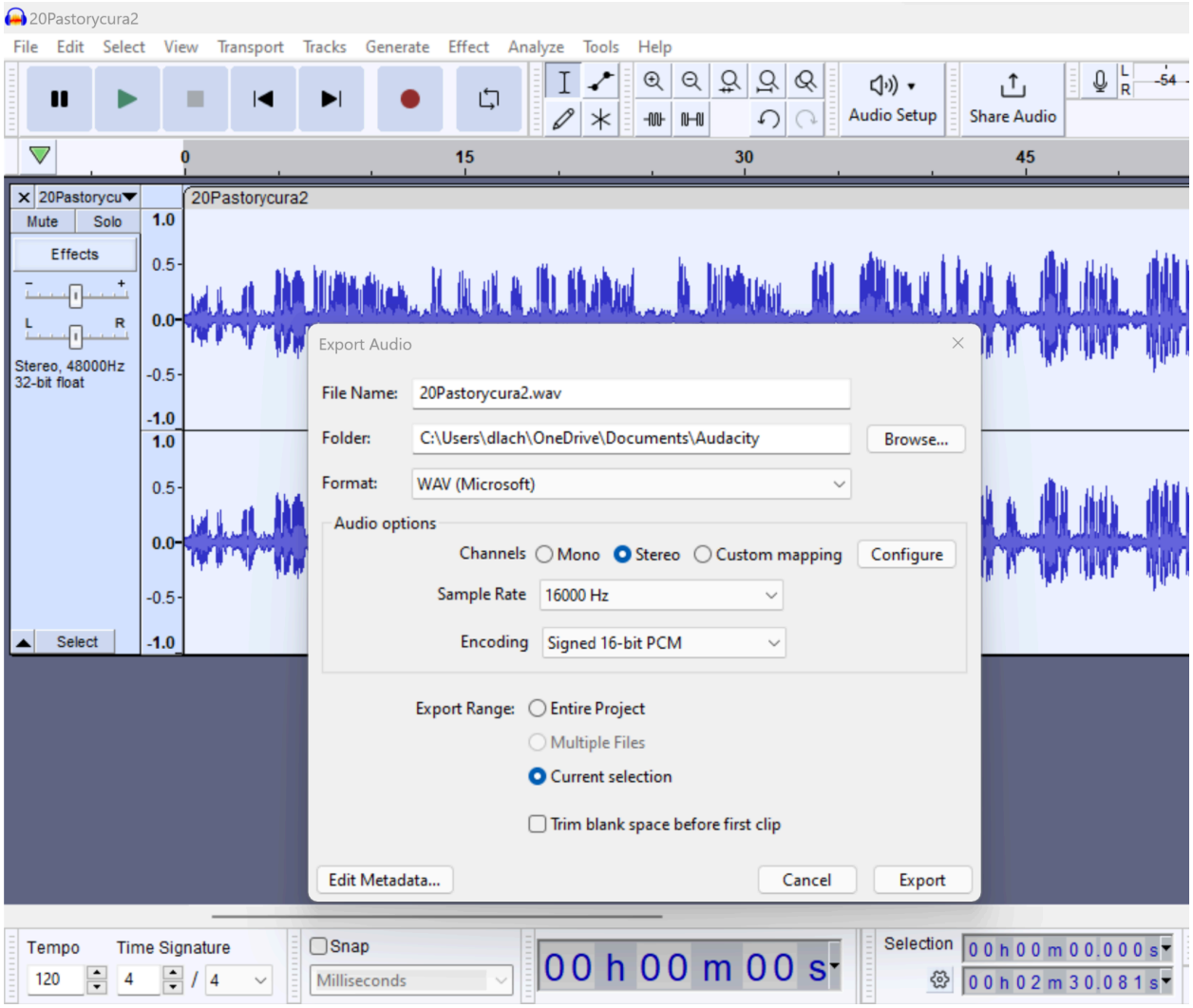
How to transcribe a Totonac audio recording

1) For Facebook MMS transcription, the audio needs to be saved with a 16 KHz sample rate. The clip length limit due to memory resources is approximately 2 ½ minutes. Audacity can be used as in the screenshot below to save a selected clip in the proper sample rate.



2) To use Facebook MMS transcription, go to this github page https://github.com/facebookresearch/fairseq/tree/main/examples/mms

   In the ASR section, there are trained language models listed from least to most resource intensive.

