Maciej EDER, Jan RYBICKI

Stylometry with R

Pedagogical University of Kraków, Poland

Stylometric studies, in all their variety of material and method, share two common features: the electronic texts they study have to be coaxed somehow to yield numbers, and the numbers themselves have to be processed with statistical software. Sometimes, the two actions are two independent parts of a given study. To give the simplest example, one piece of software is used solely to compile word frequency lists; then, one of the many commercial statistics packages takes over to extract meaning from this mass of words, draw graphs etc.

This approach works if standard statistical procedures are applied to the data. For multivariate word-frequency-based authorship attribution, for instance, once free AntConc produces word lists with frequencies, expensive (yet often university-licensed) SAS, SPSS or Statistica can be used to produce relative frequencies and correlation matrices thereof, and one of their many modules can in turn produce Principal Components (PCA), Cluster (CA), or Multidimensional Scaling (MDS) analyses. This is already a very comfortable situation when compared to what Burrows had to rely on in his seminal work on Jane Austen's style (Burrows 1987).

Yet, as stylometrists have begun to produce statistical methods of their own – to name but a few, Burrows's Delta, Zeta and Iota (Burrows 2002, 2006) and their modifications by other scholars (Argamon 2008, Craig and Kinney 2009, Hoover 2004a, 2004b) – commercial software, despite its wide array of accessible methods, becomes something of a straitjacket. This is why a number of dedicated stylometric solutions have appeared, targeting the specific analyses frequently used in this community.

Hoover's Delta, Zeta and Iota Excel spreadsheets are a pioneering and excellent example of this approach (Hoover 2004b). Constantly developed since at least 2004 (when one of the authors of this presentation received a CD-ROM with an early version), they have at least two major assets: they do exactly what the stylometrist wants (with several optional procedures) and they only require spreadsheet software that has become (for better or worse) the standard on most computers in the world today. This has been especially helpful for uses in specialist workshops and classrooms; the student only needs additional (and, often, free) software to produce word frequency lists and he/she is ready to go. Yet Excel imposes one limitation: it is overkill in terms of memory usage, which results in the slowness of its processing. Also, the two-stage nature of the process (a separate piece of software prepares word lists that can be later automatically imported into the spreadsheet) might be something of a problem simply because it would take an experienced Visual Basic programmer to make Excel compile the word lists themselves.

In this respect, Juola's JGAAP can directly import texts in a variety of formats and perform a whole variety of authorship attribution tasks with an imposing variety of methods, statistical approaches and material on which they are based (Juola *et al*. 2006, 2008). These can be further expanded by experienced programmers in Java.

Java is also the language of another software solution which takes possibly an even broader approach. Craig's Intelligent Archive (in its many flavours), apart from performing certain stylometric tasks, is also a corpus organizer; once the initial work of registering texts is done, it allows a versatile combination of individual texts and groups of texts (Craig and Kinney, 2010).

A new trend in producing stylometric tools is associated with R, a GNU project, a language and environment for statistical computing and graphics (www.r-project.org) in the image of

its commercial counterpart, S. While it came into its version 1.0 in 2000, it has been used for analyses on language only recently. Its strongest promoters in this community include authors of corpus-linguistics-oriented books on the usage of R, Baayen (2008) and Gries (2009). R has already been used in authorship attribution research by Jockers *et al*. (2008, 2010).

The scripts presented in this poster have begun with the first author's participation in an R workshop taught by Gries at University of Leipzig's 1st European Summer School "Culture and Technology." Very soon, a series of R scripts appeared, targeted at a variety of experiments with Delta and other distance measures. Very soon, too, it became evident that R is capable of processing texts and statistics in a fraction of the time needed by other tools. Also, R scripts can be relied on for doing the whole work themselves, from manipulating texts (typically, to produce word lists) all the way to graphing the results (and that in a great variety).

