

Embarking on PoLaR Explorations: A Framework for Intonational Annotation and Analysis

Byron Ahn,
Nanette Veilleux,
Stefanie Shattuck-Hufnagel,
and Alejna Brugos

draft
November 2022

Contents

Contents	i
List of Figures	ii
List of Tables	ii
1 PoLaR: A Practical Guide	1
1.1 Functionality of PoLaR Plugin for Praat	1
1.1.1 Installing the Praat PoLaR plugin	1
1.1.2 Functions supported by the PoLaR plugin for Praat	2
1.1.3 General Guide to names/functionality of scripts	5
1.2 Workflow with PoLaR in Praat	5
1.2.1 Preparing TextGrids	5
1.2.2 PoLaR Labelling	6
1.2.3 Post-labelling analysis tools	7
Bibliography	9

List of Figures

List of Tables

Chapter 1

PoLaR: A Practical Guide

This chapter will overview the PoLaR plugin for Praat, as well as walk through a sample workflow using PoLaR from beginning (creating TextGrids) to end (analyzing results). As PoLaR and associated tools are under continual development, users are encouraged to check the PoLaR repository (Ahn et al. 2021; accessible from <https://www.polarlabels.com>) for the most up-to-date guidelines, tools, tutorials, and additional resources.

In addition to this practical guide, a labeller may find it useful for their labelling to have all PoLaR labels (Basic and Advanced) compiled in one place: this can be found in the appendix.

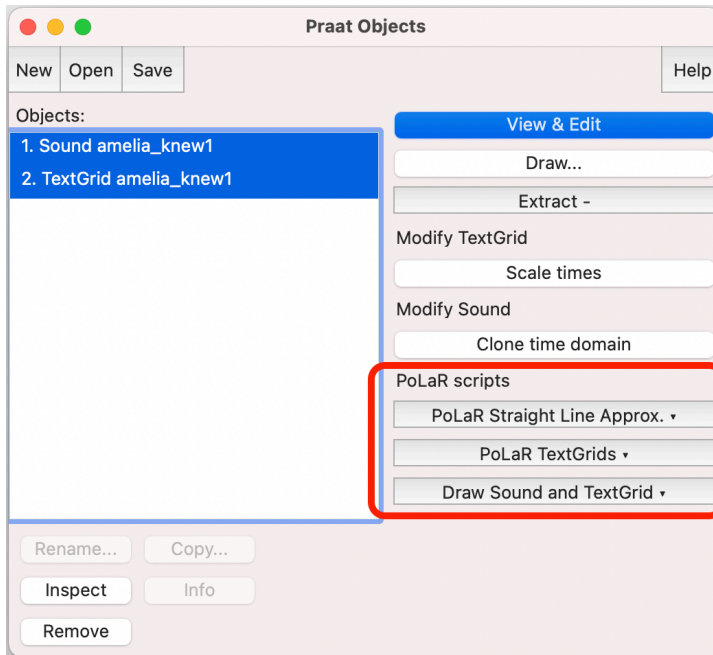
1.1 Functionality of PoLaR Plugin for Praat

We have created a suite of useful Praat scripts that automate set up, annotation and converting PoLaR labels to spreadsheets as well as draft ToBI labels. The full set of these scripts can be found and installed as a praat plugin, allowing these scripts to appear in the appropriate Praat menu. The most recent version of the PoLaR plug-in can be found at the PoLaR repository (Ahn et al. 2021; accessible from <https://www.polarlabels.com>), as a downloadable .zip file.

