STYLO R Script Mini-HOWTO

By Maciej Eder and Jan Rybicki

To cite the script in publications, use: Eder, M., Rybicki, J. (2011). Stylometry with R. In "Digital Humanities 2011: Conference Abstracts." Stanford University, Stanford, CA, pp. 308-11.

Contact with the authors:
Maciej Eder maciejeder@gmail.com
Jan Rybicki jkrybicki@gmail.com

What the script does:

It produces a most-frequent-word (MFW) list for the entire corpus; it then acquires the frequencies of the MFWs in the individual texts to create an initial input matrix of words (rows) by individual texts (columns), each cell containing a given word's frequency in a given text. The script then normalizes the frequencies; it selects words from stated frequency ranges for analysis (this is also saved to disk as "table_with_frequencies.txt"; it performs additional procedures (automatic deletion of personal pronouns and culling, see 2.5 below) to produce a final wordlist for analysis (this is saved to disk as "wordlist.txt". It then compares the results for individual texts, performing distance calculations and uses various statistical procedures (Cluster Analysis, Multidimensional Scaling or Principal Components Analysis) to produce graphs of distances between texts and lists the resulting authorship (or similarity) candidates in a logfile ("results.txt"). When the Consensus Tree option is selected, the script produces virtual Cluster Analyses for a variety of parameters, which then produce a diagram that reflects a compromise between the virtual CA graphs.

What to do to make it work:

0. Install R
   - 0.1. Using the command "install.packages()", install additional libraries required by the script: ape, tcltk, tcltk2.
   - 0.2. Each project requires a separate and dedicated working folder. You might want to give it a meaningful and scholarly name (like "Sanskrit Poetry 11" rather than "Blah blah"), since the name of the folder will appear as the title in your graphs generated by the script.
   - 0.3. Place the script(s) you will be using in your folder. Depending on your operating system (Windows versus Linux or Mac), you must select the appropriate version of the script file(s).
   - 0.4. The texts for analysis (at least two) must be placed in your folder's subfolder named "corpus" (and nothing else).
   - 0.5. The text files need to be named according to the following scheme: "authorname_title.txt". Obviously, if instead of looking for authorial attribution you want to see if translators exhibit similar "styles", you should name your files "translatorname_title.txt"; if you're looking for stylistic similarity between writers of the same gender, use "gender_title.txt", etc. The title can be followed by any other information; don't overdo it, though, or it will not fit on the graph.
   - 0.6. The texts must be either ALL in plain text format, or ALL in html, or all in XML (quite frankly, the latter two options have not been extensively tested so far).
   - 0.7. If your texts are in plain text format and your operating system is Windows, your safest bet is to put them all in ANSI codepage; if your plain text files will be used in Linux, make sure they are all in UTF-8. In spite of that, some languages will still cause problems

(especially in Windows); for some reason, French is notoriously unruly (*Nous sommes vraiement désolés*).

0.8. Run R. At the prompt, move to your folder (the main folder, NOT the "corpus" subfolder) using the command **setwd()**. You can always check if you are where you want to be with **getwd()**.

0.9. Looks like you're ready to go.

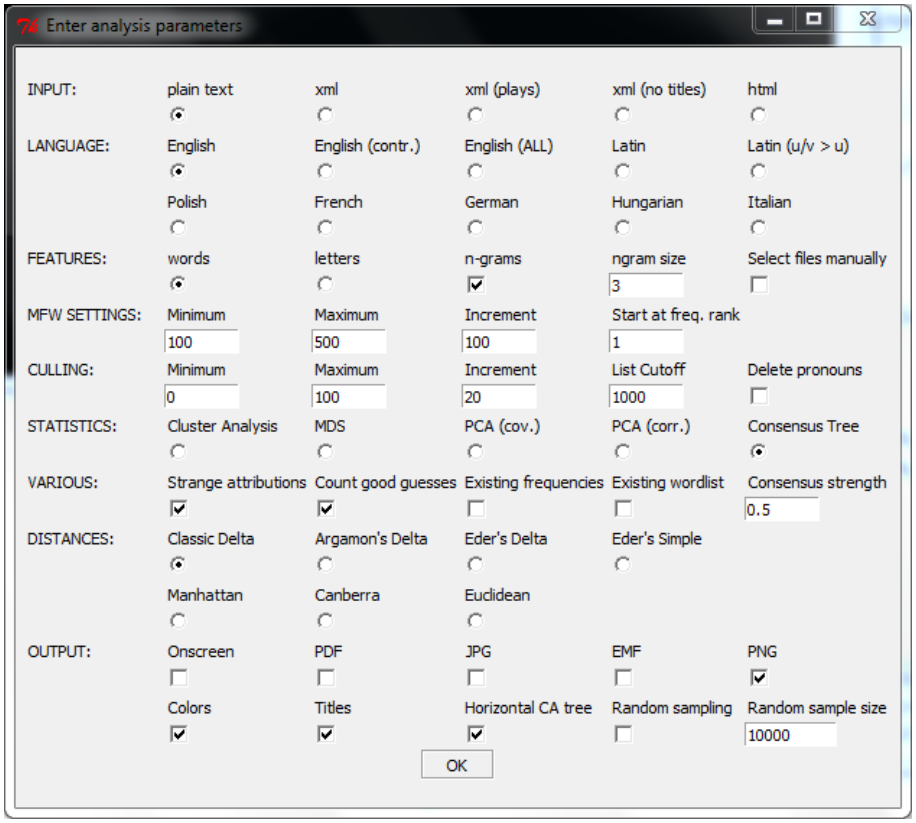
0.10. Invoke the script by typing **source("script_name")**