These capabilities were put tu use in the study of Delta's dependence on studied texts' sizes (Eder 2010), and on the behaviour of Delta at a variety of intervals in the word frequency rank lists in a variety of languages (Rybicki and Eder 2011). The R environment permitted us to cover huge statistics at unprecedented rates (often thousands of Delta iterations per hour in corpora of a hundred full-size novels). It also permitted us to use other statistical methods, such as PCA, CA or MDS, or those rarely used so far in this field – such as bootstrap consensus trees based on a study of Papuan languages by Dunn *et al*. (2005, quoted in Baayen 2008: 143-147). What is more, the entire analysis – from loading texts to final results in numeric and graphic form – can be accomplished with a single script. Our Delta script, for instance, did all the work: it processed electronic texts to create a list of all the words used in all texts studied, with their frequencies in the individual texts; normalized the frequencies with z-scores (if applicable); selected words from stated frequency ranges for analysis; performed additional procedures that (usually) improve attribution, such as Hoover's

automatic deletion of personal pronouns and culling (automatic removal of words too characteristic for individual texts); compared the results for individual texts; performed a variety of multivariate analyses; presented the similarities/distances obtained in tree diagrams; finally, produced the above-mentioned consensus tree – a new graph that combined many tree diagrams for a variety of parameter values (Fig. 1). It was our aim to develop a general platform for multi-iteration attribution tests; for instance, an alternate script produced heatmaps to show the degree of Delta's success in attribution at various intervals of the word frequency ranking list (Fig. 2).

What is more, despite the fact that the R environment might daunt (digital) humanists with its initially steep learning curve, it soon became evident that ready-made tools can be developed to make the full power of R accessible even to inexperienced users with its capability of working with Tcl/Tk graphic user interfaces (Fig. 3) prepared by the second author for two seminar groups of his MA students, who were asked to make the switch to R from the more traditional software, with success: one of these groups has already successfully defended their completed theses on authorship attribution and stylistic variety in a good number of corpora.

References

Argamon, S. (2008). Interpreting Burrows's Delta: Geometric and Probabilistic Foundations, *Literary and Linguistic Computing* **23**(2): 131-47.

Baayen, R. H. (2008). *Analyzing Linguistic Data. A Practical Introduction to Statistics using R*, Cambridge: Cambridge University Press.

Burrows, J. F. (1987). *Computation into Criticism: A Study of Jane Austen's Novels and an Experiment in Method*, Oxford: Clarendon Press.

Burrows, J. F. (2002). 'Delta': A Measure of Stylistic Difference and a Guide to Likely Authorship, *Literary and Linguistic Computing* **17**(3): 267-87.

Burrows, J. F. (2006). All the Way Through: Testing for Authorship in Different Frequency Strata, *Literary and Linguistic Computing* **22**(1): 27-48.

Craig, H. and Kinney, A. F. (eds) (2009). *Shakespeare, Computers, and the Mystery of Authorship*, Cambridge: Cambridge University Press.

Craig, H. and Whipp, R. (2010). Old spellings, new methods: automated procedures for indeterminate linguistic data, *Literary and Linguistic Computing* **25**(1): 37-52.

Dunn, M., Terrill, A., Reesink, G., Foley, R. A. and Levinson, S. C. (2005). Structural Phylogenetics and the Reconstruction of Ancient Language History, *Science* **309**: 2072-75.

Eder, M. (2010). Does Size Matter? Authorship Attribution, Small Samples, Big Problem, *Digital Humanities 2010: Conference Abstracts*, London, pp. 132-34.

Gries, S. Th. (2009). *Statistics for Linguistics with R: a Practical Introduction*, Berlin and New York: Mouton de Gruyter.

Hoover, D. L. (2004a). Testing Burrows's Delta, *Literary and Linguistic Computing* **19**(4): 453-75.

Hoover, D. L. (2004b) Delta Prime? *Literary and Linguistic Computing* **19**(4): 477-95.

Jockers, M. L., Witten, D. M. and Criddle, C. S. (2008). Reassessing authorship of the 'Book of Mormon' using delta and nearest shrunken centroid classification, *Literary and Linguistic Computing* **23**(4): 465-91.

