Pipeline for normal (reference) english speech and dutch speech

**Step 1: Locate Audio files**

INPUT: Nothing

ACTIONS: Asks user if files exist

YES: Locate these files (using part of forced aligner (run.sh))

Convert to (16kHz) WAV if necessary

NO: Record these files

RESULT: audio files (.wav)

**Step 2: Create orthographic transcription for every audio file**

INPUT: audio files (.wav)

ACTIONS: Asks user if txt files are already in the same folder for every file and have the names as the audio files (?)

YES: Use these files

NO: Create them.

1. Use ASR to transcribe every audio file

2. Do manual post processing

3. If not possible, let users create transcription for every file manually

RESULT: audio files (.wav) + matching transcription of every file (.tg), in the same folder and with the same name

**Step 3: Do forced alignment**

INPUT: audio files (.wav) + matching transcription of every file (.tg), in the same folder and with the same name

ACTIONS: run **/vol/tensusers2/mvos/clst-asr\_forced-aligner/run.sh**

RESULT: Automatic segmentation of wav files. + list of out of vocabulary words

**Step 4: Add out of vocabulary words to the dictionary**

INPUT: Dictionary and list of out of vocabulary words

ACTIONS: Use G2P (grapheme to phoneme converter) to get phoneme representation of OOV words and add these OOV words to dictionary

RESULT: The updated dictionary

**Step 5: Do forced alignment again**

INPUT: audio files (.wav) + matching transcription of every file (.tg), in the same folder and with the same name + Updated dictionary

ACTIONS: run **/vol/tensusers2/mvos/clst-asr\_forced-aligner/run.sh**

RESULT: Automatic segmentation of wav files.

Note: Step 3 through 5 should ideally be done automatically, so that the user does not have to specify anything the second time the forced alignment is done (as these should be remembered from the first time). This requires changes to be made in the run.sh file.

**Step 6: Ask user to check automatic segmentation**

INPUT: Automatic segmentation of wav files.

ACTIONS: Does user want to check segmentations (Y/N). This can be done with an already existing Praat script

RESULT: Segmentations of wav files (checked or not)

**Step 7: Run praat script to get statistics**

INPUT: Wav files and checked textgrids

ACTIONS: select word + phoneme level, all statistics, which are options in the praat script *LabeledSegmentsAnalysisISLA\_improved.praat*

RESULT: Files containing all the statistics (features) for word and phone level

**Step 8: Extract statistics from statistics file**

INPUT: File containing all the statistics (features) for word and phone level

ACTIONS: run python script (extract\_results\_new.py)

RESULT: Excel file containing a summarization of the input file

**Step 9: [OPTIONAL] determine intelligibility scores**

INPUT: File containing all the statistics (features) for word and phone level

ACTIONS: Use tool from Wei Xue & Micha Hulsbosch

RESULT: intelligibility scores from features