README for Trotman’s implementation of Kelly’s algorithm.

NOTE

Please keep me ([andrew@cs.otago.ac.nz](mailto:andrew@cs.otago.ac.nz)) informed of any use of this software. It is open source, you can do what you like with it so long as due credit is given to me for use of it or any part of it (that is, my name appears at the top of the files, or if you snip parts of it then my name appears in those snippets).

If you use this software in INEX, remember to change the participant-id in the output files.

REFERENCES

The Algorithms was originally described in:

Itakura, K. Y. and Clarke, C. L. 2007. University of Waterloo at INEX2007: Adhoc and Link-the-Wiki Tracks. In *Focused Access To XML Documents: 6th international Workshop of the initiative For the Evaluation of XML Retrieval, INEX 2007 Dagstuhl Castle, Germany, December 17-19, 2007. Selected Papers*, N. Fuhr, J. Kamps, M. Lalmas, and A. Trotman, Eds. Lecture Notes In Computer Science, vol. 4862. Springer-Verlag, Berlin, Heidelberg, 417-425. DOI= http://dx.doi.org/10.1007/978-3-540-85902-4\_35

Jenkinson (University of Otago) implemented the algorithm as a baseline for INEX 2008 and achieved substantial improvements over Itakura’s implementation (University of Waterloo). Trotman (University of Otago) saw the official scores for INEX 2008 and, unbelieving of the size of the improvements, re-implemented from scratch without referenced to any prior implementations. The Jenkinson implementation is described in:

Jenkinson, D., Leung, K.-C. and Trotman, A. 2008. Wikisearching and Wikilinking. In INEX 2008 Workshop Pre-proceedings, S. Geva, J. Kamps, A. Trotman, Eds. 330-344.

Included here is the source code for Trotman’s implementation. It was hacked up using Microsoft Visual C/C++ Version 15 (Visual Studio 2008) under Windows Vista in the two weeks before INEX 2008 and is provided as a reference implementation. Please cite the full-proceedings paper if you use this.

ALGORITHM

Kelly’s algorithm identifies all anchor-texts in the collection and all targets for those anchor-texts. It then identifies the number of documents in which the anchor-text targets that target. The document-frequency of the anchor-text is also computed and the strength of a link, gamma, is:

Gamma = docs\_targetting\_target / document\_frequency.

In the case of overlapping anchor-texts starting on the same word, the longest anchor-text is always chosen.

So, identify all possible anchor-texts in a document, rank by gamma, and then choose the top n (in the case of INEX file-to-file linking, n=250).

QUICK-START

Link\_index.bat: change directory to the root of the INEX 2008 document collection and execute this batch files. After two passes over the collection and some passes over those files an index will be generated. I’ve got no idea how long it takes, I ran it overnight on my laptop as my disk is slow

Noun\_boost.bat: generated a directory with all the INEX 2008 manually assessed topics in it called ltw\_2008\_manual\_topics and in the directory above that place the links.idx file (renamed to case\_sensitive\_links.idx). Executed this batch file and you’ll get a stack of output files that can be loaded into the link-the-wiki evaluation tool.

INDEXING

The index is built in several passes. In the first pass (link\_extract) all collection\_link links are extracted from all documents.

Usage: link\_extract <infile> [<infile>…]

The output is written to stdout (so redirect it to a file) the output format is:

<source\_document> <target\_document> <anchor\_text>

In the second pass (link\_index) the output of link\_extract is sorted first by anchor\_text, then by target\_document. The document-frequency and collection-frequency values are then computed and stored alongside the anchor\_text.

Usage: link\_index <infile>

The output is written to stdout (so redirect it to a file) the output format is:

<anchor\_text>:<target,tdf,tcf>[…]

In the third pass (link\_extract\_pass2) the entire collection is again passed over, but this time looking for the document-frequency and collection-frequency of the given anchor\_texts occurring anywhere in the files, either as (or not) a link.

Usage:link\_extract\_pass2 <infile> [<infile>…]

The output is written to stdout (so redirect it to a file) the output format is:

<document-frequency>:<collection-frequency>:<anchor\_text>

Finally, in the fourth pass (link\_index\_merge) it is necessary to merge the pieces of the index together into a single index that can be used for link-prediction.

Usage:link\_index\_merge <output\_from\_link\_extract\_pass\_2> <output\_from\_link\_index>

The output is written to stdout (so redirect it to a file) the output format is:

<anchor\_text>:[<df>,<cf>]:<target,tdf,tcf>…

LINK PREDICTION

The link-precision software (link\_this) reads the index and the orphan. It identifies the links in the orphan, removes them from the index, and then links the orphan producing file-to-file links on stdout. There are many parameters including:

-noyears

does not produced links for anchor-texts that are 4 digit numbes

-runname:runname

sets the run-name in the output XML to runname

-propernounboost:0.3

All anchor-texts in which the first letter of all words is upper case receive a boost of 0.3 to gamma

-targets:5

Each link is allowed to target up-to 5 links

-anchors:250

Identify up-to 250 anchor-texts per orphan

-lowercase

By default the index is case-sensitive. Many of the index generation programs have a –lowercase option that converts everything into lowercase, this generating a case-insensitive index. If you have a case insensitive index then use link\_this –lowercase or else you’ll get a case mismatch between the orphan and the index and the runs will perform badly.

Usage: link\_this [-lowercase] [-noyears] [-runname:name] [-anchors:<n>] [-targets:<n>] [-propernounboost:<n.n>] <index> <file\_to\_link> …

And it will generate an INEX 2008 link-the-wiki compatible output for file-to-file linking of each orphans.

DISCLAIMER

Everything is disclaimed. The software or its use does not for a contract. The software is provides “as is”. No liability for anything is accepted.