Exercise 8 Working with Aligned Data

In a previous exercise, we force-aligned words and their segments, and also did a little hand-correction of the alignments. Now that we've got some relatively reliable phone-level alignments, we're going to explore their search and annotation possibilities.

In this exercise you will

- see how the alignments affect speech rate computations,
- automatically annotate pauses between words,
- search for tokens of individual, time-aligned segments,
- create time-aligned segment annotations via Praat, and
- automatically extract features from search-result intervals using Praat

---

You will have noticed that HTK introduces pauses between the words – almost always at the beginnings and ends of utterances, and also sometimes in between. These pauses will affect the speech-rate statistics that are computed by the Statistics layer manager for the speech rate layer that we set up earlier. Now that there are pauses, these are excluded (i.e. not counted as speech) for the speech-rate statistics.

However, for this change to be affected, the speech rate layer needs to be regenerated.

1. Go to the 'transcripts' page and open the transcript `UC207YW.eaf`, if you don't still have it open.
2. Tick the *speech rate* layer for display, and make a note of some of the speech rate annotations (which ones are there depends on what you selected as the scopes for the layer).
3. At the top of the transcript, in the *formats* menu, click the 'regenerate layers' option.
   This will display a list of managed layers.
4. Select the *speech rate* layer and click *Regenerate.*
   You will see a progress bar while the statistics for the transcript are computed again.
5. Once that's finished, go back to the transcript and check the speech rate annotations again. You should see that at least some of them are different.

These pauses can also be directly annotated, in case you're interesting in finding them for analysis.

6. Add a new word layer with the following attributes:
   - *description:* `Length of previous pause`
   - *short description:* `previous pause`
   - *layer type:* 'Number'
   - *layer manager:* 'Context Layer Manager'
   - *alignment:* 'Not aligned'
   - *generate:* 'Always'
   - *project:* 'pauses'
7. Configure the layer with the following settings:
   - *Layer to summarize:* 'transcript'
   - Select the 'Pause Detection' option
   - Leave *Minimum Pause Length:* unticked
   - Leave *Maximum Pause Length:* unticked

   You may be interested in looking at the online help to find out what kinds of annotations the Context layermanager can create.
   Click *Save* and *Regenerate.*
8. Once the layer is generated, go back to the `UC207YW.eaf` transcript and display the *previous pause* layer (you might have to tick the pauses project to make the layer option visible).
   (You also might want to untick some of the other layers, to avoid clutter.)

9. You should see that a subset of words are annotated with a number, which is the length of the pause before that word.
   Open one or two such utterances in Praat to check that the lengths are accurate.
   You may notice that pauses in the middle of utterances are always right, but the pause before the first word in the utterance seems wrong. See if you can figure out why.

Now we're going to search for some instances of vowels of interest (the 'FLEECE' vowel in this case), and annotate them with formant measurements.

10. First of all, create a new project called `formants`
11. Now click the *segment layers* menu option.
    We're going to add a layer that annotates segments – i.e. individual phones within words
12. Fill in the new-layer form at the bottom with the following details:
    - *description:* `First formant`
    - *short description:* `F1`
    - *layer type:* 'Number'
    - *alignment:* 'Time Stamps'
    - *layer manager:* (leave it blank, this is a custom layer)
    - *generate:* (not relevant as it's not a managed layer)
    - *project:* 'formants'

    Click *New.*
13. Select *search*, select everyone, and click *Layered Search.*
14. Tick the *segments* layer, and click *Set Search Matrix.*
    The segments layer contains annotations at the sub-word level – i.e. there are potentially multiple annotations per word, each annotation representing a phone of the word. You will see that, as with other layers, there is a box on the segments layer for a regular expression.
    As with other patterns in the search matrix, the pattern that you enter in the box is matched against individual annotations. So if you enter `i` in the in the box, it will match each 'FLEECE' vowel segment in each word in the database.

    NB It's important to realise that if you enter a pattern that would match more than a single character on this layer (i.e. more than a single phoneme) then no search results will be returned, because each annotation on this layer is only a single character long (remember the DISC encoding uses one character per phoneme). For example, if you enter `.*IN` for your search, intending to match all words ending in "…ing", then no results will be returned, because no single segment will ever match that pattern.
15. We want to search for all instances of the 'FLEECE' vowel, so enter `i` in the segments pattern box.
16. In our particular database, if there's any annotation on the segments layer, then we can be sure that it's been aligned at least by HTK (if not manually corrected). However, there are configurations of CELEX layers that would put unaligned annotations on that layer.
    To make sure we only get words that have been aligned by HTK or manually, tick the 'only match words that are aligned' option.
17. Click *Search.*
    Once the search is finished, you should notice that the only transcripts returned are those that include speakers you've done force-alignment on.
18. Click on the first result, to open its transcript.
19. Scroll to the top and tick the 'include empty layers' option.
20. Tick the 'formants' project option.
    This will reveal the (empty) *F1* layer below segments in the list of layers.
21. Tick the *F1* layer.
    The transcript will reload to include that layer (even though it's currently empty).

