NCHLT Praat plugin

Jörg Mayer
jmayer@lingphon.net

December 2018

Contents

1 Description

This plugin enables Praat to search in the orthographic transcriptions of the NCHLT Speech Corpus and open the audio files of corresponding search results. The NCHLT Speech Corpus contains orthographically transcribed broadband speech corpora for all of South Africa's eleven official languages. Each language corpus consists of a large training suite (trn) and a smaller test suite (tst). Launching this plugin, you first select one of your installed languages, pick one of the suites (or both), and specify a search pattern. The XML orthographic transcription is searched for matches of the pattern. You can further refine the search results using speaker attributes (gender, age, location). Search results are

available as a Praat table object (including orthographic transcription, speaker ID, age, gender, and location). Corresponding audio files can be viewed and analyzed.

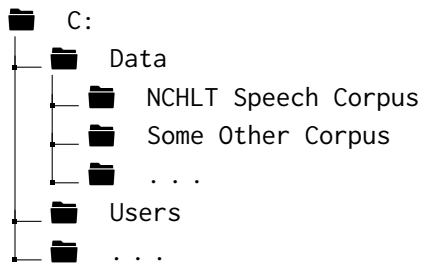
2 Requirements

- Computer running Windows, macOS, or Linux
- NCHLT Speech Corpus (at least one language)
- Praat 5.4.x or newer

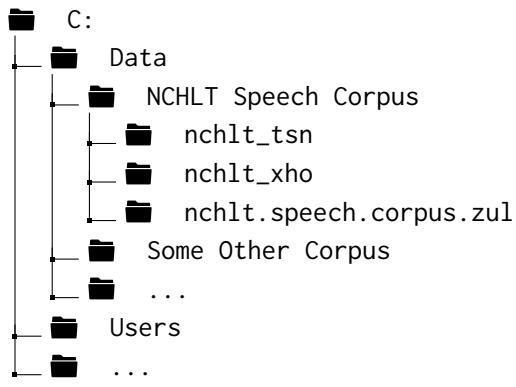
2.1 NCHLT Speech Corpus

At least one language of the NCHLT Speech Corpus has to be available on your computer. The corpus may be obtained online from the South African Centre for Digital Language Resources (SADiLaR): www.sadilar.org/

Create a new folder somewhere on your computer. You are free to choose any name and location for this folder but be sure to remember your choice, we'll need this information later. This folder is called the ***base directory*** of the NCHLT Speech Corpus in this manual. As an example, I called the base directory *NCHLT Speech Corpus* and created it within a Data folder, which contains other corpora as well:

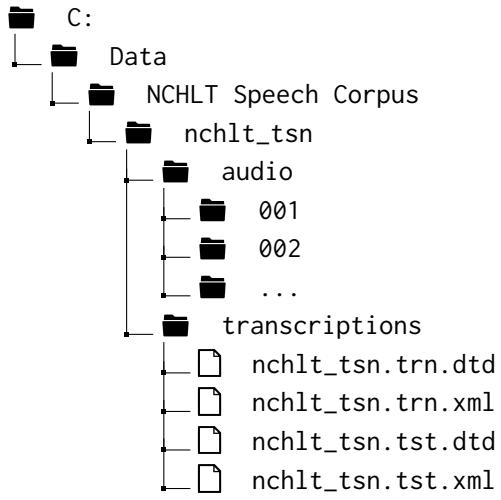


Now move the downloaded language archive(s) to the base directory and extract the data. There should be a folder for each downloaded language inside the base directory:



In the example above, 3 languages are available: Setswana (`nchlt_tsn`), Xhosa (`nchlt_xho`), and Zulu (`nchlt.speech.corpus.zul`). You'll notice two different folder naming patterns: "`nchlt_`" plus 3-letter language code (e.g. `nchlt_tsn`) which was used by archives obtained from the former provider, *South African Language Resource Management Agency*, and "`nchlt.speech.corpus.`" plus 3-letter language code (e.g. `nchlt.speech.corpus.zul`) which is used by archives from the current provider, *SADiLaR*. The plugin supports both naming patterns even if they are mixed as in the example above (but be careful, you should only have one folder per language!).

If everything went well each language folder contains two subfolders, *audio* and *transcriptions*. Within *audio* you find subfolders for each speaker which contain the actual audio files. Within *transcriptions* you find two XML transcription files (training and test) and associated document type definitions:



If you later decide to add languages just extract the new languages to the corpus base directory as decribed above. The plugin will recognize any new language.

