

Asynchronous Transcription



In addition to real-time transcription, Recall.ai also supports transcribing asynchronously after the call has ended.

Quicklinks

- [Create Async Transcript](#)
- [Webhooks](#)

Quickstart

Receive the `recording.done` webhook

You'll be notified when a given recording is ready for transcription by receiving a `recording.done` Status Change event:

JSON

```
{
  "event": "recording.done",
  "data": {
    "data": {
      "code": string,
      "sub_code": string | null,
      "updated_at": string
    },
    "recording": {
      "id": string,
      "metadata": object
    },
    "bot": {
      "id": string,
      "metadata": object
    } | null
  }
}
```

Ask AI



```
}
```

Upon receiving this, you can kick off an async transcript job, assuming your recording has generated an artifact suitable for transcription (e.g. a video or audio artifact).

Start an async transcription job

To kick off an asynchronous transcription job, call the [Create Async Transcript](#) endpoint.

At minimum, you must specify a `provider` configuration that should be used to transcribe the recording.

Example:

```
curl --request POST \
      --url https://us-east-1.recall.ai/api/v1/recording/{RECORDING_ID}/
create_transcript/ \
      --header "Authorization: $RECALLAI_API_KEY" \
      --header "accept: application/json" \
      --header "content-type: application/json" \
      --data '
{
  "provider": {
    "assembly_ai_async": {
      "language_code": "en"
    }
  }
}
```

In this example, we choose AssemblyAI as the provider, and configure the language as English. For a full list of providers and their options, please see the [Create Async Transcript](#) API reference.



Only 10 Transcripts Allowed Per Recording

As each transcript created using the above triggers a transcription on the underlying provider incurring usage costs, we've limited maximum number of transcripts per recording to 10. This helps avoiding cases where bad loop on the consumer end can lead to large number of transcripts being created for the same recording.

Incase you run into this limit for a recording, remediation steps are to delete existing transcript on the recording and retry.

Ask AI

Success

The transcript.done webhook

If the async transcription job completes successfully, you will receive a `transcript.done` Artifact Status Change event when it completes:

JSON

```
{
  "event": "transcript.done",
  "data": {
    "data": {
      "code": "done",
      "sub_code": null,
      "updated_at": "2024-12-04T23:25:56.339940Z"
    },
    "transcript": {
      "id": "7d7387b1-874f-4950-a5b9-1ba6660e2f95",
      "metadata": {}
    },
    "recording": {
      "id": "03d06804-0cb2-42f8-a255-5b950dde7c57",
      "metadata": {}
    },
    "bot": {
      "id": "0b85d2f9-d54a-47f6-b28d-4c63229f4035",
      "metadata": {}
    }
  }
}
```

Fetching the transcript

Once you receive the `transcript.done` webhook, you can fetch the transcript data by calling [Retrieve Transcript](#) endpoint using its ID:

cURL

```
curl --request GET \
  --url https://us-east-1.recall.ai/api/v1/transcript/{TRANSCRIPT_ID}/ \
  --header "Authorization: $RECALLAI_API_KEY" \
  --header "accept: application/json"
```

Ask AI

The response will contain details about the transcript, such as the configuration used, as well

as a pre-signed URL to access the transcript data:

JSON

```
{
  "id": "3fa85f64-5717-4562-b3fc-2c963f66afa6",
  "recording": {
    "id": "03d06804-0cb2-42f8-a255-5b950dde7c57",
    "source": {
      "bot": {
        "id": "0b85d2f9-d54a-47f6-b28d-4c63229f4035"
      }
    }
  },
  "created_at": "2024-11-27T20:10:19.719Z",
  "expires_at": "2024-12-04T20:10:19.719Z",
  "status": {
    "code": "done",
    "sub_code": null,
    "updated_at": "2024-11-27T20:10:19.719Z"
  },
  "data": {
    "download_url": "..."
  },
  "diarization": {
    "use_separate_streams_when_available": false
  },
  "metadata": {
    "custom_field": "some_value"
  },
  "provider": {
    "assembly_ai_async": {
      "language": "en"
    }
  }
}
```

Error

The `transcript.error` webhook

If an async transcription job fails, you will receive a `transcript.failed` Artifact Status Change webhook event notifying you about the failure:

JSON



Ask AI

```
{
  "event": "transcript.failed",
  "data": {
    "data": {
      "code": string,
      "sub_code": string | null,
      "updated_at": string
    },
    "transcript": {
      "id": string,
      "metadata": object,
    },
    "recording": {
      "id": string,
      "metadata": object
    },
    "bot": {
      "id": string,
      "metadata": object
    } | null
  }
}
```

 Updated 18 days ago

← Real-Time Transcription

Meeting Caption Transcription →

Did this page help you?  Yes  No

Ask AI