - The system deleted the job. Usually because the job was in the <i>done</i> state for a very long time. | Yes |
| config    | <a href="#">JobConfig</a> |   | No  |

### RetrieveJobsResponse

| Name | Type                           | Description | Required |
|------|--------------------------------|-------------|----------|
| jobs | [ <a href="#">JobDetails</a> ] |             | Yes      |

### RetrieveJobResponse

| Name | Type                       | Description | Required |
|------|----------------------------|-------------|----------|
| job  | <a href="#">JobDetails</a> |             | Yes      |

### DeleteJobResponse

| Name | Type                       | Description | Required |
|------|----------------------------|-------------|----------|
| job  | <a href="#">JobDetails</a> |             | Yes      |

### JobInfo

Summary information about an ASR job, to support identification and tracking.

| Name       | Type                         | Description                                | Required |
|------------|------------------------------|--|----------|
| created_at | dateTime                     | The UTC date time the job was created.     | Yes      |
| data_name  | string                       | Name of data file submitted for job.       | Yes      |
| duration   | integer                      | The data file audio duration (in seconds). | Yes      |
| id         | string                       | The unique id assigned to the job.         | Yes      |
| tracking   | <a href="#">TrackingData</a> |  | No       |

### RecognitionMetadata

Summary information about the output from an ASR job, comprising the job type and configuration parameters used when generating the output.

| Name                 | Type                                | Description   | Required |
|----------------------|-------------------------------------|---|----------|
| created_at           | dateTime                            | The UTC date time the transcription output was created. | Yes      |
| type                 | string                              |   | Yes      |
| transcription_config | <a href="#">TranscriptionConfig</a> |   | No       |

|               |                              |    |
|---------------|------------------------------|----|
| output_config | <a href="#">OutputConfig</a> | No |
|---------------|------------------------------|----|

### RecognitionDisplay

| Name      | Type   | Description | Required |
|-----------|--------|-------------|----------|
| direction | string |             | Yes      |

### RecognitionAlternative

List of possible job output item values, ordered by likelihood.

| Name       | Type                               | Description | Required |
|------------|------------------------------------|-------------|----------|
| content    | string                             |             | Yes      |
| confidence | float                              |             | Yes      |
| language   | string                             |             | Yes      |
| display    | <a href="#">RecognitionDisplay</a> |             | No       |
| speaker    | string                             |             | No       |
| tags       | [ string ]                         |             | No       |

### RecognitionResult

An ASR job output item. The primary item types are `word` and `punctuation`. Other item types may be present, for example to provide semantic information of different forms.

| Name         | Type                                       | Description   | Required |
|--------------|--|---|----------|
| channel      | string                                     |   | No       |
| start_time   | float                                      |   | Yes      |
| end_time     | float                                      |   | Yes      |
| entity_class | string                                     | If an entity has been recognised, what type of entity it is. Displayed even if <code>enable_entities</code> is false  | Yes      |
| spoken_form  | array                                      | For <code>entity</code> results only, the <code>spoken_form</code> is the transcript of the words directly spoken. Only valid if <code>enable_entities</code> is true   | No       |
| written_form | array                                      | For <code>entity</code> results only, the <code>written_form</code> is a standardized form of the spoken words. Only valid if <code>enable_entities</code> is true  | No       |
| is_eos       | boolean                                    | Whether the punctuation mark is an end of sentence character. Only applies to punctuation marks.  | No       |
| type         | string                                     | New types of items may appear without being requested; unrecognized item types can be ignored. Current types are <code>word</code> , <code>punctuation</code> , <code>speaker_change</code> , and <code>entity</code> | Yes      |
| alternatives | [ <a href="#">RecognitionAlternative</a> ] |   | No       |

### RetrieveTranscriptResponse

| Name   | Type   | Description   | Required |
|--------|--------|---|----------|
| format | string | Speechmatics JSON transcript format version number. | Yes      |

|          |                                       |  |     |
|----------|---------------------------------------|--|-----|
| job      | <a href="#">JobInfo</a>               |  | Yes |
| metadata | <a href="#">RecognitionMetadata</a>   |  | Yes |
| results  | [ <a href="#">RecognitionResult</a> ] |  | Yes |

### UsageResponse

| Name    | Type                                   | Description  | Required |
|---------|--|--|----------|
| since   | string                                 | Start date for usage in ISO 8601 date format.  | Yes      |
| until   | string                                 | End date for usage in ISO 8601 date format.  | Yes      |
| summary | [ <a href="#">UsageSummaryResult</a> ] | Total usage over all languages and operating points for each combination of SaaS mode and job type. Can be <code>null</code> . | Yes      |
| details | [ <a href="#">UsageDetailsResult</a> ] | Usage for each combination of SaaS mode, job type, language and operating point. Can be <code>null</code> .                    | Yes      |

### UsageSummaryResult

| Name         | Type    | Description   | Required |
|--------------|---------|---|----------|
| mode         | string  | SaaS mode: always 'batch'   | Yes      |
| type         | string  | Job type: one of <code>transcription</code> or <code>alignment</code>                         | Yes      |
| count        | integer | Total number of jobs with this <code>type</code> and <code>mode</code>                        | Yes      |
| duration_hrs | float   | Total audio file duration in hours for jobs with this <code>type</code> and <code>mode</code> | Yes      |

### UsageDetailsResult

| Name            | Type    | Description  | Required |
|-----------------|---------|--|----------|
| mode            | string  | SaaS mode: always 'batch'  | Yes      |
| type            | string  | Job type: one of <code>transcription</code> or <code>alignment</code>  | Yes      |
| language        | string  | Job language code  | Yes      |
| operating_point | string  | Operating point for transcription. One of <code>standard</code> or <code>enhanced</code> .   | Yes      |
| count           | integer | Total number of jobs with this <code>type</code> , <code>mode</code> , <code>language</code> and <code>operating_point</code>                        | Yes      |
| duration_hrs    | float   | Total audio file duration in hours for jobs with this <code>type</code> , <code>mode</code> , <code>language</code> and <code>operating_point</code> | Yes      |