22. Click on the first search result (which is highlighted in green).
23. Select the 'Open TextGrid in Praat' option.
    This will open the audio of the line, with a TextGrid that includes the aligned words and segments. There's also a tier for the *F1* layer. (If there isn't, it's because the alignment setting for the layer is set to 'Not aligned' instead of 'Time Stamps')
24. In Praat, find our instance of the 'FLEECE' vowel, and click on a good point in the spectrogram for measuring it's F1 value.
25. Get the F1 value (hit the < `F1` > key on your keyboard)
26. Copy the value that is displayed on to the clipboard
    (i.e. select it and hit < `Ctrl` >+< `C` > on your keyboard)
27. Back in the TextGrid, add a boundary on the *F1* tier at that point
    (if it's the fourth tier, hit < `Ctrl` >+< `4` > on your keyboard)
28. Paste the F1 value that you copied earlier
    (i.e. hit < `Ctrl` >+< `V` > on your keyboard)
    Now you've annotated the vowel with its F1 value, and we want to save that annotation back to the LaBB-CAT database.
29. Click back on the LaBB-CAT transcript window.
30. Click the 'Import Changes' button that has appeared to the left of the line with the first match.
    You should see a message indicating that the annotations has been saved.
31. Repeat the above steps for the next few matches in the transcript.
32. Once you've added at least a handful of annotations on the *F1* layer, refresh the interactive transcript page (i.e. use the reload button in your browser, or the < `F5` > key).
    You will see that for each match you've annotated, the F1 value you entered appears below the corresponding word. The transcript doesn't display the time-alignment information, but that is also stored in the database.
33. Open the TextGrid for one of the lines you've annotated.
    You should see the annotation that you made is in the TextGrid, at the corresponding point in time.

These manual annotations are also searchable (so for example you could search for all the 'FLEECE' vowels with an F1 measure within a particular range), and can be exported in CSV search results. To see that in action:

34. Repeat the segment search we did before (i.e. all aligned vowels).
35. Once you see the results page, click the *[options]* link, and tick the *F1* layer.
36. Click *Export results to CSV*.
37. Save and open the resulting file.
    You will see that there are two columns, one called "Target F1", which contains the annotations you have made, and the other called "Target F1 start", which contains the time of the annotation, in seconds from the beginning of the recording.

We will now see that you can use Praat to automatically extract certain measurements, given start and end times from search results. In order for this to work, we first need to ensure that the LaBB-CAT server knows where Praat is installed:

38. In LaBB-CAT, click the *system attributes* menu option.
    This shows a form with various options on it, one of which is *Praat Path:*
39. If the *Praat Path:* option is blank, enter the location of Praat on your system, and click *Save*. The setting should be the path to the folder that contains praat, e.g.
    - on Windows, this might be "C:\Program Files\Praat"
    - on OS X, this might be "/Applications"

Alternatively, you can click *Choose File* at the bottom, select your "Praat.exe" program, and then click *Install Tool*.
If you have problems, try the online help for more information.

Let's say we want to extract F1 and F2 from all our aligned 'FLEECE' vowels.

40. We are going to use the CSV file you just extracted.
    If you don't have it any more, repeat the search and export the results to CSV.
    The CSV file includes a column called "Target Segments", which contains the annotation that matched the pattern (in this case they will all be "i"), and columns called "Target segments start" and "Target segments end" – these are the start and end times of each matching 'FLEECE' vowel.
    We are going to use these start/end times to get Praat to take formant measurements for us.

41. In LaBB-CAT, click the *upload* menu option.

42. Click the *process with praat* option.
    If the option is not there, it means that the *Praat Path:* attribute is not set. Go back to step 38 and ensure that *Praat Path:* is set befor continuing…

43. Click *Choose File* and select the CSV results that you saved above. Then click *Upload*.
    You will see a form to fill in, and the first couple of settings (*Transcript Name column:* and *Participant column:* should be already filled in.

44. For the *Start Time column:*, ensure that the 'Target segments start' option is selected.

45. For the *End Time column:*, ensure the 'Target segments end' option is selected.
    These two settings define the start/end times of the phone. For some measurements you might extract from Praat, processing signal that includes surrounding context is usually a good idea. You'll see there's a setting for that (which you can leave at the default of `0.5`s), and you will see options for various measurements.
    The default options are for 'F1' and 'F2' only, but if you feel like getting other measurements, feel free to tick those options too. There are also some *[advanced]* settings, which allow you to specify more detail about how Praat should do its computations. Again, feel free to look at those and try different settings.

46. Click *Run Batch*.
    You will see a progress bar while LaBB-CAT generates a Praat script and runs it with Praat.

47. Once Praat has finished processing the intervals, you will get a CSV file (you might have to click the *CSV file with measurements* link) – save and open it.
    You will see that it's a copy of the CSV file you uploaded, with some extra columns added on the right.
    Depending on your settings, this will include at least one column per measurement you selected (the formant columns also include on that contains the time at which the measurements were taken), and a final column called "Error" which is hopefully blank, but which might contain errors reported back by Praat (e.g. if it couldn't find the audio file or ran into any other problem during processing).

During this exercise, you have seen that the inter-word pauses created by forced alignment introduce the possibility of more accurate speech-rate statistics, and can themselves be automatically annotated, in case they are of interest for search or analysis.

You've also seen that you can create manual time-aligned annotation layers, which can be used to annotate phones (they can also be used for words or spans of words, by creating word layers or meta/freeform layers), and that you can also use intervals from CSV search results to extract acoustic measurements automatically using Praat.

Obviously these measurements are as reliable as the intervals themselves, so care needs to be taken to maximise the likelihood of good HTK alignments, and to check and possibly manually-correct those alignments.