2.2 Praat

The plugin requires Praat 5.4.x or newer. Download Praat (available for Windows, OS X, and Linux) from: www.praat.org

You don't need to 'install' Praat. Just open the downloaded archive (Windows: zip, OS X: dmg, Linux: tar.gz) and extract the Praat binary to any location on your disk. Launch Praat at least once (double click the Praat binary) to test whether it works well. In addition, this will create a special folder that we'll need in the next step.

3 Installation

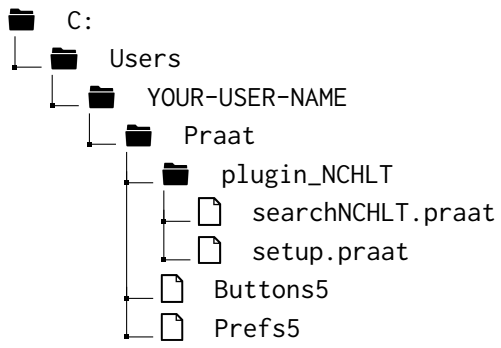
Quit Praat and get the latest plugin version:

github.com/jouml/NCHLT-Praat-Plugin/archive/master.zip

Extract the downloaded archive (`NCHLT-Praat-Plugin-master.zip`) and change into the newly created folder *NCHLT-Praat-Plugin-master*. Copy the included subfolder *plugin_NCHLT* to your Praat preferences folder. The Praat preferences folder is a special folder which is created automatically when Praat is launched the first time (therefore be sure to run Praat at least once before you attempt to install the plugin). The location of this folder depends on your operating system:

- Windows: `C:\Users\YOUR-USER-NAME\Praat\`
- macOS: `/Users/YOUR-USER-NAME/Library/Preferences/Praat Prefs/`
- Linux: `/home/YOUR-USER-NAME/.praat-dir/`

You should end up with a subfolder called *plugin_NCHLT* in your Praat preferences folder. Outcome on a Windows machine:



`C:\Users\YOUR-USER-NAME\Praat`	Your Praat preferences folder
`plugin_NCHLT`	The plugin subfolder
`searchNCHLT.praat`	Main Praat script that does all the work
`setup.praat`	Plugin setup; adds a menu entry to Praat's Open menu and links the main script to it
`Buttons5 & Prefs5`	Standard Praat preference files

That's it. Next time, when Praat is launched you find a new entry at the bottom of the Open menu called NCHLT: Search corpus....

4 Usage

4.1 First Usage: Corpus Base Directory

Launch Praat then select Open > NCHLT: Search corpus... . The very first time you use the plugin it will ask for the location of the corpus base directory. For this purpose, the standard folder selection dialog of your operation system pops up and you can navigate to the corpus base directory to select it. The corpus base directory is the **parent folder which contains all the language subfolders** (see section 2.1). In the example from section 2.1 this would be:

C:\Data\NCHLT Speech Corpus

Be careful to select the encompassing parent folder and not one of the language subfolders, like e.g. *nchlt_tsn*.

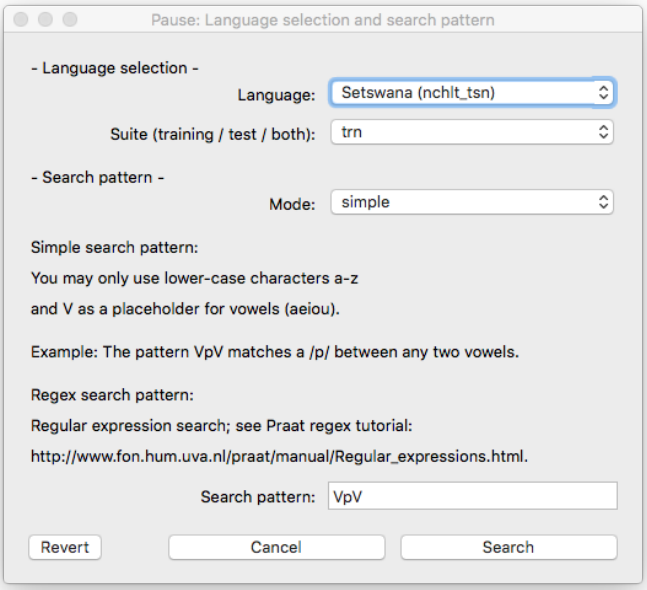
After you selected the base directory the location is saved to a settings file, so you don't need to repeat this step in the future. The settings file is located in the plugin folder and is called *CorpusDir*. On Windows for example:

C:\Users\YOUR-USER-NAME\Praat\plugin_NCHLT\CorpusDir

If you later decide to move your corpus to a new location or if you encounter any problems with the plugin finding the corpus just delete that file. The plugin will ask for the base directory the next time you run it.