1. The suggested first step for beginners is to use the graphical user interface (GUI), which allows you to set all its parameters without tampering with the script. If you prefer to tamper, simply set the first setting in the script, interactive.mode.with.GUI = TRUE (default) to FALSE, and edit your parameters manually.

1.1 If you prefer to use the GUI, each run of the script will generate a config.txt file (saved in your working folder) which you can review (for instance, if you've forgotten your latest parameters), which is then retrieved at each subsequent runs of the script (so you don't need to set your favourite values all over again).

1.2 When you hover your cursor over the labels of each of the entries in the GUI, tool tips will appear.

2. The GUI



2.1. Input.

This is where you specify the format of your corpus. The available choices are:

2.1.1. **plain text**: plain text files (see 0.6 above).

2.1.2. **xml**: XML files; this option gets rid of all tags and TEI headers.

2.1.3. **xml (plays)**: XML files of plays; in this case, all tags, TEI headers, and speakers' names between <speaker>...</speaker> tags are removed.

2.1.4. **xml (no titles)**: XML contents only: all tags, TEI headers, and chapter/section (sub)titles between <head>...</head> tags are removed.

2.1.5. **html**: every attempt is made to get rid of HTML headers, menus, links and other tags.

2.2. Language:

This setting is needed to make sure that pronoun deletion (see 2.5.5 below) works correctly. If you decide not to remove pronouns from your corpus (which improves authorship attribution is some languages), this setting is immaterial (unless you're using English; see immediately below).

2.2.1. **English**: this setting makes sure that contractions (such as "don't") are NOT treated as single words (thus "don't " is understood as "don" and "t"), and that compound words (such as "topsy-turvy") are NOT treated as one word (thus "topsy-turvy" becomes "topsy" and "turvy").

2.2.2. **English (contr.)**: this setting makes sure that contractions (such as "don't") ARE treated as single words (thus "don't " is understood as "don^t" and counted separately), but compound words (such as "topsy-turvy") are still NOT treated as one word (thus "topsy-turvy" becomes "topsy" and "turvy").

2.2.3. **English (ALL)**: this setting makes sure that contractions (such as "don't") ARE treated as single words (thus "don't " is understood as "don^t" and counted separately), and that compound words (such as "topsy-turvy") ARE treated as one word (thus "topsy-turvy" becomes "topsy^turvy").

2.2.4. **Latin**: this setting makes sure that "v" and "u" are treated as discrete signs in Latin texts;

2.2.5. **Latin.corr**: since some editions do not distinguish between "v" and "u", this option provides a consistent conversion to "u" in each text.

2.2.6. For all other languages, apostrophes do NOT join words and compound (hyphenated) words are split .

2.3. Features

In classical approaches, frequencies of the most frequent words (MFW) are used as the basis for multidimensional analyses. It has been argued, however, that other features are also worth considering, especially word and/or letter n-grams. The general concept of n-grams is to combine a string of single words/letters into a sequence of n elements. Given a sample sentence "This is a simple example", the letter 2-grams are as follows: "th", "hi", "is", "s ", " i", "is", "s ", " a", "a ", " s", "si", "im", "mp", etc. The same sentence split into word ???3-grams reads "this is", "is a", "a simple", "simple sentence". Another question is whether it really increases the accuracy of attribution.

Further reading: Eder, M. (2011). Style-markers in authorship attribution: A cross-language study of the authorial fingerprint, "Studies in Polish Linguistics" 6: 101-16. Also, David Hoover will be presenting on the subject at DH2012.