Jockers, M. L. and Witten, D. M. (2010). A Comparative Study of Machine Learning Methods for Authorship Attribution, *Literary and Linguistic Computing* **25**(2): 215-23.

Juola, P., Noecker, J., Ryan, M., and Zhao, M. (2008). JGAAP3.0 – Authorship Attribution for the Rest of Us, *Digital Humanities 2008: Book of Abstracts*, Oulu, pp. 250-51.

Juola, P., Sofko, J. and Brennan, P. (2006). A Prototype for Authorship Attribution Studies, *Literary and Linguistic Computing* **21**(2): 169-78.

Rybicki, J. and Eder, M. (2011). Deeper Delta Across Genres and Languages: Do We Really Need the Most Frequent Words?, *Literary and Linguistic Computing* **26** (forthcoming).

Figure 1. An example of a consensus tree diagram generated by R.

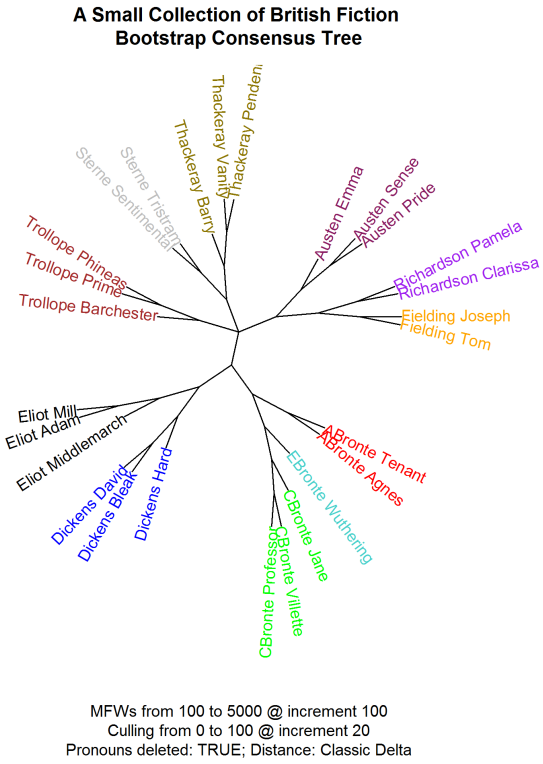

Figure 2. An example of a heatmap generated by R, showing the percentage of successful Delta authorship attributions at combinations of word list length and initial word rank in frequency lists

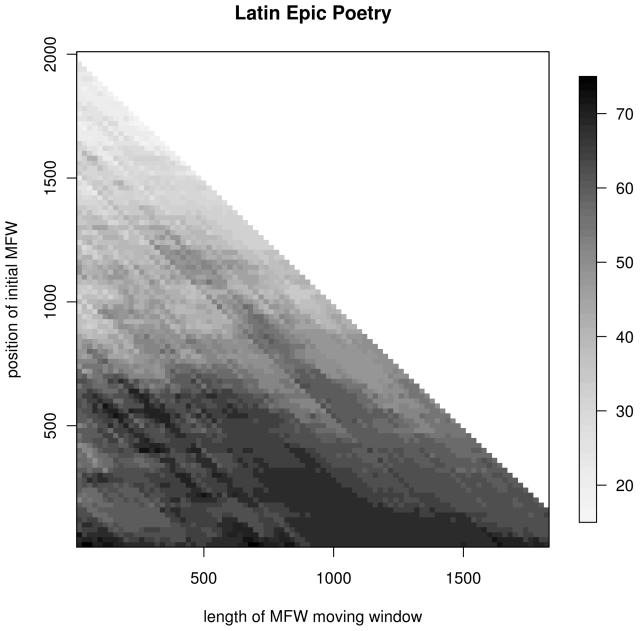

Figure 3. The GUI for the R script used to generate the diagram in Fig. 1.