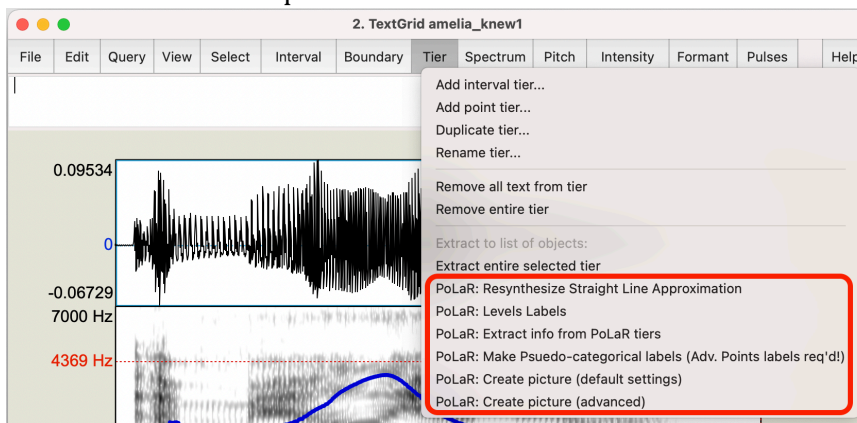
1.1.1 Installing the Praat PoLaR plugin

(Before installing, this might be an excellent time to update your Praat from praat.org.)

- Download **plugin_PoLaR-DATE.zip** (compressed file) from the OSF repository page
- Extract the script files from plugin_PoLaR-DATE.zip into a temporary folder (you will move the whole folder soon)
- Open the **README_FIRST.txt** file and follow the instructions to place the scripts in the correct **Praat preferences folder** (different on macOS/Windows/Linux; location described in this readme file) – this will install the plugin
- To test if it has been installed properly, open Praat (if Praat is open, close and re-open it), and open a Sound file and a TextGrid file.
 - Select both the Sound and TextGrid objects in the Praat Objects window
 - If the plugin has been installed directly, on the right hand side of the the Praat Objects window, you will see three drop-down menus: PoLaR Straight Line Approx., PoLaR TextGrids and Draw Sound and TextGrid



- Similarly, in an Editor window, the Tier tab menu shows PoLaR plugin commands – it has fewer options but includes links to the most critical scripts



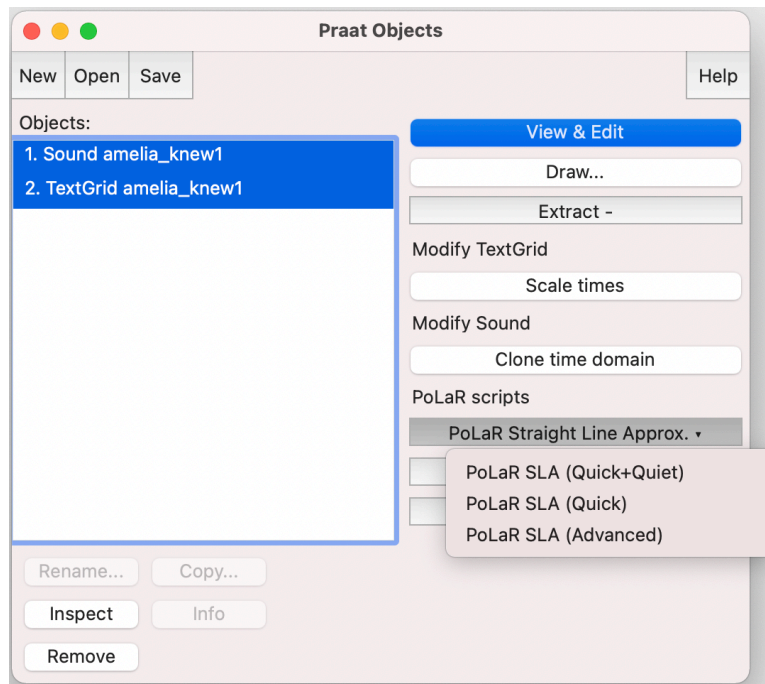
1.1.2 Functions supported by the PoLaR plugin for Praat

The PoLaR plugin for Praat enables several functions that may be useful for a PoLaR labeller. The bullet points below name some of these, with other functions (such as the function to resynthesize straight line approximations) being described in section 1.2 below.