Finetuned models

ASR

Model	Languages	Dataset	Model	Dictionary*	Supported languages	
MMS-1B:FL102	102	FLEURS	download	download	download	🤗 Hub
MMS-1B:L1107	1107	MMS-lab	download	download	download	🤗 Hub
MMS-1B-all	1162	MMS-lab + FLEURS + CV + VP + MLS	download	download	download	🤗 Hub

There are two dialects of Totonac included in the Supported Languages that are trained in the larger models. The Totonac dialects are identified by Iso Code and Language Name (Region).

Iso Code	Language Name
toc	Totonac, Coyutla
tos	Totonac, Highland

For the purposes of transcribing the Coatepec recordings available, the Totonac, Highland, dialect provided the most accurate transcription based on McQuown Levy annotations.

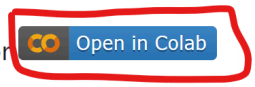
3) For transcription, Google Colab can be used to run the supplied python script in the Commands to run Inference section of the github page highlighted below:

Commands to run inference

ASR

Run this command to transcribe one or more audio files:

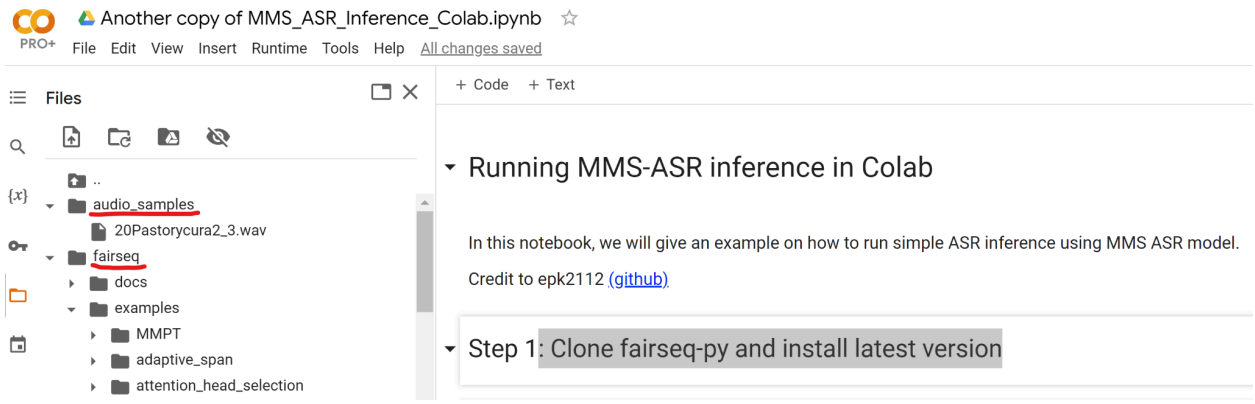
```
cd /path/to/fairseq-py/
python examples/mms/asr/infer/mms_infer.py --model "/path/to/asr/model" --lang lang_code \
  --audio "/path/to/audio_1.wav" "/path/to/audio_2.wav" "/path/to/audio_3.wav"
```

We also provide an Ipython notebook example inside `asr/tutorial` folder ipynb or 

Note that in order to use the larger language models containing Totonac, a Google Colab-Pro subscription is required for increased RAM capacity.

4) Follow these steps within the MMS_ASR_Inference_Colab notebook. Note that execution times will vary depending on internet connection speeds, but each step can be expected to take approximately a few minutes.

   a) Create an audio samples folder under the main content directory and upload the audio file(s) to be transcribed.



   b) Execute “Step 1 Clone fairseq-py and install latest version” in the notebook, this will load the code into the fairseq folder that will do the transcription.

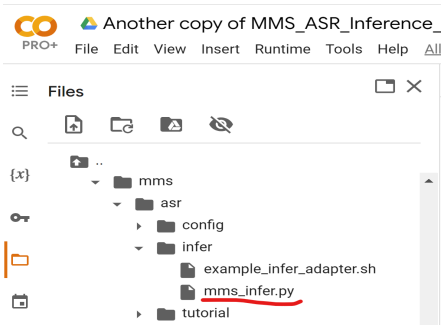
      Note that for audio recordings exceeding approximately 30 seconds, one of the configuration files in the fairseq folder needs to have a parameter increased, otherwise the following error will occur:

      AssertionError: Sentences lengths should not exceed max_tokens=1440000

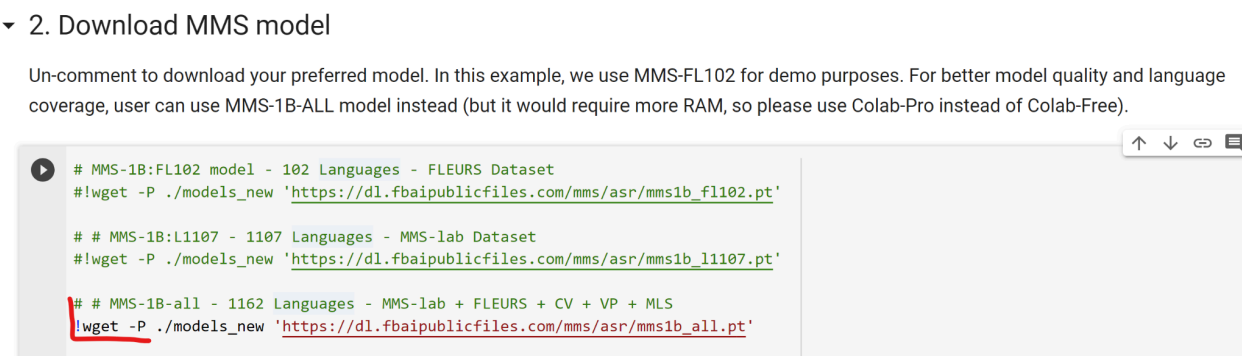
      Download a copy of the mms_infer.py file from the fairseq\examples\mms\infer folder and increase the following parameter from its default setting by adding a 0 to it:

      default: dataset.max_tokens=1440000
      required: dataset.max_tokens=14400000

      Upload the modified mms_infer.py file back to the infer folder.



c) Execute Step “2. Download MMS Model” to download the language model with the sought after Iso Code. In the case of Totonac, download the largest MMS-1B-all model.



d) Skip Step “3. Prepare audio file” as this is prepared using Audacity or similar external audio editing software. Note, however, this step can be used to convert an mp3 to wav file if desired.

e) In Step “4: Run Inference and transcribe your audio(s)”, edit this line of code to specify the model used, the language Iso Code to use for the transcription, and the name of the uploaded audio file itself. Note that you can transcribe multiple audio files that have been uploaded one after the other by duplicating this line for each audio file to transcribe. Each transcription takes approximately one minute.

```
!python examples/mms/asr/infer/mms_infer.py --model
"/content/fairseq/models_new/mms1b_all.pt" --lang "tos"
--audio "/content/audio_samples/20Pastorycura2_3.wav"
```

If an Out of Memory error is received, split the audio recording into shorter clips and upload them to continue, adding a separate python code line for each new shortened clip name, then re-execute Step 4.

A successful transcription will appear similar to the following:

Input: /content/audio_samples/20Pastorycura2_3.wav
Output: tlan mąhuiyą́n lakcatzanąn pąxquiyą́n tlan chakąniyą́n milhákąt pacs tlan tlan quimąhuí quimpąxquiy xmąn huą huą ni quimakapąxuhuay aktzutupupacu kalhi ctitapuxuhuahu xlitahuilh tu kalhiy aquit puhuán pi ni quilá nalayąchątum ti xlá chí huą pąli ni catuntujlat ni milą puuaná camaquitá tamą libro xatutlancá nacacxilha lą huán tzuculh ácxilhla ni xlicąnąxpatzamą clib c-libro caj xlacapąstacmą tlancaliyą cajuą xmąkalhtupitzimimą xlibro chíkahuasa ni xcatziy hpąpi caj xlacapąstacmą pąli lą nahuaniy acxni lacapąstackolh unú huan huánilh

huąpį huą tzumąt lihui xakuán xastląn xatlihuaka pį huą natątapųxuhuą cskalana palęcua xlįaktątipapąn camala acxilhtą chiyų ųnú huan huánilh chí kahuasa nį tzinú xcatzį xkalhtahuaką cąnájlalh tų huánilh pąliy inį xaccatzįy pąpį chuną quimątzankąní huą tląn nįtú capuhuanti inį naquintlahuaniyą lįtlán naquimąpąhuaniyą tzinú tų naclįtamąhuanán huą kahuasa tląn huą pąliy tuncan maxquįlh laktiypęsua canąpąxtacatzįnilh ąpuntzú huą lįhuipąxuhuą alh kahuasa c-xchic nį catzįlh pąpį caj akskahuicą