4.2 Language Selection and Search Pattern

After launching the plugin, the follwing dialog box appears:

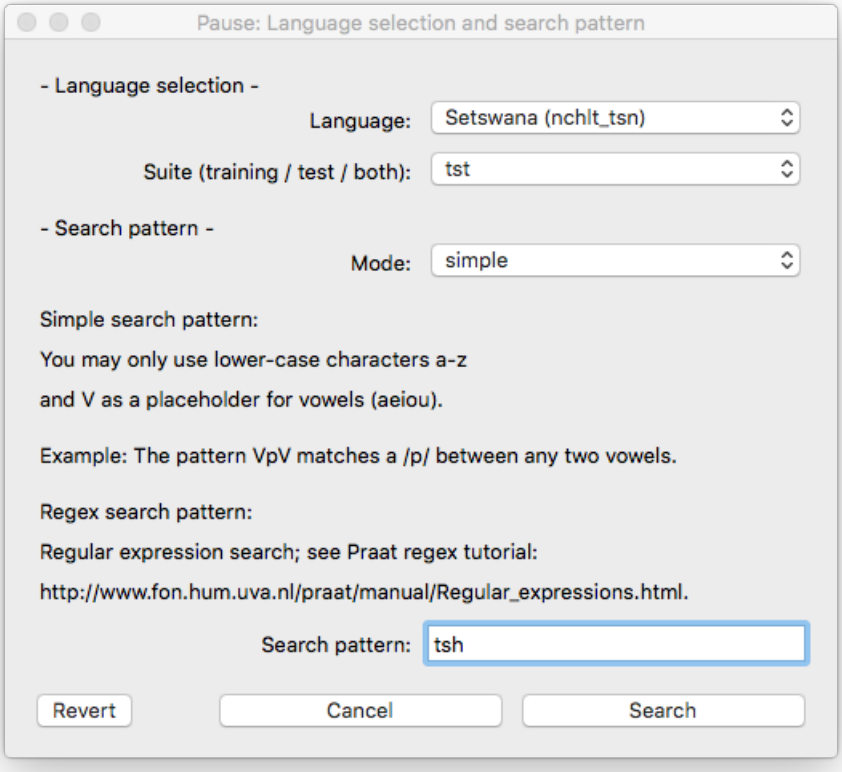


First, select one of the installed languages. If you added a new language to the corpus since the last plugin run this language is recognized automatically and appears in the drop-down menu. Next, select a suite. You can search in the (large) training suite, in the (small) test suite, or in both suites at the same time. Finally, you can specify a search pattern. Two modes are provided for the formulation of the search pattern: *simple* and *regex* (regular expression).

Simple search pattern

If the *simple* mode is active, most of your pattern is interpreted literally, i.e. the search results will consist of items with orthographic transcriptions that contain exactly the specified sequence of characters. Since all NCHLT transcriptions (as far as I know) are exclusively composed of lower-case characters, only lower-case characters are valid in simple search patterns. The only exception to this rule is the upper-case **V** character. **V** is a metacharacter, i.e. it is not interpreted literally but as a placeholder for any one vowel (*a*, *e*, *i*, *o*, or *u*).

For example, if you are interested in the Setswana consonant cluster /tsh/ you might select Setswana test suite (smaller but faster) and the simple search pattern **tsh**:



This will produce search results like:

*sa dumalane le **tsh**wetso e*
*tlamela ka **tsh**edimosetso e e*
*e e bon**tsh**itsweng mo setifikeiting*
*dikopo tsotlhe di **tsh**wanetse go*
*a bo**tsh**abi a tsaya dingwaga*
…

To exclude items where *tsh* is followed by another consonant (e.g. *w*, see examples 1 and 4) use this search pattern instead: **tshV**, i.e. all items containing the literal sequence *tsh* followed by a vowel. Results:

*tlamela ka **tshe**dimosetso e e*
*e e bon**tshi**tsweng mo setifikeiting*
*a bo**tsha**bi a tsaya dingwaga*
*mo**tshe**gang*
*nang le phitlhelelo go **tshe**dimosetso*
…

Of course, you can also specify literal vowels: **phi** searches for the literal sequence *phi*. Or you can combine literal vowels and the vowel class symbol in one pattern: **phitV** searches for the literal sequence *phit* followed by any vowel.