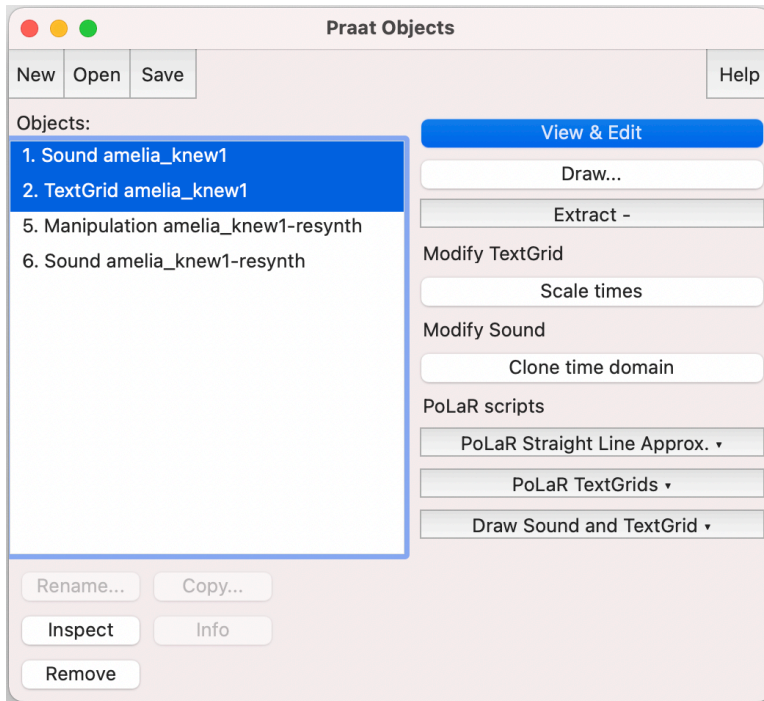
- The “Levels Labels” functionality automatically labels the Levels tier
 - Once a TextGrid has Points and Ranges annotated, the “Levels Labels” command automatically adds the appropriate values to the Levels tier at the appropriate times (see § 2.1.1.2).
 - Running this script can be done from within an Editor window (for a single Sound/TextGrid pair), or from the Objects window on multiple Sound/TextGrid pairs.
- The “Straight Line Approximations” commands can be used to create an f0 contour

based on PoLaR Points labels.

- Once a TextGrid has Points (and ideally Ranges also) tiers labelled, the resynthesize Straight Line Approximation tools can be used to evaluate whether the Points labels capture the essence of the pitch contour (see § ?REF?).
- When using this tool, be on the lookout for f0-tracking errors and use comma override values to estimate missing or incorrect f0 values (see § ?REF?).
- Running this tool can be done from within an Editor window or from the Objects window (when one Sound/TextGrid pair is selected).
- When this tool is called from the Objects window, the labeller can choose one of three options



- * The Quick+Quiet option simply creates the resynthesis, according to default settings.
 - * The Quick option uses default settings as well, and also plays the original and resulting resynthesized versions back-to-back.
 - * The Advanced option gives the user many settings to adjust, and plays the original and resulting resynthesized versions back-to-back.
- Note that whenever the tool is used, a resulting Manipulation objects and a new Sound object will appear in the Objects window:



You can use these files to (re-)play the original and the resynthesized Sound objects and/or to manually adjust the f0 turning points in the Manipulation object.

- Labellers are reminded that it is a best practice to iteratively run the SLA resynthesizer while adding/removing Points, to ensure that your labels include only and all necessary f0 turning points to felicitously capture the pitch contour.
- The “Draw Sound and TextGrid” commands produce figures for publication.
 - The default settings are to draw the spectrogram with an overlaid f0 contour above all the tiers of the TextGrid, using the settings described on page **?REF?**.
 - Running the advanced script opens a new window, where many settings can be manipulated, and other options such as one to allow the user to select which TextGrid tiers to include.
 - Running this tool can be done from the Objects window or an Editor window.
- You can export PoLaR-Related Values to formatted text for spreadsheet software and/or statistical analysis code.
 - The information that is exported includes:
 - * the labels themselves
 - * the timing of the labels (both start and end times, where appropriate)
 - * where the labels occur with respect to intervals on Words and Phones tiers,
 - * the labels of the Levels and Ranges tiers, where each label occurs
 - * the f0 values at/during each label
 - * the intensity values at/during the label
 - Running this tool can be done from the Objects window or an Editor window.
 - * It can be run on objects loaded into Praat (either from the Objects win-

dow or an Editor window): it outputs text that can be copy-pasted into a spreadsheet¹ or into a text file (and then saved as a .tsv [tab separated values] file).

