CHILDES & CLAN Workbook

Lisa Pearl

March 14, 2011

Contents

1 Introducing CHILDES

1.1 Background & Reading

The main website *http://childes.psy.cmu.edu/*

CHILDES stands for **CHI**ld **L**anguage **D**ata **E**xchange **S**ystem, and is one of the most useful freely available sources of empirical data on child language input and output. As a computational modeler, you can often find the data you need to ground your model appropriately. Check out *http://childes.psy.cmu.edu/intro/* for a more thorough introduction to how awesome CHILDES is. Some highlights:

- Browsable transcripts:
  http://childes.psy.cmu.edu/browser/
- Downloadable transcripts:
  http://childes.psy.cmu.edu/data/local.html
- Downloadable audio & video:
  http://childes.psy.cmu.edu/media/
- Downloadable datasets that have been phonetically transcribed:
  http://childes.psy.cmu.edu/derived/

Downloading corpora is very straightforward - simply follow the link to the appropriate corpus that you're interested in, and click on it. To get an overview of the different datasets available (including language, age of children, number of children, etc.), check out *http://childes.psy.cmu.edu/manuals/*. Take a moment now to browse through some of the available corpora.

1.2 Exercises

1. Name at least five languages that CHILDES provides child language data for.
2. Consider the American English data available, as described in the American English manual here: *http://childes.psy.cmu.edu/manuals/*. Name three corpora that include data directed at children between the ages of 2 and 4 years old. Make sure to identify what ages each corpus includes.
3. Identify one American English corpus that includes both transcript data (look here: *http://childes.psy.cmu.edu/data/Eng-USA/*) and audio & video data (look here: *http://childes.psy.cmu.edu/media/*).

4. Do any of the derived corpora listed at *http://childes.psy.cmu.edu/derived/* have audio & video available for them as well? If so, name them.

All of these corpora can be wonderfully useful. But how do you actually find and analyze the data that you're looking for? Fortunately, CHILDES comes with a handy tool called CLAN.

2 CLAN

2.1 Background

CLAN stands for **C**omputerized **L**anguage **AN**alysis, and is a freely available tool provided through the CHILDES project. *From the CLAN manual*: "It is a program that is designed specifically to analyze data transcribed in the format of the Child Language Data Exchange System (CHILDES). Leonid Spektor at Carnegie Mellon University wrote CLAN and continues to develop it. The current version uses a graphic user interface and runs on both Macintosh and Windows machines. Earlier versions also ran on DOS and Unix without a graphic user interface. CLAN allows you to perform a large number of automatic analyses on transcript data. The analyses include frequency counts, word searches, co-occurrence analyses, MLU {*Mean Length of Utterance*} counts, interactional analyses, text changes, and morphosyntactic analysis."

2.2 Install CLAN on your computer & get the latest CLAN manual

The website containing downloadable install programs and source code for CLAN, along with a link to the current CLAN manual:

http://childes.psy.cmu.edu/clan/

Read section 1.2 "Installing CLAN" in the clan manual for more detailed installation instructions on how to install CLAN for your particular computer's operating system.

Direct link to current manual: *http://childes.psy.cmu.edu/manuals/clan.pdf*

In general, it pays to keep the manual handy as a reference, though the CLAN program itself also has ways for you to get help directly.

2.3 A quick CLAN tutorial

To familiarize yourself with the basic layout of the CLAN program, work through sections 2.1 and 2.2 of the CLAN manual (pp.9-15).

Useful note: Typing a command by itself with no arguments will cause CLAN to produce the list of possible arguments that command can take. For example, type `freq` in the command window. You should see the following appear on the command window:

```
FREQ creates a frequency word count
    Usage: freq [cN oN dN fS k pF rN re sS tS u xN yN zN ] filename(s)
    +c : find capitalized words only
```

