Step 1. Prepare speech data

1.1 Retrieve speech data from the server

Open terminal.

Log into server computer:

ssh server address

e.g., ssh webexperiments@hjpatt-136.umd.edu

Type in password (nothing will appear when you type):

(type in server password)

Access the folder that contains your speech data

cd path/to/your_folder

e.g., cd Sites/Web_Experiments/Phillips/Rosa/uploads

Check that the folder contains your speech data

ls

Go back to the main folder

cd ..

Compress the folder containing speech files to zip:

zip -r name_of_zip ~/path/to/folder/containing/recording_files/

e.g., zip -r uploads.zip uploads/

→ compresses ‘uploads’ folder and name the zipped folder ‘uploads.zip’

Check that the zip file is created in the main folder:

ls

Exit the server:

exit

Download the zip file from the server to your local computer (requires server password):

scp -r server_address:~/path/to/zip_in_the_server ~/path/to/local_destination_folder

(type in server password)

e.g., scp -r webexperiments@hjpatt-136.umd.edu:Sites/Web_Experiments/Phillips/Rosa/recordings.zip ~/PlanetCloze/data/backup

→ downloads ‘recordings.zip’ to a local folder ‘backup’

Output of this section: a zip file containing speech files saved locally on your computer

1.2 Convert .webm files into .wav files

Unzip all individual speech files using terminal:

```
cd path_to_folder
unzip \*.zip
```

Convert the .webm files into .wav files:
(This step requires having *ffmpeg* installed. You can install it using Homebrew, following the suggestion here: https://superuser.com/questions/624561/install-ffmpeg-on-os-x)

```
cd path_to_folder
for i in *.webm; do ffmpeg -i "$i" "${i%.*}.wav"; done
```

Rename the original folder to “webm_wav”

Make a new folder “wav” and copy-paste only the .wav files into the folder

Output of Step 1: “wav” folder containing all speech files in .wav format

Step 2. Get transcriptions

1. Download the *get_transcriptions.zip* file, unzip, and upload it to your Google Drive (under My Drive)

2. Replace the *wav* folder with your local *wav* folder containing all .wav speech files

3. Open and run *get_transcriptions.ipynb* using Google Colaboratory

Output of Step 2: “tg_transc” folder containing textgrid files with transcriptions

Step 3. Get timing data

1. Download the **tg_transc** folder to your local computer

2. Run Montreal Forced Aligner: https://montreal-forced-aligner.readthedocs.io/en/latest/

3. Store results in a folder and name it **tg_raw**

How to run the Montreal Forced Aligner:

(to be added)

Output of Step 3: “tg_raw” folder containing textgrid files with transcriptions + timing data

Step 4. Manual inspection

1. Prepare **tg_raw** and **wav** folders

2. Download the **Check_and_Edit** script from the Github repository

3. Open Praat

4. Praat --> Open Praat script --> Check_and_Edit --> Run
   Designate paths:
   - tg_path: path to **tg_raw** folder
   - wav_path: path to **wav** folder

5. Check and edit transcriptions and alignments (changes are automatically saved)

6. Rename **tg_raw** to **tg_checked**

7. Upload **tg_checked** folder to the **get_transcriptions** folder in Google Drive

Output of Step 4: “tg_checked” folder in your “get_transcriptions” folder in Google Drive; containing textgrid files with transcriptions + timing data that have been manually inspected

Step 5. Generate results file

Open and run *process_results.ipynb* using Google Colaboratory

Output of Step 5: a csv file containing transcriptions + timing measures from all textgrid files

Supplementary Notes

Useful terminal commands

To see a list of all files in a folder, ordered (most recent files on top):

ls -1 -tr

To create a folder:

mkdir your_folder_name

e.g., mkdir recordings

→ creates a folder named ‘recordings’ in the location you are currently working in

To compress files in a folder:

zip -r name_of_zip ~/path/to/folder_containing_files/

e.g., zip -r uploads.zip uploads/

→ compresses ‘uploads’ folder and name the zipped folder ‘uploads.zip’

To unzip a folder:

unzip folder_name.zip -d ~/path/to/destination/name_of_unzipped_folder

e.g., unzip ‘uploads.zip -d ~/uploads_unzipped

→ unzips ‘uploads’ folder and name the unzipped folder ‘uploads_unzipped’

To upload and download files to and from the server:

<u>On your local computer:</u>

To upload a local folder to the server:

scp -r ~/path/to/your/local/folder.zip webexperiments@hjpatt-136.umd.edu:~/path/to/your_folder_in_the_server

(It will require the server password. Type in password and press enter.)

e.g., scp -r ~/images.zip webexperiments@hjpatt-136.umd.edu:Sites/Web_Experiments/Phillips/Rosa/PlanetCloze/stims

→ uploads ‘images.zip’ to the ‘stims’ folder on the server computer

To download files from the server to your local computer:

scp -r webexperiments@hjpatt-136.umd.edu:~/path/to/your_folder_in_the_server ~/path/to/your/local_folder

(It will require the server password. Type in password and press enter.)

e.g., scp -r webexperiments@hjpatt-136.umd.edu:Sites/Web_Experiments/Phillips/Rosa/recordings.zip ~/PlanetCloze/data/backup

→ downloads ‘recordings.zip’ to a local folder ‘backup’