- * It can be run on all .wav and .TextGrids in a directory (which must have exactly matching file names) by choosing “PoLaR: Extract PoLaR info from files in a directory” from the “New” menu in the Objects window: it outputs data directly into a .tsv file.
- * Any .tsv file can then be imported into spreadsheet software, fed as input to statistical analysis scripts, or submitted to machine learning algorithms.
- You can convert PoLaR labels to pseudo-categorical (AM-style) labels.
 - One implementation of an algorithm to do this is in draft state, in the PoLaR plugin for Praat (see § [?REF?](#))
 - Note that, in its current implementation, there needs to be a Phones tier (perhaps previously automatically created by a forced aligner) in addition to Advanced labels on the Points tier

There are other planned (i.e., not yet available) functionalities in the PoLaR plugin, which include the ability to automatically estimate (a draft set of) pitch turning points to be used as a starting point for a human labeller creating Points Tier labels.

1.1.3 General Guide to names/functionality of scripts

Most users of the PoLaR plugin for Praat will not need to explore the .praat files that are the basis of the functionality described here and elsewhere. However, some users may want to delve into these files (e.g., to troubleshoot, better understand the algorithm, or simply run the scripts directly instead of through Praat).

The scripts have been named according to the following conventions:

- CORE scripts: contain the core functionality
- DIR scripts: perform the core action on all files in a Directory/Folder
- SLA scripts: Straight Line Approximation scripts with various options
- QUICK scripts: use the default settings (as defined by the QUICK-SETTINGS file).
- TGE scripts: Text Grid Editor scripts. Praat makes a distinction between Editor scripts and standalone scripts. All others are standalone.
- TSV scripts: extract information to TSV (tab separated values) files

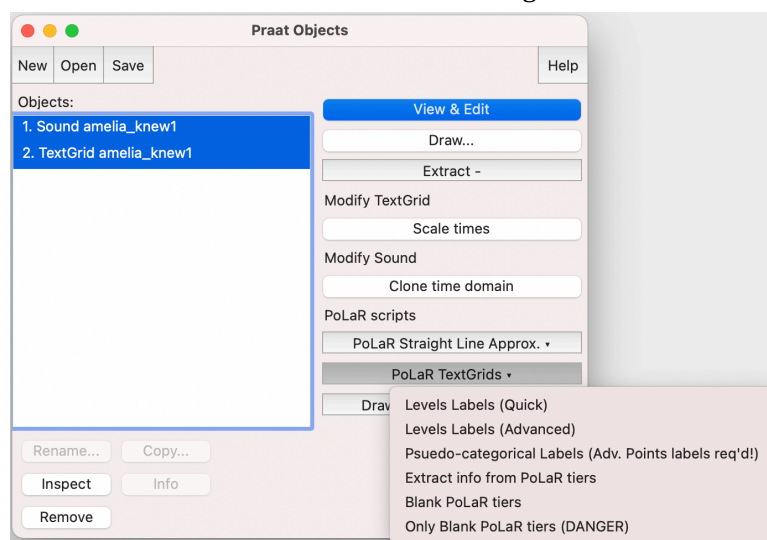
1.2 Workflow with PoLaR in Praat

1.2.1 Preparing TextGrids

- Create initial Textgrids with Words and Phones tiers labelled for each Sound file.

¹If copy-pasting into an Excel spreadsheet doesn't separate values across columns, try using “Paste Special” and choose “Unicode Text”.

- Labellers are strongly recommended to use a forced aligner such as the Montreal Forced Aligner (MFA; McAuliffe et al. 2019) to do this.
 - To use the MFA, first transcribe the speech in XYZ.wav in English orthography, and save that as a (plain text) text file as XYZ.lab (i.e., as a .lab file with a name that matches the .wav file).
 - Put all pairs of .wav and .lab files for a given speaker into one folder, and run Montreal Forced Aligner (see instructions here).
 - * (N.B.: the MFA assumes all paired .wav/.lab files that are in the same directory are spoken by the same individual. If there are multiple speakers, place the .wav/.lab files into separate subfolders named for each speaker, to get optimal alignments.)
- After creating the TextGrid(s) with Words and Phones tiers, open and select all of them in the Praat Objects window.
- From the PoLaR TextGrids menu on the right hand side, choose “Blank PoLaR Tiers”



