Yoshikoder: An Open Source Multilingual Content Analysis Tool for Social Scientists

Will Lowe*
will.lowe@nottingham.ac.uk
Methods and Data Institute, University of Nottingham

This short paper is about the Yoshikoder[1], an open-source desktop tool for performing classical computer-aided content analysis in multiple languages. The paper starts with some background on content analysis, continues with a short technical characterization of the Yoshikoder as a content analysis tool, and concludes with a some necessarily brief examples of the kind of analysis the Yoshikoder makes possible.

Classical Content Analysis

By classical content analysis I mean the tradition of examining word frequencies, creating concordances, and building content dictionaries in order to operationalize substantively interesting aspects of document meaning (West, 2001; Neuendorf, 2002, for reviews).

There are, of course, other traditions of content analysis e.g. discourse analysis, cognitive mapping, and collocational clustering, with specialized software available often available to apply each method (see Herrera and Braumoeller, 2004, for some comparisons). Content analysis also borrows technology from computational linguistics (Manning and Schütze, 2000; Jurafsky and Martin, 2000). However, the Yoshikoder is designed primarily for classical content analysis so I will not discuss alternative methods.

Classical content analysis was originally performed manually (see Krippendorff, 1980, for a history), but its emphasis on classifying and counting individual words made it quite straightforward to automate (see e.g. Stone, 1997, for an early example). As of 2006 there exists a wide variety of computer packages to help researchers perform classical content analyses (see Lowe, 2002, for a functional typology and review).

Yoshikoder as a Content Analysis Tool

Among existing existing packages, the Yoshikoder is the only package I am aware of that runs on any operating system, is distributed for free as open-source software, and deals with documents in any natural language. Let me provide some motivation for these features.

Yoshikoder is written in the Java language as a desktop application that runs on all major operating systems. Moreover, the content analysis machinery does not require any particular interface, and may be used in a server environment[2].

It is a minimal requirement of the scientific replication standard (King, 1995) that the algorithms in a content analysis package used for academic research be known. And it is helpful if

*Thanks to Michael Laver and John Garry for providing their U.K. political manifesto dictionary. Work on the Yoshikoder has been funded by Harvard University's Weatherhead Center for International Affairs at Harvard and the Princeton Center for Advanced Study.

[1] See http://www.yoshikoder.org for details.

[2] Yes, a web service version of the Yoshikoder is under development

the financial cost of replication is kept low. The Yoshikoder fulfils both by making all its source code publically available for download[3].

Computerized versions of classical content analysis have traditionally been performed on West European language sources – a fact no doubt partly determined by path dependencies in the history of computing. However, current computer languages now have excellent implementations of the universal character set and character encoding standard Unicode (equivalently ISO 10646)[4]. The practical consequence of this technological advance is that the Yoshikoder's document import mechanism allows users to work with documents in almost any encoding, whilst operating internally in Unicode. For example, a project might contain russian language documents encoded variously in ISO-8859-5, KOI-9, and Macintosh Cyrillic.

To ensure that the document is segmented into words appropriately, users may also specify a document locale[5] e.g. Russian, as spoken in Russia, in order to distinguish it from other languages also written in cyrillic script such as Serbian. For some languages, notably Chinese, Japanese, and Thai, segmenting a text into words automatically is a difficult and computationally demanding task. For these cases, the Yoshikoder allows third parties to write and distribute 'tokenizer plugins' that perform the relevant segmentation. A tokenizer plugin for Chinese, as spoken in the People's Republic of China, is currently available.

Using the Yoshikoder

The first thing to do with the Yoshikoder is to make a project. A project consists of a single content analysis dictionary and a set of documents from your filesystem. When the program starts it loads the last project you were working on, or the default dictionary with no entries and an empty document set.

Adding Documents

Now to add some documents. For the examples below I'll use the U.K. political party manifestos from the 1992 and 1997 elections[6]. When you add or import a document Yoshikoder does not copy the document but simply keeps a reference to where it is, how it's encoded and what locale it is written for. Only when you click on its name in the interface is the text loaded from the file. This means you can have a project with more documents in it than would fit in computer memory.

Computing Word Frequencies

Before moving to the dictionary there are several things we can ask, starting with a sortable word frequency breakdown for each document. This is perhaps the simplest form of report available. It lists the number of times each word type occurs and the corresponding proportion of the text its tokens take up. Reports are shown as tables that can be saved in various formats or just highlighted, copied, and pasted into other applications. For comparative purposes it is often more useful to have a unified frequency report where the same document statistics are listed for every word type appearing in any document and collected in one large table. The unified frequency report combines the word frequency statistics for all the documents you select in the interface. For example, the unified report notes that in the 1992 manifestos 'police' contributes 34 tokens to the Conservatives, 14 to the Liberal Democrats, and only 4 to the Labour Party.

[3] http://www.yoshikoder.org also hosts content dictionaries in Yoshikoder format to foster replication and reuse.

[4] Language support for Unicode is now often *better* than that of the underlying operating system, so the problem of working with foreign language materials is reduced to locating a suitable font to display them in.

[5] A locale is the combination of a language and a country. Sometimes both are necessary to determine appropriate word segmentation

[6] Available to download from http://www.wordscores.com

Content Analysis Dictionaries

Word frequency statistics are useful for getting a feel for your documents, but for further analysis a dictionary is helpful. A Yoshikoder dictionary is a tree of possibly nested categories containing patterns. A pattern is an possible wildcarded string that matches one or more words in a text. The asterisk is used to indicated one or more unspecified letters, e.g. chin* matches both 'china' and 'chinese'. Each category and pattern has a name and an optional numerical score.

