text to data to insight

Whitt Kilburn, Department of Political Science and Data Inquiry Lab;
Matt Schultz, University Libraries

We provide an overview of text mining tools and strategies, with hands-on exercises from the digital humanities and social sciences....The workshop will focus on introductory theory and application of natural language processing, sentiment analysis, and document clustering techniques; no prior experience is required. More in-depth workshops on these subjects are available for the fall semester from the Data Inquiry Lab....Attendees should bring at least a networked tablet; to participate in all hands-on exercises, bring a laptop with open source software installed following the instructions at `https://www.gvsu.edu/datainquirylab/text-mining-11.htm`.

Learning objectives: text to data to insight 2017

1. Describe basic methods and tools of two areas of natural language processing, sentiment analysis and stylometry.
2. Use 'tokenization to explore word and character n-gram frequencies and relationships within a text.
3. Differentiate function or stop words from content words.
4. Explain Zipf's law of rank/frequency dependence in function and content words.
5. Use R tools to:
   (a) Apply lexicon based sentiment theory to an sentiment analysis of a text and visualize results.
   (b) Apply stylometric theory to a scaling and cluster analysis of most frequent words co-occurrences across texts of a corpus.
6. Reflect on potential applications of these tools within your discipline.

Order of topics

1. Introductions, us and you: Why are we here? What can we do together in about two hours?
2. Quick overview of topics we will study: sentiment analysis and stylometry
3. Use of a web-based tool for tokenization and text visualization
4. (short break)
5. R laptop setup and preparation for analysis
6. stylometric analysis of function words with the `stylo()` package and menu ; sentiment analysis with various R tools and command lines
7. QA and further resources

text to data to insight: A really brief step by step guide to using the stylo() package

This guide describes the steps for an analysis of the most frequent word co-occurrences in a text corpus.

1. Identify your text documents; store each document as a plain text file. Or if files are in HTML, store all in HTML. The package also reads XML. But file types must all be the same.

2. In stylo(), the typical end product — a visualization — displays the first part of each file's name, the part preceded by an underscore '_' character. If you want to display author names in the visualizations, label the files as 'authorname_title.txt'.

3. Place your corpus in a directory on your machine. Make sure the directory is labelled "corpus". The working directory in R needs to be set to one directory *above* this directory labelled "corpus". For example, the Shakespeare corpus is in a folder labelled 'Shakespeare', while the actual text documents are in a subfolder labelled 'corpus'. The working directory is set to the folder labelled 'Shakespeare'. Use the `getwd()` function to identify your working directory. Change the working directory if needed via TK.

4. Use the `dir()` function to list the contents of your working directory. After running the command in the R console — type `dir()`, do you see "corpus" in the results? — The directory containing your set of text files? If so, you have succeeded in setting your working directory where it needs to be! Again, the working directory needs to be one directory above the corpus; the `stylo()` package will place some useful files about your analysis in the working directory, and storing the files above the corpus of the analysis makes sense.

5. The package needs to be installed prior to use; package installation on your own laptop is one and done. After you installed it for the workshop — with `install.packages(``stylo")` — each time you start R, to load the `stylo()` package, type in the R console `library(stylo)`.

6. Now to use the point-and-click, graphical user interface (GUI) for `stylo`: type `stylo()` at the R prompt.

7. The GUI for stylo should appear. There are hints below, but stylo includes tooltips with the computer mouse for various options. Hover the mouse pointer over an option to see the tooltip.

8. Under the menu "Input" it should have the box "plain text" automatically checked — this makes sense if your corpus is in plain text.

9. Next to Language, the first option labelled 'English', will ensure that contractions, such as "can't", are **not** treated as a single word , and hyphenated words are split into constituent parts. The option "English (contr.)" treats contractions as single words ("don't" is counted as a whole, separately from "do not"), while compound words are split, and "English (ALL)" treats both contractions and hyphenated words as single words. Unless otherwise

10. Click "Features" — what do you want to analyze? Single word co-occurrences? If so, you would specify word n-grams of size "1". "MFW" stands for most frequent words. The default is

100, to analyze the 100 most frequent words. The differences in “Maximum” and “Minimum” are running a series of tests over different MFW frequencies.

   (a) *Important time saving advice* The ‘tokenization’ of word frequencies – or the creation of a table of frequencies — is time consuming. For a given type of n-grams, this process only needs to be performed once. Under ‘Features’, click the box for ‘Existing frequencies’ to save time by using a previously created set of frequencies.

11. “Culling” refers to the potential removal of words from the MFW word list if those words do not appear in all corpus texts. For example, a culling value of 30 means that only words appearing in at least 30 percent of the texts within the corpus would be potentially included in the list. Leave it at 0 to ignore culling.

12. Next, click “Statistics”. The default analysis is a cluster analysis. (For more info on setting cluster analysis types, see the stylo documentation). For a multidimensional scaling analysis, MDS, click the MDS button. And select the corresponding distance metric. In the workshop, we reviewed Burrow’s Delta (Classic Delta) and Euclidean.

13. Click Output and ‘OK’.

Questions for discussion and exploration: Stylometry

Using any of the corpora provided for the workshop,

1. Try a different count of single most frequent words (MFWs). How well do counts of 10 or 50 or 100 MFWs (word 1-grams) differentiate the corpus texts?

2. Consider different n-grams. How well do different word and character n-grams differentiate the corpus texts? For example, with texts encoded as parts of speech tags, typically word n-grams of five or larger are necessary to differentiate texts.

3. In your own area of inquiry, what are potential applications of stylometric analyses?

Files created by stylo() and what they are useful for

After you run a `stylo()` analysis, the package will create a series of files in your corpus directory. These files are the following:

1. ‘stylo_config’, which lists the stylo command configurations you selected in the GUI.

2. ‘wordlist’, listing the MFWs selected in the GUI — the words used to construct the distance measures.

3. ‘table_with_frequencies’ with normalized z-score frequencies for all words. (Notice how large the text file is. There are ways to create a raw count of word frequencies using `stylo()` command lines, a subject covered in the stylometry DIL workshop for the Fall semester.

4. ‘frequencies_analyzed’, containing the normalized (z-score) word frequencies – the ones used in the analysis.

5. ‘distance_table’, for the number of MFWs you analyzed. This file reports the distance scores, such as Delta or Euclidean for words by text.

6. a .csv file with ‘nodes and edges’ in the title is file with network structured data showing strength of connection between texts.

There are many resources online for further study of stylometry and sentiment analysis. See the LibGuide on text mining. Use Google. Attend fall DIL workshops on sentiment analysis and stylometry.