- This adds the PoLaR tiers to the TextGrid files, leaving them ready to label.
- (*Don't forget to save all TextGrid objects as .TextGrid files!*)

1.2.2 PoLaR Labelling

- Begin PoLaR labelling.
 - Most tiers can be labelled in any order (though the Levels tier should be labelled after the Points and Ranges tiers).
 - Each tier can also be labelled by different individuals, allowing for a “divide and conquer” approach.
 - It may be easiest to first add Basic PoLaR labels, and then to revise with Advanced PoLaR labels. (It may make sense for the Basic labellers and the Advanced labellers to be different individuals.)
- Label the PrStr tier.
 - Since this is done according to intuitions of human listeners, there are no auto-

mated scripts to use here.

- As alluded to in the guidelines, these labels can also be generated on the basis of methodology like those used in RPT tasks (cf. Cole et al. 2014, 2017).
- Label the Ranges tier.
 - Since this is done according to intuitions of human listeners, there are no automated scripts to use here.
 - As alluded to in the guidelines, Basic labels may attend to the min/max attested values, but more Advanced labellers ought to attend to more granular intuitions (cf. § ?REF?).
- Label the Points tier.
 - This tier encodes intuitions as well, but there is a tool to help labellers “check their work”: the Straight Line Approximation resynthesis tool (cf. § ?REF?).
 - Note this tool works best if Ranges tiers are already labelled (but can work without such labels as well).
- Label the Levels tier.
 - This tier ought to be labelled using the Levels Label tool.
 - Labellers can use Levels labels to “check their work” with the Ranges tier (e.g., if the pitch at a particular time sounds high to a listener, but has a Levels label of 2 or 3, Ranges labels ought to be revisited; see § ?REF?).
- We encourage a practice whereby each file is labelled by multiple labellers, who then discuss labels together after completing their tasks.
 - PoLaR labels are specifically designed to facilitate discussions about interlabeller differences; these discussions often result in more fine-tuned labels for a recording, which may provide better analytic results.

1.2.3 Post-labelling analysis tools

- After labelling is complete, there are several analytic approaches one could take; we name a few below.
- Impressionistically review the labels.
 - Useful for identifying patterns in prominence/boundary locations, Levels sequences, Range changes, etc.
- Add labels from other labelling systems (e.g., ToBI, IViE, etc.).
 - Useful for investigating how categorical labels are realized across utterances / speakers / contexts / etc.
 - As mentioned in § ?REF?, the PoLaR plugin for Praat can create pseudo-categorical labels which may be useful in identifying intonational categories from the bottom-up.
- Submit PoLaR labels to statistical analysis/computational modelling.
 - To do this, the researchers are encouraged to make use of the Extract Info from PoLaR tiers tool, to distill PoLaR labels to various vectors.

- (Note this tool can be run on an entire directory of files, through the “PoLaR: Extract PoLaR info from files in a directory” option in the “New” menu of the Praat Objects window.)

Bibliography

- Ahn, Byron & Veilleux, Nanette & Shattuck-Hufnagel, Stefanie & Brugos, Alejna. 2021. PoLaR annotation guidelines (version 1.0). Available at <https://osf.io/usbx5/>. doi:10.17605/OSF.IO/USBX5
- Cole, Jennifer & Mahrt, Timothy & Hualde, José I. 2014. Listening for sound, listening for meaning: Task effects on prosodic transcription. *Proceedings of Speech Prosody* 7 859–863.
- Cole, Jennifer & Mahrt, Timothy & Roy, Joseph. 2017. Crowd-sourcing prosodic annotation. *Computer Speech and Language* 45. 300–325. doi:10.1016/j.csl.2017.02.008
- McAuliffe, Michael & Socolof, Michaela & Mihuc, Sarah & Wagner, Michael & Sonderegger, Morgan. 2019. Montreal forced aligner [computer program] (version 1.0.1). Retrieved from <http://montrealcorpus-tools.github.io/Montreal-Forced-Aligner/>. doi:<http://doi.org/10.5281/zenodo.2630943>