Experiences Parallelising an Existing NLP Pipeline: Tagging Hansard

Stephen Wattam*, Paul Rayson*, Marc Alexander* & Jean Anderson*

This poster describes experiences processing the two-billion-word Hansard corpus using a conventional batch-oriented NLP pipeline on a high performance cluster. We report how we were able to parallelise and apply a “traditional” single-threaded application to a platform that differs greatly from that for which it was originally designed.

The Corpus

Hansard is the official published report of oral and written UK Parliamentary proceedings. It is an edited verbatim report of speeches in both the House of Commons and the House of Lords. Members’ words are recorded by Hansard reporters and then edited to remove repetitions and obvious mistakes without changing the meaning.

The Hansard corpus is one of the biggest humanities data sets in the UK but was limited in its use by being tagged only by speaker and date. To provide greater utility we wanted the speeches to be searchable by parts of speech (POS) and by topic. Historic records back to 1803 have been digitized in XML format and are publicly available on the Historic Hansard web site created by the Commons and Lords Libraries and Millbank Systems. There one can search by word or phrase then filter results by speaker or House (Lords or Commons).

The full 200-year collection is 2,271,985,142 words and 32,732 xml of data, including mark-up. The corpus is split into 7,545,103 XML files, each representing one speech, with a median size of 147 words.

Method & Toolchain

We downloaded the XML corpus from the Historic Hansard site and standardized the XML coding in order to prepare it for the NLP toolchain. As well as POS tagging, we applied the USAS semantic tagger. Once these annotation steps were complete, we needed to produce word, POS and semantic frequency lists for each speech as per the standard Wmatrix tag wizard pipeline. Finally, in order to expose the key topics of each speech we compared each semantic frequency list to a standard reference corpus, the British National Corpus spoken sampler (one million words) using the log-likelihood statistic to find a set of key semantic tags.

The toolchain being used to tag the corpus was a combination of tools comprising the tag wizard of the Wmatrix system. This consists of a number of tagging and analysis tools, each communicating using intermediate files and managed using a series of shell scripts. The tag wizard shares a lot of commonalities with standard NLP processing pipelines as it consists of two annotation stages—CLAWS and USAS—and a frequency profiling and keyness comparison step known as Tmatrix.

The Wmatrix tag wizard toolchain was originally designed and developed to run on commodity computing hardware in a batch processing fashion. The toolchain thus respects a number of common limits that apply to desktop computers (e.g. low memory limits) and attempts to exploit other resources that are available in abundance (fast sequential file I/O). Many of these properties do not transfer to larger clusters, and a number of changes had to be made in order to align the processing stages with the resources available on the HEC system.

The maximum memory configuration is 96 megabytes using six 16-megabyte DSIMMs.

In many cases, the alternative to deployment on a HEC cluster will be use of one or many commodity desktop machines. The desktop system (i5, RAID) runs individual jobs significantly more quickly. Fitting a linear model indicates that the desktop is able to run jobs approximately 2.1 times faster than a single HEC core. The same model indicates a 400 second job startup overhead on the HEC (including copying of files). Two weeks was spent developing and testing the HEC deployment of the toolchain. Of this, a significant portion was spent adapting the toolchain to run on the scheduler without restriction from the shared filesystem.

The problem is embarrassingly parallel, and the design of many HEC facilities is well suited to exploit that. It is certainly the case that, even if we had manually split the data and run it on (faster) commodity hardware, we would still have incurred significant overhead in doing so.

Further Information

Tagged for the Parliamentary Discoverse project at the University of Glasgow: http://parldev.jiscresolv.org/ top. Toolchain elements include CLAWS (http://nrlx.lancs.ac.uk/claws/) and USAS (http://nrlx.lancs.ac.uk/usas/), and form much of the Wmatrix corpus analysis and comparison tool (http://nrlx.lancs.ac.uk/wmatrix/)

*Dept. of English Language, University of Glasgow, marc.alexander@glasgow.ac.uk jean.anderson@glasgow.ac.uk
*School of Computing & Communications, Lancaster University, s.wattam@lancaster.ac.uk p.rayson@lancaster.ac.uk

With thanks to Mike Pacey, HPCC Manager, Information Systems Services, Lancaster University for his assistance with this project. Central quote is from the WPARI(ition) UK services manual, the same model upon which CLAWS was developed.