The Yoshikoder can read dictionaries in its own format (a simple XML dialect), and also files created by the DOS-based content analysis program VBPro[7]. The Yoshikoder allows users to add, edit and move categories and patterns manually, but for these examples I will use Laver and Garry's dictionary of policy position terms[8] (Laver and Garry, 2000). The Laver-Garry dictionary organizes 594 patterns into 9 top level and 18 nested policy categories.

Comparing Documents using a Dictionary

Applying the dictionary to the 1992 Labour manifesto reveals that about 0.5% (55 words) of the manifesto consists of terms in the category law and order. The category with the highest proportion is economy taking up 8.3% (952) of the manifesto's words. If these seem like small proportions, bear in mind that about 50% of all tokens in English text are contentless grammatical function words.

To examine the policy differences between 'old' and 'new' Labour, run a report comparing the 1992 (old) and 1997 (new) Labour manifestos with respect to dictionary categories. This reveals that the proportion of words in the category economy>pro-state that is, the subcategory of economy containing words indicating more governmental influence in the economy, has shrunk by half (from 0.04 to 0.025), whereas representation of the category law and order has more than doubled (from 0.005 to 0.011), consistent with substantive theory about the policy preferences of the current U.K. Labour party.

Making Reliable Comparisons

Since these are large changes in small proportions it is useful to have a measure of reliability. The Yoshikoder offers a statistical comparison report that computes risk ratio estimates and confidence intervals for each dictionary category. For example, to test the reliability of the increase in representation of law and order, the Yoshikoder computes the ratio of the probability of seeing a law and order term given that we are reading the 1997 Labour manifesto and the same probability given that we are instead reading the 1992 manifesto. Call the ratio r. If $r > 1$ then the 1997 manifesto contains $(r - 1)100\%$ more law and order words. Alternatively, if $r < 1$ it contains $(r^{-1} - 1)100\%$ fewer law and order words. If, moreover, the confidence interval for r *also* excludes one, then the percentage change in law and order words is statistically significant. In these manifestos the risk ratio for for law and order is 2.32 with 95% confidence interval $[1.72, 3.12]$, an increase of 132% between old and new Labour platforms. The estimate and interval for economy>pro-state is 0.62 $[0.54, 0.71]$, a 60% decrease.

Measuring Local Context

The Yoshikoder is designed to compare whole documents on the basis of the categories in a content analysis dictionary. However, it is sometimes useful to apply the dictionary to more local contexts, e.g. to gain a idea of how some person, place or theme is talked about in subsections

[7] Intermittently available from http//mmmiller.com/vbpro/vbpro.htm. VBPro format has a particularly simple form that lets users create large dictionaries offline from existing wordlists rather than adding patterns manually via the interface.

[8] Available from http://www.yoshikoder.org.

of the document. One way to get at this information is to organize a dictionary so that it has categories for e.g. positive and negative language, and another category capturing references to the subject. Generate a suitably wide concordance for the subject category, and run the Yoshikoder's concordance report. The concordance report bundles all the left and right surrounding context from each line of the concordance, centered on references to the subject, to form a pseudo-document of local contexts. The dictionary is then applied to this pseudo-document generating a characterization of the context local to the subject in the form of a regular dictionary report[9].

The Bigger Picture

The Yoshikoder is designed to help non-technical social scientists perform classical content analyses on text in arbitrary languages. Using the Yoshikoder helps support the replication standard and annoys people who sell similar functionality in proprietary packages, but it is also part of a larger project to unify, standardize, and disseminate the theory and technology of content analysis. To this end, the Yoshikoder homepage also hosts a free application for converting PDF, Word documents and web pages into plain text in bulk for subsequent content analysis, and should soon host the older but widely-used DOS-based content analysis program VBPro. The homepage also hosts content analysis dictionaries in various languages. If you have a dictionary you'd like to make more widely available, I'll translate it into Yoshikoder format and host it there too.

References

Herrera, Y. F. and Braumoeller, B. F., editors (2004). *Symposium on discourse and content analysis*. Qualitative Methods Newsletter. Available from http://www.people.fas.harvard.edu/%7Eherrera/papers.html.

Jurafsky, D. and Martin, J. H. (2000). *Speech and Language Processing*. Prentice-Hall, Upper Saddle River NJ.

King, G. (1995). Replication, replication. *PS: Political Science and Politics*, 28(3):443–499.

Krippendorff, K. (1980). *Content Analysis: An Introduction to Its Methodology*. Sage, Beverly Hills CA.

Laver, M. and Garry, J. (2000). Estimating policy positions from political texts. *American Journal of Political Science*, 44(3):619–634.

Lowe, W. (2002). A review of content analysis packages. Available from http://www.wcfia.harvard.edu/misc/initiative/identity/.

Manning, C. D. and Schütze, H. (2000). *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge MA.

Neuendorf, K. A. (2002). *The Content Analysis Guidebook*. Sage, Thousand Oaks CA.

Stone, P. J. (1997). Thematic text analysis: New agendas for analyzing text content. In Roberts, C., editor, *Text Analysis for the Social Sciences*. Lawrence Erlbaum Associates.

West, M. D., editor (2001). *Theory, Method,and Practice in Content Analysis*, volume 16 of *Progress in Communication Sciences*. Ablex, Westport CT.

[9] This process is a rather noisy way to detect predication, but the full linguistic analysis is unlikely to be more reliable, particularly when there is no one to write the parser for the language involved.