Regular expression search pattern

Regular expression search patterns are more complex but also much more flexible than simple patterns. Whereas simple patterns are aware of only one metacharacter, regex patterns make extensive use of metacharacters and other special symbols. Furthermaore, in order to compose a well-formed regular expression of literal characters, metacharacters, and special symbols some flavor of a regex syntax must be respected. The Praat flavor of regex syntax as well as the inventory and semantics of metacharacters and special symbols is described in the Praat manual:

www.fon.hum.uva.nl/praat/manual/Regular_expressions.html

Examples

In simple search mode, **V** is a placeholder for a set of characters, namely all possible vowels (*a, e, i, o, u*). With regular expression, you can define your own character sets. To search for *tsh* followed by a high front vowel, use this regex pattern:

tsh[ie]+

tsh are literal characters matching exactly the sequence *tsh*. Square brackets specify a set of characters matching either one member of the set. Hence, **[ie]** matches either i or e. **+** is a quantifier meaning one or more occurences of the

preceding element. The complete pattern means: *tsh* followed by at least one of either *i* or *e*. This finds:

> *tlamela ka* ***tshe****dimosetso e e*
> *e e bon****tshi****tsweng mo setifikeiting*
> *mo****tshe****gang*
> *nang le phitlhelelo go* ***tshe****dimosetso*
> …

but not:

> *a bo****tsha****bi a tsaya dingwaga*
> *le monon****tsha*** *o o kwadisitsweng*
> *phemithi ya mokopi wa bo****tsha****belo …*

(BTW: The regex pattern **tsh[aeiou]** behaves exactly like the simple pattern **tshV**.)

A range of characters in a set can be abbreviated: **[a-f]** = **[abcdef]**; all lower-case characters: **[a-z]**; all digits: **[0-9]** etc.

The following regex pattern narrows down the previous search to word initial occurrences of *tshi* or *tshe*:

\stsh[ie]+

\s is an escape sequence meaning *whitespace*, matching a space, a tabstop, beginning of line etc. This finds:

> *tlamela ka* ***tshe****dimosetso e e*
> *nang le phitlhelelo go* ***tshe****dimosetso*
> *go naya* ***tshe****dimosetso malebana le*
> *sa lenyalo sa* ***tshi****mologo kgotsa*
> …

but not:

> *e e bon****tshi****tsweng mo setifikeiting*
> *mo****tshe****gang*
> …

To restrict the search to phrase initial occurrences of *tshi* or *tshe*, use one of the special anchor symbols:

^tsh[ie]+

The anchor **^** matches the beginning of a line (**$** matches the end of a line). This finds only word initial occurrences at the beginning of a transcription:

> ***tshe****dimosetso e e umakiwang mo*
> ***tshe****dimosetso nngwe le nngwe e*
> ***tshe****dimosetso e e leng teng*
> …

The last example finds occurrences of *tshi* or *tshe* in the first word of the phrase but not necessarily in the word initial position:

^\w*tsh[ie]+

\w is an escape sequence matching 'word' characters, i.e. any letter, digit, or underscore (this is equivalent to the character set **[a-zA-Z0-9_]**). The asterisk ***** is a quantifier meaning zero or more occurrences of the preceding element. So, the pattern translates to: Search at the beginning of lines looking for any number (incl. zero) of word characters (i.e. letters, digits, or underscore, but not whitespace) followed by *tshi* or *tshe*. This finds:

> *bo**tshe**lo jo bo botoka go*
> *mo**tshe**gang*
> ***tshe**dimosetso e e umakiwang mo*
> ***tshe**lang ka bolwetsi jo bo*
> *kgotla**tshe**kelo o ka solofela gore*
> *bon**tshi**tsweng mo setifikeiting sa irp*
> …

As mentioned above, regular expression search is complex and error prone. A faulty regex pattern may generate results that represent only a fraction of the data your are interested in. For this reason, be sure to carefully read the regular expression chapter of the Praat manual before you start using the regex mode. Additionally, you'll find many useful online tutorials teaching regular expressions.

When you are finished with language selection and the search pattern, press the Search button. The plugin loads and browses through the specified transcription, which can take some time (especially, if the large training suite was selected)—please be patient.

4.3 Managing Results

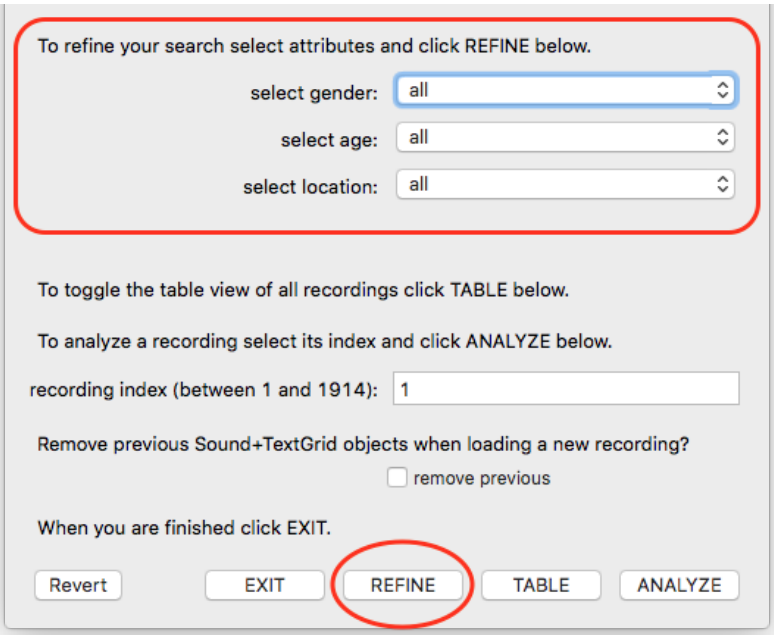
After the search is finished, Praat info notifies you about the number of evaluated recordings and the elapsed time. You can close that window if you wish. If the search yields any results a new window appears: the results dialog; if nothing was found Praat info will say so and you can try again with a different search pattern.

