Semi-automated pitch and formant analysis for mid-sized corpora

Although there is a culture of script-sharing among users of Praat speech analysis software,[1] differences in the interests of researchers and differences in the recording details of their corpora can render shared scripts difficult to use. One source of difficulty stems from the need for user-specified parameters for Praat's formant- and pitch-tracking algorithms. The optimal parameters vary depending on the voice of the talker and the quality of the recording; however, scripts may leave the parameters unspecified (in which case results will depend on whatever settings are in place on a particular computer at the time the script is run), or may hard-code the parameter values into the script's source code (making the parameters difficult to change for inexperienced programmers).

Even if a script makes Praat's algorithm parameters easy to change before running (e.g., through well-documented source code, or a dialog box presented when a script is initially run), there is a further problem with using scripts that are designed to *automatically* measure formants or pitch: corpora comprising diverse talkers often cannot be fruitfully analyzed with a one-size-fits-all set of algorithm parameters. Errors such as pitch halving, pitch doubling, and spurious or unstable formant tracking become increasingly likely as automated scripts are applied to larger and more diverse speech corpora. To be clear: the algorithms are performing well under such conditions; the problem is that they are being used as "fire and forget" tools, but are designed to be tweaked and tuned by human users in order to yield optimal results.

"Praat semiauto" is a collection of scripts designed to alleviate these problems by automating only the aspects of the measurement task that are error-free when performed by the computer unsupervised: opening files, finding the temporal spans of interest, placing the cursor, tabulating measurements, etc. What the semiauto tools do *not* automate is the act of measurement itself: when the scripts run, each measurement location is displayed onscreen at a consistent zoom level along with a dialog box displaying the current algorithm parameters. Users may alter the parameters at will until satisfied with the analysis, and the parameters are saved alongside the measurements to ensure reproducibility. Measurement locations are specified with a regular expression operating on a TextGrid tier, and users can control the number of measurements taken at each measurement location (i.e., the script can keep all analysis frames within the measurement window, or users can specify to take n measurements at n equally-spaced timepoints in the window).

REFERENCES

1. Boersma P, & Weenink D, "Praat: Doing phonetics by computer." 2016.