2.3.1. **words**: words are used as the unit. Of course, the longer the n-grams, the fewer they are.

2.3.2. **letters**: letters are used as the unit.

2.3.3. **n-grams**: if checked, this option makes sure n-grams are used as the unit rather than individual words or letters (but it doesn't make much sense to perform your analyses on individual letters, does it?)

2.3.4. **n-gram size**: this is where you specify your n for your n-grams.

2.3.5. **select files manually**: normally, the script performs the analysis on all files in your "corpus" subfolder; if this option is checked, a window appears for you to

choose your files from the subfolder. Make sure you select more than two, or the script will complain.

2.4. MFW Settings: this is where you decide the size of the most-frequent-word list that will be used for your analysis.

2.4.1. **Minimum**: this setting defines how many words from the top of the word frequency list for the entire corpus will be used in your analysis in the first (or only) run. Thus a setting of 100 results in your analysis being conducted on 100 most frequent words in the entire corpus.

2.4.2. **Maximum**: this setting defines how many words from the top of the word frequency list for the entire corpus will be used in your analysis in the last (or only) run. Thus a setting of 1000 results in your analysis being conducted on 1000 most frequent words in the entire corpus**.**

2.4.3. **Increment**: this setting defines the increment by which the value of **Minimum** will be increased at each subsequent run of your analysis until it reaches the **Maximum** value. Thus a setting of 200 (at a **Minimum** of 100 and a **Maximum** of 1000) provides for an analysis based on 100, 300, 600 and 900 most frequent words.

N.B. For all statistics settings (see 2.6 below) except **Consensus Tree**, it is advisable to set **Minimum** and **Maximum** to the same value (this makes the **Increment** setting immaterial) unless you want to produce a great number of resulting Cluster Analysis, Multidimensional Scaling or Principal Components Analysis graphs.

2.4.4. **Start at freq. rank**: sometimes you might want to skip the very top of the frequency list, and this is where you specify how many words from the top of the overall frequency rank list for the corpus should be skipped. Normally, however, you should leave this at 1.

2.5. Culling

"Culling" is David Hoover's word for automatic manipulation of the wordlist. The culling values determine the degree to which words that do not appear in all the texts of your corpus will be removed. Thus a culling value of 20 states that words that appear in at least 20% of the texts in the corpus will be considered in the analysis. A culling setting of 0 means that no words will be removed; a culling setting of 100 means that only those words will be used in the analysis that appear in ALL texts of your corpus.

2.5.1. **Minimum**: this setting defines the first (or only) culling setting in your analysis (similarly to the minimum MFW setting).

2.5.2. **Maximum**: this setting defines the last (or only) culling setting in your analysis (similarly to the maximum MFW setting).

2.5.3. **Increment**: this setting defines the increment by which the value of **Minimum** will be increased at each subsequent run of your analysis until it reaches the **Maximum** value. Thus a setting of 20 (at a **Minimum** of 0 and a **Maximum** of 100) provides for an analysis using culling settings of 0, 20, 30, 60, 80 and 100.

N.B. For all statistics settings (see 2.6 below) except Consensus Tree, it is advisable to set Minimum and Maximum to the same value (this makes the Increment setting immaterial) unless you want to produce a great number of resulting Cluster Analysis, Multidimensional Scaling or Principal Components Analysis graphs.

2.5.4. **List cutoff**: Usually, it is recommended to cut off the tail of the overall wordlist; if you do not want to cut the list, then the variable may be set to an absurdly big

number (and then you are advised to use a fast computer and exercise patience). This is independent of the culling procedure.

2.5.5. **Delete pronouns**: (this, too, is independent of the culling procedure). If this option is checked, make sure you have selected the correct language for your corpus (see 2.2. above). This activates a list of pronouns for that language inside the script. Advanced users can use this part of the script to remove any words they want. So far, we have pronoun lists for English, Polish, Latin, French, German, Italian, and Hungarian.

2.6. Statistics

2.6.1. **Cluster Analysis** of Delta distance table. This option makes sense if there is only a single iteration (or just a few). This is achieved by setting the MFW Minimum and Maximum to equal values, and do the same for Culling Minimum and Maximum.

2.6.2. **MDS**: Multidimensional Scaling of Delta distance table. This option makes sense if there is only a single iteration (or just a few). This is achieved by setting the MFW Minimum and Maximum to equal values, and do the same for Culling Minimum and Maximum.