Summary

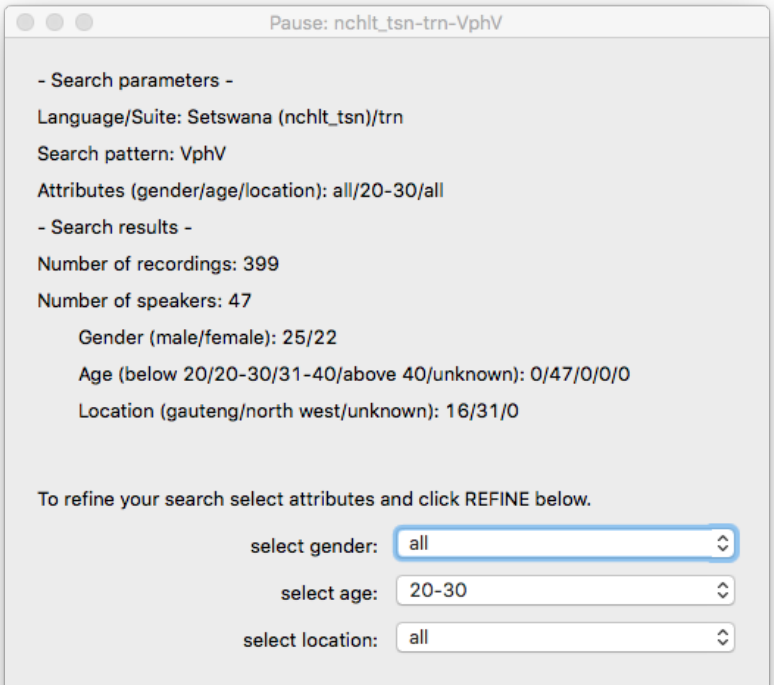
The first section of the results dialog supplies information about search parameters and results. In the example above, the Setswana training suite (line 2) was searched for **VphV**, i.e. *ph* between two vowels (line 3). With no filters active (line 4: gender=all, age=all, location=all). This yields 1914 recordings (line 6) from 198 speakers (line 7). Lines 8 to 10 depict speaker characteristics in terms of gender, age range, and location: 104 of 198 speakers are female (line 8). As most of the time, speakers with unknown age make up the largest group (115 in the example), while speakers older than 40 are rare (5 in the example) (line 9). 140 of 198 speakers are from the Gauteng region (line 10).

Filters

The second section of the results dialog lets you refine your search by specifying filter parameters. At the beginning, all parameters are unspecified (gender=all, age=all, location=all).



To restrict the intitial search results to young adult speakers, select the approprite age range (e.g. 20-30) from the drop down menu and press the REFINE button at the bottom of the dialog. An updated results dialog will appear, summarizing the filtered results.



The active filter (age range 20-30) is indicated in line 4. 399 recordings from 47 speakers remain from the intitial search results (lines 6 & 7) and all speakers are between 20 and 30 years old (line 9).

