

# OpenASR Evaluation ToolKit

---

**Version:** 0.1.2

**Date:** August 6th, 2021

## Table of Content

---

[Overview](#)

[Setup](#)

[Usage](#)

[Reference/Submission Structure](#)

[Report a Bug](#)

[Authors](#)

[Copyright](#)

## Overview

---

The 2020 Open Automatic Speech Recognition Challenge (OpenASR20) was an open challenge created out of the IARPA (Intelligence Advanced Research Projects Activity) MATERIAL (Machine Translation for English Retrieval of Information in Any Language) Program that encompasses more tasks, including CLIR (cross-language information retrieval), domain classification, language identification, and summarization. The goal of OpenASR20 was to assess the state of the art of ASR technologies for low-resource languages.

In 2021, ASR under low-resource language constraints is being offered again, but with new languages and case-sensitive scoring added.

This README file describes a set of tools used in OpenASR. Developers can use these tools to mimic the process NIST uses to process and score their submissions locally.

- STM reference file converter tool - converts the provided transcript file into an STM reference file format.
- CTM system file generation tool - generates a perfect CTM system file from an STM reference file (for testing purpose).
- Validation tool - confirms that a submission follows the rules set in the OpenASR21 Evaluation Plan.

## Setup

---

The tools included in the package can be run under a shell terminal and have been confirmed to work under OS X and Linux.

### Prerequisites

- [Python](#) >= 3.0
- [Pandas](#) >= 0.23.0

### Installation

Uncompress the tool package using the following command:

```
tar -xvzf /path/to/openasr_toolkit-x.x.x.tgz
```

## Usage

---

The following commands should be run within the `openasr_toolkit-x.x.x/` directory.

### Reference File Generation Tool

To get the **reference file generation tool** usage:

```
python3 scripts/OpenASR_convert_reference_transcript.py -h
```

To convert the given transcript file into an STM reference file:

```
python3 scripts/OpenASR_convert_reference_transcript.py -l <language> -i <transcript_file> -c <channel> -o <output_directory>
```

### Required Arguments

- `-l` : indicates the language of the input transcript file (e.g., Mandarin)
- `-i` : input file containing the transcript in dev/train data
- `-c` : indicates the channel ID. If the file is part of conversational telephone speech, the inline channel is designated as 1 while the outline channel is designated as 2. Otherwise, the channel ID is 1.

### Optional Arguments

- `-o` : output directory containing the transcript in STM format

### Example

```
python3 scripts/OpenASR_convert_reference_transcript.py -l Cantonese -i BABEL_BP_101_10470_20111118_172644_inLine.txt -c 1 -o tmp
```

## System File Generation Tool

To get the **system file generation tool** usage:

```
python3 scripts/OpenASR_generate_ctm_file.py -h
```

To create a perfect CTM file from an STM reference file:

The following command should be run within the `openasr_toolkit/` directory.

```
python3 scripts/OpenASR_generate_ctm_file.py -f <STM_file> -o <output_dir>
```

### Required Arguments

- `-f` : STM reference file

### Optional Arguments

- `-o` : output directory containing the perfect corresponding CTM system file

### Example

```
python3 scripts/OpenASR_generate_ctm_file.py -f BABEL_BP_101_10470_20111118_172644_inLine.stm -o tmp
```

## Validation Tool

To get the **validation tool** usage:

```
python3 OpenASR_validate_submission.py -h
```

To get the **validation tool version** installed:

```
python3 OpenASR_validate_submission.py -v
```

### ASR Validation

To **validate the format of a submission** directory against a reference directory to make sure the files under the `submission_directory` and the `reference_directory` have the same shape.

```
python3 OpenASR_validate_submission.py -s <submission_dir> -ref <reference_dir> -l <language>
```

### Required Arguments

- `-s` : directory containing a submission
- `-ref` : reference directory

### Optional Arguments

- `-l` : indicates the language of the submission files. If not given, the default language is English.

### Example

```
python3 OpenASR_validate_submission.py validate -s test_suite_validation/submissions/PASS_submission \
-ref test_suite_validation/reference
```

## Reference/Submission Structure

### Reference Structure

The reference files follow one of the following two formats, depending on whether the audio was recorded in one or two channels:

`<DocID>_<ChannelID>.stm` (for one of two channels of two-channel recording)

<DocID>.stm (for single channel recording)

The reference directory expected by the validation has the following structure:

```
REFERENCE_DIR/  
  REFERENCE_FILE  
  ...
```

An example is available in `test_suite_validation/reference`

## Submission Structure

The system output files follow one of the following two formats, depending on whether the audio was recorded in one or two channels:

<DocID>\_<ChannelID>.ctm (for one of two channels of two-channel recording)

<DocID>.ctm (for single channel recording)

The submission directory expected by the validation has the following structure:

```
SUBMISSION_DIR/  
  SYSTEM_OUTPUT_FILE  
  ...
```

An example is available in `test_suite_validation/submissions/PASS_submission`

## Report a Bug

---

Please send bug reports to [openasr\\_poc@nist.gov](mailto:openasr_poc@nist.gov)

For the bug report to be useful, please include the commandline, input files, output files, and any error messages. Please also include the OS version as well as Python version.

### Test case bug report

A test suite has been developed and is runnable using the following command within the `openasr_toolkit/` directory:

This will run the tests against a set of submissions and reference files available under `test_suite_validation/submissions` and `test_suite_validation/reference`.

```
python3 -m unittest
```

## Authors

---

Sarra Chouder <[sarra.chouder@nist.gov](mailto:sarra.chouder@nist.gov)>

Jennifer Yu <[yan.yu@nist.gov](mailto:yan.yu@nist.gov)>

## Copyright

---

This software was developed at the National Institute of Standards and Technology by employees of the Federal Government in the course of their official duties. Pursuant to Title 17 Section 105 of the United States Code this software is not subject to copyright protection within the United States and is in the public domain. MATERIAL is an experimental system. NIST assumes no responsibility whatsoever for its use by any party, and makes no guarantees, expressed or implied, about its quality, reliability, or any other characteristic.

We would appreciate acknowledgement if the software is used. This software can be redistributed and/or modified freely provided that any derivative works bear some notice that they are derived from it, and any modified versions bear some notice that they have been modified.

THIS SOFTWARE IS PROVIDED "AS IS." With regard to this software, NIST MAKES NO EXPRESS OR IMPLIED WARRANTY AS TO ANY MATTER WHATSOEVER, INCLUDING MERCHANTABILITY, OR FITNESS FOR A PARTICULAR PURPOSE.