Quantitative Text Analysis
Exploring and comparing texts

Kenneth Benoit

In this class we will continue to explore methods for text analysis available in Wordstat. We will analyze another corpus, Irish budget speeches from 2010. The class will focus on exploring and comparing texts, using search and retrieval methods, finding keywords, and identifying collocations.

To work with the Irish budget texts, you will need to either open the project you created with them in the first week. To begin with, add at least one variable and apply it to each text, so that we can analyze later results across more than just the document level.

For a starting point, we will work with the party variable. The parties, indicated in the filename, are FF (Fianna Fáil), FG (Fine Gael), The Green Party, Labour, and SF (Sinn Féin). Fianna Fáil and The Green Party were the coalition government at the time of the debate. If we were only interested in the side taken by each party in the debate, we could alternatively create a variable with two values, GOV (FF and Green) and OPP (all other parties).

1. Exploring keywords and context.

   (a) Load the UK manifesto texts project. For this set of texts, we will be researching (searching) and coding the position of each party on the adoption of the euro. If you encounter problems related to a 'teamwork' user logon, try choosing the 'Admin' user with the password 'admin'.

   (b) First, create a search for "single currency", "euro", "European currency", "British pound", and "pound". These should be an `or` search:

   From the top bar, chooose: `Retrieval -> Text Retrieval` Choose 'search unit' to be 'sentence'

   (c) Create a set of codes to be applied to sentences containing the search terms. This will be a five-part category scheme: Against the adoption of the euro; In favour of the euro; mildly positive (for instance given the right conditions); mildly negative (in favour of keeping the pound unless something really exceptional happens); and in favour of a popular referendum to decide the matter. Note that if there are additional categories that arise from coding, feel free to add them. To add a new code, from the 'search hits' window, click on the blue plus. Add a code name (e.g. 'mildly positive') and a code type (e.g. 'euro position')

   (d) Examine each text for the searched terms. For each natural sentence in which search terms occur, apply one of the codes. Keep track of "false positives" or sentences returned that are not about the current issue. From the main document window, you can add comments to codes by right clicking on the code mark at the right of the text.

   (e) Summarize the codes once you are finished to characterize how pro or anti-euro each party is over time. How you do this is open-ended, explore the options under Analyse → coding frequency and Analyse → coding by variable.

   (f) If you have coded enough segments, you may find the cluster analysis `(Analyse -> Cluster Analysis)` interesting. You can also see a similar cluster analysis based on word frequencies in WordStat `(Crosstab, Correspondence Analysis)`

   (g) Once you have created the project and added some variables to the documents, open Wordstat by choosing 'Analyze → Content Analysis'.

2. Preliminary dictionary analysis.

   (a) From the Wordstat dictionary file, we will be using the Laver-Garry dictionary. From the dictionaries screen you should be able to open the Laver-Garry dictionary by choosing "Open" next to the Categorization dictionary dropdown.

   (b) The dictionary has a hierarchical structure. Explore the sub-categories by clicking on the folders in the lower half of the dictionary screen. The third level shows the actual terms (word stems) that are grouped under each category.

   (c) View the results of the Dictionary categorization in both the frequency and cross-tab screens. Use the 'Level' option to show more or fewer categories, and to show full hierarchical paths or just counts summed under each heading. Note the effect that exclusion or pre-processing filters might have on your results — for example, if you have chosen, in the options screen, to exclude terms that occur in more than 80% of documents, common function words and common political terms will not be counted, even if they are in your categorization dictionary.

   (d) Explore the 'Phrase Finder' screen. In English, there are many multi-word phrases that would be a single word in a more agglutinative language. Sometimes it is useful to identify these terms, both for manual content analysis and to help our word frequency matrix differentiate between terms like 'social welfare' and 'social science'.

   (e) Adjust the min words and max words options, and sort by TF x IDF to see the most salient phrases. You will need to set the exclusion dictionary or pre-process to exclude very common words. The button to the right of 'abc' allows you to view the phrase distribution by variable, and the dendrogram button to the right of that uses a statistical measure to cluster similar phrases. The Wordstat documentation provides further information on these functions.

   (f) In the cross-tab screen, use the 'statistic' dropdown to examine how terms variable according to the government and opposition variables. For example, the Chi-square statistic is a measure of how each term varies among the variables you have added to each text. How do these values change if you apply the Porter stemmer before examining the term variation? For example, compare the entries for 'CUT' and 'CUTS' before stemming, and the single entry for 'CUT*' after stemming.