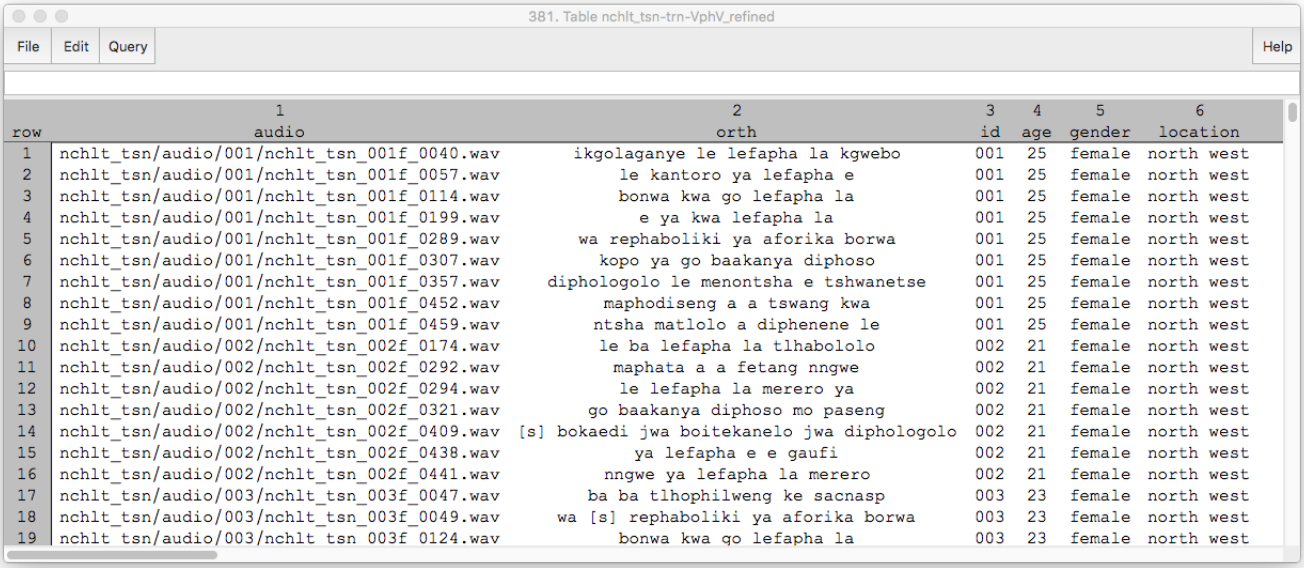
If desired, add additional filters: Select specific gender and/or specific location and press the REFINE button again. To reset a filter just select *all* from the drop-down menu and press REFINE.

As soon as you are satisfied with your filter settings have a closer look at the results.

Data exploration

The third section of the results dialog is devoted to data exploration. With the TABLE button at the bottom of the dialog the table view is toogled on and off. Pressed the first time, a new window appears presenting symbolic information about the results in a table. Pressed again, the table disappears.

For each found item, the table contains an index, (column "row") the path to the associated sound file (column "audio"), the orthographic transcription (column "orth"), and speaker characteristics (id, gender, age, location). Browse through the table to get an overview of the data and/or to discover interesting items.



row	1 audio	2 orth	3 id	4 age	5 gender	6 location
1	nchlt_tsn/audio/001/nchlt_tsn_001f_0040.wav	ikgolaganye le lefapha la kgwebo	001	25	female	north west
2	nchlt_tsn/audio/001/nchlt_tsn_001f_0057.wav	le kantoro ya lefapha e	001	25	female	north west
3	nchlt_tsn/audio/001/nchlt_tsn_001f_0114.wav	bonwa kwa go lefapha la	001	25	female	north west
4	nchlt_tsn/audio/001/nchlt_tsn_001f_0199.wav	e ya kwa lefapha la	001	25	female	north west
5	nchlt_tsn/audio/001/nchlt_tsn_001f_0289.wav	wa rephaboliki ya aforika borwa	001	25	female	north west
6	nchlt_tsn/audio/001/nchlt_tsn_001f_0307.wav	kopo ya go baakanya diphoso	001	25	female	north west
7	nchlt_tsn/audio/001/nchlt_tsn_001f_0357.wav	diphologolo le menontsha e tshwanetse	001	25	female	north west
8	nchlt_tsn/audio/001/nchlt_tsn_001f_0452.wav	maphodiseng a a tswang kwa	001	25	female	north west
9	nchlt_tsn/audio/001/nchlt_tsn_001f_0459.wav	ntsha matlolo a diphenene le	001	25	female	north west
10	nchlt_tsn/audio/002/nchlt_tsn_002f_0174.wav	le ba lefapha la tlhabololo	002	21	female	north west
11	nchlt_tsn/audio/002/nchlt_tsn_002f_0292.wav	maphata a a fetang nngwe	002	21	female	north west
12	nchlt_tsn/audio/002/nchlt_tsn_002f_0294.wav	le lefapha la merero ya	002	21	female	north west
13	nchlt_tsn/audio/002/nchlt_tsn_002f_0321.wav	go baakanya diphoso mo paseng	002	21	female	north west
14	nchlt_tsn/audio/002/nchlt_tsn_002f_0409.wav	[s] bokaedi jwa boitekanelo jwa diphologolo	002	21	female	north west
15	nchlt_tsn/audio/002/nchlt_tsn_002f_0438.wav	ya lefapha e e gaufi	002	21	female	north west
16	nchlt_tsn/audio/002/nchlt_tsn_002f_0441.wav	nngwe ya lefapha la merero	002	21	female	north west
17	nchlt_tsn/audio/003/nchlt_tsn_003f_0047.wav	ba ba tlhophilweng ke sacnasp	003	23	female	north west
18	nchlt_tsn/audio/003/nchlt_tsn_003f_0049.wav	wa [s] rephaboliki ya aforika borwa	003	23	female	north west
19	nchlt_tsn/audio/003/nchlt_tsn_003f_0124.wav	bonwa kwa go lefapha la	003	23	female	north west