+c1: find words with upper case letters in the middle only
+o : sort output by descending frequency
+o1: sort output by reverse concordance
+o2: sort output by reverse concordance of first word;
preserve the whole line
+d : outputs all selected words, corresponding frequencies, and line numbers
+d1: outputs word with no frequency information. in KWAL or COMBO format
+d2: sends output to a file for STATFREQ. Must include speaker specifications
+d3: sends statistics only to STATFREQ. Must include speaker specifications
+d4: outputs only type/token information
+d5: outputs all selected words, including the ones with 0 frequency count
+fS: send output to file (program will derive filename)
-f : send output to the screen or pipe
+k : treat upper and lower case as different
+pF: define punctuation set according to file F
+rN: if N = 1 then "get(s)" goes to "gets", 2- "get(s)", 3- "get"
4- recognize prosodic symbols in words, 5- no text replacement: [: *]
6- exclude repetitions: </>, <//>, <///>, </-> and </?>,
7- do not remove '/' and ':' characters
+re: run program recursively on all sub-directories.
+sS: search for word S or words in file @S in an input file (+s@ for more info).
-sS: exclude word S or words in file @S from a given input file (-s@ for more info)
+tS: include tier code S
-tS: exclude tier code S
+u : merge all specified files together.
-u : compute result for each turn separately.
+xN: include only utterances which are longer than N
-xN: include only utterances which are shorter than N
+y : work on TEXT format files one line at the time
+y1: work on TEXT format files one utterance at the time
+zN: compute statistics on a specified range of input data
File names can be "*.cha" or a file of list of names "@:filename"

You may not need many of these options, but some of them are very handy. Also, it's always helpful to remind yourself what options are available for a given command - there may be something you need to do that CLAN does automatically!

Useful note 2: Shell commands.
Check out section 7.1 (page 46 of the CLAN manual), which briefly discusses some basic unix-like shell commands that are useful to know, including these:

- **batch**: allows you to place a bunch of clan commands in a text file and then execute them all at once - extremely useful if you have to execute a large number of repetitive commands.
- **cd**: allows you to **c**hange the **d**irectory you're in
- **info**: displays commands that are available to be executed
- **dir**: displays the contents of the current **dir**ectory

2.3.1 Exercises

1. Download the Brown corpus transcripts from the American English corpus to a convenient directory on your computer. This should contain files for Adam, Eve, and Sarah. Change your working directory so it's in the Adam folder. Type the following command to get only the type/token frequency information from the child-directed speech of all the Adam files:

   freq +d4 -t*CHI +u *.cha

   This should give you the following frequency statistics about the Adam corpus:

   3797 Total number of different word types used

   118222 Total number of words (tokens)

   0.032 Type/Token ratio

   How would you modify this command so that you get only the type/token frequency information from the child-produced speech from the Adam corpus?

2. Now change your working directory to Brown (so the sub-folders Adam, Eve, and Sarah are visible). How would you use the +re command to get type/token frequency information only from all the child-directed speech in the Adam, Eve, and Sarah files? You should be able to do this with a single command. How many word tokens does this combined child-directed speech corpus include?

3. Now change your working directory to the Sarah sub-folder. Suppose you are only interested in the frequency information from the files sarah100.cha through sarah139.cha, specifically the child-directed speech and the child-produced speech, and you'd like to output this to a file named "sarahstats.txt". We're going to do this with a batch

file. The file sarahfreq.txt contains the following lines:

freq +d4 -t*CHI +u sarah10*.cha sarah11*.cha sarah12*.cha sarah13*.cha > sarah-stats.txt
freq +d4 +t*CHI +u sarah10*.cha sarah11*.cha sarah12*.cha sarah13*.cha >> sarah-stats.txt

Place this file into the Brown/Sarah sub-folder (which is your current working directory). Then, type

batch sarahfreq.txt

This command executes the commands in sarahfreq.txt. The first command calculates the frequency information for the child-directed speech in the selected sarah files and outputs it to a newly created file named sarahstats.txt. The second command calculates the frequency information for the child-produced speech in these files and appends it to the sarahstats.txt. The output file also contains a record of which commands were executed.
Now create a batch file that calculates the type/token frequency information for the child-directed and child-produced speech in all the Eve files where Eve is under the age of 2. Note: This will require you to look at the age information at the beginning of the files to see how old Eve is during each session. The files are in chronological order.

