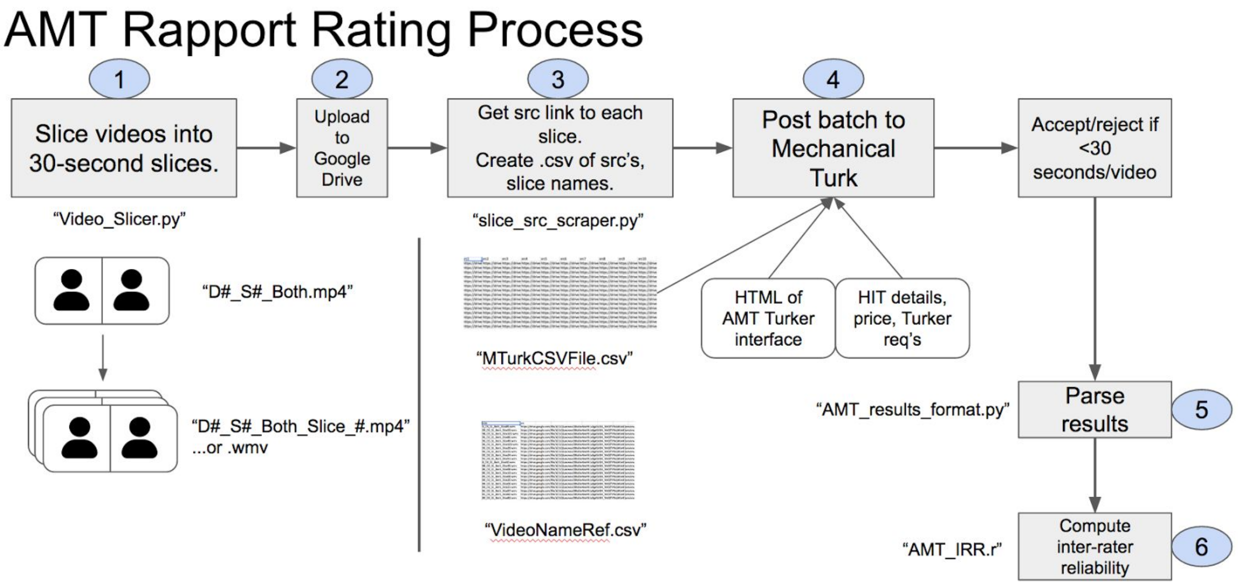
Thin-Slice Rapport Rating using Amazon Mechanical Turk


AMT Rapport Rating Process
1
Slice videos into 30-second slices.
"Video_Slicer.py"
"D#_S#_Both.mp4"
"D#_S#_Both_Slice_#.mp4" ...or .wmv
2
Upload to Google Drive
3
Get src link to each slice. Create .csv of src's, slice names.
"slice_src_scraper.py"
"MTurkCSVFile.csv"
"VideoNameRef.csv"
4
Post batch to Mechanical Turk
HTML of AMT Turker interface
HIT details, price, Turker req's
Accept/reject if <30 seconds/video
5
Parse results
"AMT_results_format.py"
6
Compute inter-rater reliability
"AMT_IRR.r"

1. **Slice videos**
   a. Place your video files in a single folder, by themselves.
   b. Write down the total number of 30-second slices in each file.
   c. Open "Video_Slicer.py" in a Python IDE or text editor.
   d. Change the lines that need changing and run the script
      i. Filepath
      ii. Video file names
      iii. Total # of slices
      iv. Change any other formatting as necessary to reflect your video file names
      v. If working in a Python IDE (PyCharm, Atom), click run.
      vi. If working in a text editor (Sublime Text)
         1. Save the .py file
         2. Open "Terminal" (if on Mac)
         3. Change directory to the folder where Video_Slicer.py is (using "cd Desktop\Slicing" etc)
         4. Type "python Video_Slicer.py"
   e. Find the output file "video_slicer.bat" and move it to a Windows machine, where you have "ffmpeg" installed. ("C:\Users\mmadaio\Desktop\ffmpeg\bin") **-** Make sure it's in the bin folder within the "ffmpeg" folder.

f. In Windows Command Prompt, change directory (also with the “cd <filepath>” command) to that bin folder.
g. Type “video_slicer.bat”
h. Watch it slice your videos!

2. **Upload slices to a Google Drive folder**
    a. We use an Articulab account that has a large amount of storage. Ask Michael to share it with you.
    b. I would recommend creating a folder for each Batch, and a folder for the Training Videos.
    c. Verify that all videos have been successfully uploaded.
    d. Select all videos, right click, select Share > Advanced, and check the box for “Disable options to download, print, and copy… etc”.