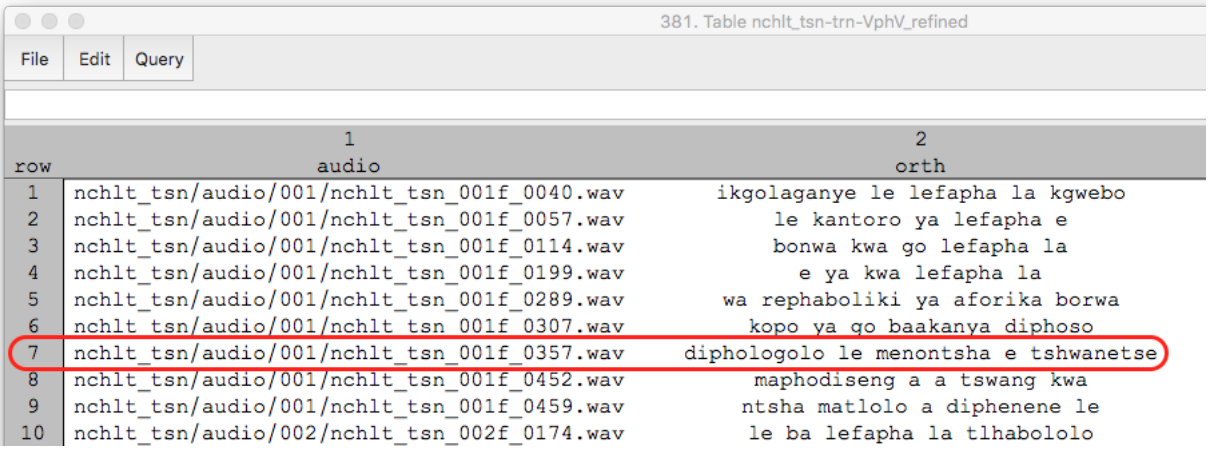
To open a recording for acoustic analysis, specify the recording's index in the *recording index* field of the results dialog and press the ANALYZE button. A Praat TextGrid editor window appears with the waveform and the orthographic transcription in a TextGrid tier. When finished just close the editor window.

The *recording index* field is pre-filled with index 1 and automatically increased by 1 everytime you press the ANALYZE button. This mechanism allows you to analyze each recording one by one by just pressing ANALYZE repeatedly.

If you don't want to analyze all recordings but only some specific ones that you have discovered in the table, type the index of a recording into the *recording*

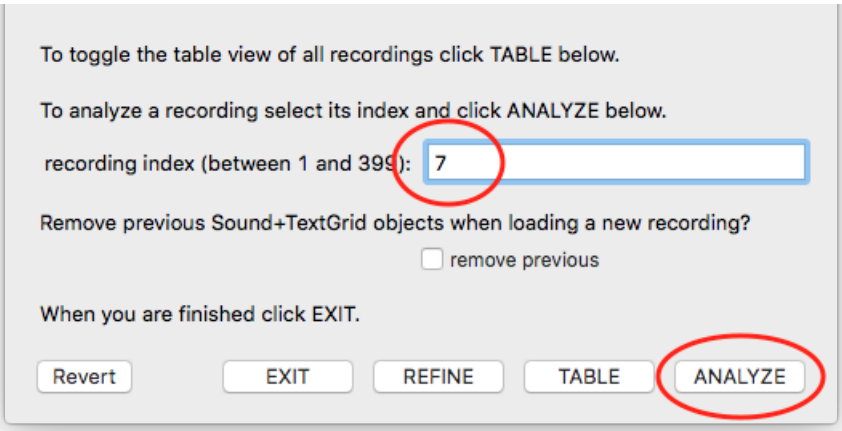
index field of the results dialog and press the ANALYZE button. You find the recordings' indices in the first column of the table.