2.4 A closer look at freq

Section 8.8 in the CLAN manual discusses the freq command in much more detail. Read and work through this section (though you may feel free to skip sections 8.8.3, 8.8.9, and 8.8.10).

2.4.1 Exercises

1. Make the Brown/Sarah folder your working directory. Copy the demonstratives.txt file into that directory. To find out how often certain demonstrative words like "this" and "that" are said to Sarah, type

   freq +s@demonstratives.txt -t*CHI +u *.cha

   How would you modify this command to find how often the irregular past tense forms "ate", "drank", and "slept" appear in Sarah's input? (Hint: You probably

want to create your own text file containing these forms.)

2. Type the following line in to CLAN in order to identify how frequently reflexive pronouns appear in Sarah's input:

   freq +t%mor -t*CHI +s"pro:refl|*" +u sarah*.cha

   Change this command so it counts how often indefinite pronouns appear in Sarah's input. (Hint: The identifier for indefinite pronouns is pro:indef.) What if we're only interested in the word "one" when it's used as a pronoun - what command could we use then? What if we're interested in all uses of "one" (not just pronoun uses)? What other category classifications does "one" have besides indefinite pronoun (pro:indef)?

3. Now let's look at verb forms. Type the following command in to CLAN in order to identify how often Sarah hears any verbal usage of "eat" in her input:

   freq +t%mor -t*CHI +s"v|eat*" +u sarah*.cha

   How would we modify this command to get all past tense forms Sarah hears in her input? (Hint: Notice from the output of the command above that the signal on the %mor line for past tense is "&past".) What are the two most frequent past tense forms Sarah hears? (Hint: Use the +o option to sort the results by descending frequency.)

   Now let's look at how to identify any form of "eat" that Sarah hears in her input (check out the tables in section 10.5.2 of the CLAN manual to see other useful %mor line notation):

   freq +t%mor -t*CHI +s"@r-eat" +u sarah*.cha

   From this, we can see that "eat" is used as a noun (n), a participle (part), and a verb (v). Let's focus on participles. How would you count how many past participles total (identifier: part|*&perf) that Sarah hears in her input? Which two past participles appear most frequently?

2.5 Using kwal & combo.

kwal is short for **k**ey**w**ord **a**nd **l**ine, and can be used to pull out key words of interest and the line(s) they appear on in the corpus. Read and work through section 8.15 in the CLAN manual, starting on p.97 (though feel free to skip the section on tier selection.) combo is used to create boolean search strings. Read and work through sections 8.4.1 through 8.4.3 of the CLAN manual.

2.5.1 Exercises

1. Suppose we're interested in the indefinite pronoun "one", and want to look at Sarah's input. After changing your working directory to Brown/Sarah, type the following command to output the example usages of indefinite pronoun "one" that Sarah hears in her input to the file sarahproone.txt, and include the preceding ten utterances before each usage:

   kwal +t%mor -t*CHI +s"pro:indef|one" -w10 +d1 sarah*.cha > sarahproone.txt

   Now suppose that we want to specifically isolate the instances where the pronoun "one" is preceded by an adjective or determiner, which indicates that "one" is able to stand in for a grammatical category smaller than a noun phrase (unlike other pronouns like "'he"). Since we've already pulled out all the pronoun uses of "one", and if we don't feel like waiting for CLAN to re-process the entire Sarah corpus, we can use the following combo command to use the examples in sarahproone.txt and identify which usages have an adjective or determiner preceding them:

   combo +t%mor +s"(det*+adj*)^pro:indef|one" -w10 +d1 sarahproone.txt > sarahnon-NPone.txt

   Though we don't have to do this now, we could potentially use this output file to look for which usages of "one" are mentioned with a preceding determiner or adjective and which also have a potential referent that mentions a property (e.g., "Let's play the new game." "No - I want to play the old one." Here, "one" refers to "game", and not "new game".) We would do this by examining the utterances preceding each usage of "one" for the antecedent of "one". This could be useful if we're trying to determine how often the property mentioned in the antecedent of "one" is true of the referent of "one". In the parenthesized example, the property "new" is not true of the referent of "one". In general, this kind of kwal/combo output can be useful as input for many other kinds of data analysis.