3. **Get Google Drive link in a format that Mechanical Turk can use**
    a. Follow the instructions here to authorize your Google account to access the Google Drive API, but do Steps 1 and 2 only.
    b. Instead of running the quickstart.py file, you’ll be running the “slice_src_scraper.py” file. You can (like before) do this through a Python IDE like PyCharm, or through Terminal.
    c. Make sure you have the “client_secret.json” file in the same folder as “slice_src_scraper.py”, and that your “drive-python-quickstart.json” file is wherever the script says it is (It’s in my Documents folder, but you can change as you like).
    d. Change necessary parts:
        i. Change the Google Drive folder ID’s (called “q” here) to reflect the actual folder for the Batch you’re working on, and for the Training Video folder.
        ii. Change the “pageSize” variable to be equal or more than the number of slices in your folder. None of mine are over 800, so I keep it at 800, and it works fine with batches of fewer slices. It just gets up to 800 results from the Google Drive folder.
        iii. Change the output files (“MTurkCSVFile_batch#.csv” and “VideoNameRef_batch#.csv”) to reflect the batch you’re on.
        iv. Since this format uses 10 slices in each Mechanical Turk “HIT”, we only take a number of slices that’s divisible by 10. The rest get flagged with an “M” (for “missing” - they’re not missing, just not included in the batch), and can be included in later batches. Alternatively, you may want to only put a number of slices on Google Drive divisible by 10, to avoid this problem entirely.
        v. To adapt to a different project: think about how the file naming convention will impact the script any time it looks for a certain part of the file name (“Dyad_#” or “Session_#”). Change if necessary.
    e. Run script.

4. **Post batch to Mechanical Turk**
    a. Create an Amazon Mechanical Turk Requester account here.
    b. Add sufficient funds to the account under:
        i. My Account > Purchase Pre-Paid HITs
        ii. Use calculator here to figure out how much you need. See below for price.
    c. Create New Project (if first time) or Create New Batch from Existing Project.
    d. Set details:
        i. We pay $1.7, which equals ~$7.50 an hour, calculated with the average completion time for our HITs.
        ii. 4 "assignments" (workers) per HIT (10 slices)
        iii. Time allotted: 25 minutes (or more)
        iv. HIT expires in: 7 days (though it usually takes a few hours)
        v. Auto-approve and pay workers in: 3 days (in case you forget to approve!)
        vi. Require that workers have "Masters": no (it doesn't improve quality)
        vii. Hit approval rate: >97%
        viii. Location is: US
        ix. Number of previous HITs approved: > 1000
        x. Visibility: public
    e. Design Layout: paste in the "Rapport_rating_AMT_Interface.html"
        i. Change as necessary:
            1. Rapport description
            2. Likert scale labels
            3. Free text entry field
        ii. Save, finish
    f. Click Publish Batch
        i. Upload the MTurkCSVFile_Batch#.csv
        ii. Click through.

5. **Parse results**
    a. Download results when finished.
    b. Based on average time completed, change payment for next batch.
    c. Accept or reject HITs
        i. In the past, we have rejected HITs if their time to complete was less than the total time to watch all videos, and we tried to make this clear in the HIT instructions.
        ii. In Excel, write a simple formula in the Accept and Reject columns:
            1. Accept formula:
                a. =if("WorkTimeinSeconds" > TotalTime, "x", "")
                b. Replace "WorkTimeinSeconds" with the column that that value is in.

- c. Replace TotalTime with whatever value you chose (e.g. 300).
- 2. In the Reject column, paste that same formula, but with the reason why it was rejected.
  - a. =if(“WorkTimeinSeconds” > TotalTime, “”, “Did not watch all of the videos in their entirety.”)
- iii. Re-upload to AMT Results page.
- iv. Wait for the remaining HITs you rejected to be completed by other people.
- v. Check your email and respond to any Turker’s concerns; handle appropriately and professionally.

- d. Open the results file and do some minor cleaning:
  - i. Remove all rejected rows
  - ii. Remove all commas
  - iii. Move the “Rapport10” column to the right place (after “Rapport9” not “Rapport1”).
  - iv. Make sure there are no strange line breaks in the free text cells that might cause your next script to break.
  - v. [If someone wants to write a quick Python script to automate those, that’d be cool.]
- e. Open “AMT_results_format.py”
  - i. Change the batch number in “os.chdir(“batch#/”)”, the “VideoNameRef_batch#.csv”, and your output file in “convert_to_csv” at the end.
  - ii. Make sure you have a subfolder created called “R_output” inside a folder called “batch#”. Make sure the cleaned AMT results csv is in this folder, along with the VideoNameRef_batch#.csv file.
  - iii. Run script.
  - iv. It should output a “Batch#_results_formatted.csv” file and N individual csv files, one for each HIT. These are also in the R_output folder, in a stripped down format, with just the ratings for the 10 slices by 4 raters for each HIT.

6. Calculate Inter-Rater Reliability (IRR)

- a. Move the R_output folder to your main R folder for this project. Add “_batch#” to the end of that folder name.
- b. Change the input folder path
- c. Change the number of HITs in the for loop
- d. Change the output file names.
- e. Output: 1 column of Krippendorff’s alpha, and one for ICC.
- f. [If someone wants to change these last few lines to have it output a single file with 3 columns, that’d be great!]