For instance, if you are interested in the highlighted item below:



row	1 audio	2 orth
1	nchlt_tsn/audio/001/nchlt_tsn_001f_0040.wav	ikgolaganye le lefapha la kgwebo
2	nchlt_tsn/audio/001/nchlt_tsn_001f_0057.wav	le kantoro ya lefapha e
3	nchlt_tsn/audio/001/nchlt_tsn_001f_0114.wav	bonwa kwa go lefapha la
4	nchlt_tsn/audio/001/nchlt_tsn_001f_0199.wav	e ya kwa lefapha la
5	nchlt_tsn/audio/001/nchlt_tsn_001f_0289.wav	wa rephaboliki ya aforika borwa
6	nchlt_tsn/audio/001/nchlt_tsn_001f_0307.wav	kopo ya go baakanya diphoso
7	nchlt_tsn/audio/001/nchlt_tsn_001f_0357.wav	diphologolo le menontsha e tshwanetse
8	nchlt_tsn/audio/001/nchlt_tsn_001f_0452.wav	maphodiseng a a tswang kwa
9	nchlt_tsn/audio/001/nchlt_tsn_001f_0459.wav	ntsha matlolo a diphenene le
10	nchlt_tsn/audio/002/nchlt_tsn_002f_0174.wav	le ba lefapha la tlhabololo

Find the index of the recording in the first dark gray column (called "row"): it's index 7. Type 7 into the *recording index* field and press ANALYZE.



When finished with the analysis, close the editor window and continue with the next recording:

find index — type index — press ANALYZE

Options and Exit

While the results dialog is open, you can adjust filters at any time. To return to the original results (based on your intitial language selection and search pattern)

set all filters to *all* and press REFINE. Toogle the table view on and off as often as you like.

If you analyze acoustic data consider the last option in the results dialog: *remove previous*. Background: Every time you press ANALYZE, a sound file is opened by Praat (as Sound object) and a TextGrid is generated containing the orthographic transcription (as TextGrid object). Both objects reside in Praat's main window called *Praat Objects*. When you close the editor after the analysis, these objects will remain in the objects list, i.e. they are still available to Praat and consume your computer's RAM. They'll vanish not until you quit Praat. This is the default behavior with *remove previous* unchecked. If the option is checked, as soon as you press ANALYZE again, both previous objects (Sound and TextGrid) are removed before the two new objects are generated. This results in a clean objects list and better RAM hygiene. However, previous objects are removed without warning. That's a serious disadvantage if you plan to edit the TextGrids, because all changes are lost when the TextGrid object is removed (unless you save the object to a file before (!) analyzing the next recording).

Recommendation: If you don't intend to edit TextGrids during your analysis you can safely check this option. If you intend to edit TextGrids (adjust transcriptions, add tiers for annotations etc.) you better keep the option unchecked. You have to deal with manually saving your modified TextGrids anyway and I would recommend to do this at the latest before you close the editor window (which would allow you to check the option). But with the option unchecked you preserve the opportunity to catch up on saving later.

After processing all results, press EXIT at the bottom of the results dialog. The dialog window disappears, table objects of the finished search cycle are removed, and you can start over with a new search cycle or quit Praat.

4.4 Conclusion: The Search Cycle

- launch the plugin: Open » NCHLT: Search corpus...
- first dialog box appears
- select language and specify search pattern; press Search
- *Praat Info* appears with feedback on progress
- second dialog box (results dialog) appears
  - apply filters at any time; select options from the drop-down menus and press REFINE
  - toogle table view at any time; press TABLE
  - view, analyze, and listen to audio; [type index] and press ANALYZE
- finish cycle; press EXIT

4.5 Exploit Results

Search and filtering results are available as Praat table objects while a search cycle is active. The plugin's TABLE button allows you to view these table objects, but Praat provides many more functions to explore and analyse table objects (see dynamic menu buttons in *Praat Objects*). Furthermore, it is possible to export table objects to comma-separated files (csv format). This allows you to process search results outside Praat.

You can find the table objects in the objects list of Praat's main window. They are named according to the following schema:

nchlt_*language-suite-searchpattern*

For instance, if you searched for **VphV** in the Setswana training suite the table object is called:

nchlt_tsn-trn-VphV

(Some symbols that may be part of the search pattern—especially in regex mode—are substituted with underscores in the context of object naming.)

Each search cycle generates two table objects. The first object—called like described above—contains the original unfiltered results and is not modified during a search cycle. The second object is identical to the first object initially. As soon as the first filter is applied (REFINE is pressed), the suffix _refined is added to the object name and it contains only the filtered results. Following the above example, the second object is called:

nchlt_tsn-trn-VphV_refined

after application of a filter. If you need to process results outside Praat be sure which table to choose (filtered or unfiltered results), select a table object accordingly and save it:

Save > Save as comma-separated file...

You have to do this *before* you press EXIT in the results dialog because at the end of a search cycle both table objects are removed.

5 Help

If you find a bug or need help please send me an email:

jmayer@lingphon.net

6 License