2.6.3. **PCA (cov.)**: Principal Component Analysis using a covariance matrix. This option makes sense if there is only a single iteration (or just a few). This is achieved by setting the MFW Minimum and Maximum to equal values, and do the same for Culling Minimum and Maximum.

2.6.4. **PCA (corr.)**: Principal Component Analysis using a correlation matrix (and this is possibly the more convincing option of the two) . This option makes sense if there is only a single iteration (or just a few). This is achieved by setting the MFW Minimum and Maximum to equal values, and do the same for Culling Minimum and Maximum.

2.6.5. **Consensus Tree**: this option achieves a compromise between a number of virtual CA results for a variety of MFW and Culling parameter values.

2.7. Various

2.7.1. **Strange attributions**: this saves “wrong” authorship attributions to “results.txt”.

2.7.2. **Count good guesses**: this reports the number of correct guesses for each iteration and produces a ranking of the least unlikely authorship (or similarity) candidates in the logfile “results.txt”.

2.7.3. **Existing frequencies**: Normally, the script computes a huge table of thousands word frequencies for all texts in your corpus. This is a non-trivial task and the most time-consuming one of the entire procedure. If done once, there is no need to waste time and do it again, because the tables are also saved in the output file "table_with_frequencies.txt". To retrieve all the word frequencies from the file, check this option. BUT it MUST be UNCHECKED when you switch corpora in the same R session (or simply add or remove a text or two from your corpus), or when you switch from word to letter analysis, or change your n for your n-grams (or if you've suddenly remembered you've picked the wrong language).

2.7.4. **Consensus strength**: For Consensus Tree graphs, direct linkages between two texts are made if the same link is made in a proportion of the underlying virtual Cluster Analyses. The default setting of 0.5 means that such a linkage is made if it appears in 50% of the Cluster Analyses. Legal values are 0.4 – 1. This setting is immaterial for any other Statistics settings.

2.8. Distances:

This is where you choose the statistical procedure used to analyse the distances (i.e. the similarities and differences) between the frequency patterns of individual texts in your corpus. Although this choice is not trivial, some of the following measures seem to be more suitable for linguistic purposes than others. On theoretical grounds, Euclidean Distance and Manhattan Distance should be avoided in stylometry. Canberra Distance is quite troublesome but effective e.g. for Latin (it should be combined with careful culling settings and a limited number of MFW taken into analysis). For English, usually Classic Delta is a good choice. A theoretical explanation of the measures implemented in this script is forthcoming (???).

The available distance measures are as follows:

Classic Delta as developed by Burrows

Argamon's Linear **Delta** based on Euclidean principles;

Eder's Delta: (explanation and mathematical equation: soon;

Eder's Simple: (explanation and mathematical equation: soon;

Manhattan Distance: obvious and well documented;

Canberra Distance: risky, but sometimes amazingly good;

Euclidean Distance: basic and the most "natural".

2.9. Output

2.9.1. **Onscreen**: check this if you want to display the graph on the screen.

2.9.2. **PDF**: check this to obtain a PDF file with your graph.

2.9.3. **JPG**: check this to obtain your graph in jpeg format.

2.9.4. **EMF**: check this to produce a windows metafile of your graph (this does not work in Linux or Mac).

2.9.5. **PNG**: check this to obtain your graph in PNG format (probably the best option).

2.9.6. **Colors**: when this option is checked, the script will automatically assign the same colors to texts with the same first segment of their file names (the first string ending in "_"). Sadly, this pretty option does not work for Cluster Analysis graphs (yet?).

2.9.7. **Titles**: If this is checked, the graphs will contain a main (top) title (based on the name of your folder and your choice of the statistics option) and a subtitle (bottom) listing your selected distance, MFW, Culling, Feature and Consensus settings.

2.9.8. **Horizontal CA tree**: a horizontal Cluster Analysis tree is sometimes more legible.

2.9.9. **Random sampling**: when the analyzed texts are significantly unequal in length, it is not a bad idea to prepare samples as randomly chosen "bags of words". If this option is switched on, the desired size of the sample should be indicated. N.B. This only makes sense only if **Existing frequencies** is unchecked.
Further reading: Eder, M. (2010). Does Size Matter? Authorship Attribution, Short Samples, Big Problem. In "Digital Humanities 2010: Conference Abstracts." King's College London 2010, pp. 132-35.

2.10. Press OK

After a while, you should see some progress as names of files processed appear on the screen. When the process is completed with no errors, you will either see your graph displayed in R's graphic device (onscreen), or you can start looking for the various output files in your working folder.