Titolo: Laboratorio di Algorithm Engineering

PROGETTI A.A. 2014-15

DOCENTI: P. Ferragina, R. Venturini (3 CFU l’uno)

**Searching Dictionary of Strings with Errors**

**(in collaboration with Yahoo! Research Barcelona)**

The problem consists of building a data structure to index a large dictionary of strings of variable length that supports efficient search with edit-distance at most one. This problem has important applications in correcting misspelled keywords in search engine queries.

The project consists of designing and implementing a solution based on one/more of libraries below in which the strings to be indexed are the queries in dataset. Then, the goal is to execute a serie of queries sampled from the provided dataset and possibly changed in one character (per query) in order to simulate misspells. The students have to profile and optimize their code and provide measurements of its time/space efficiency.

**Datasets**

MSN Query Logs (2006 and 2007). [14M and 100M queries].

**Software and Libraries**

* Compressed Permuterm Index - https://code.google.com/p/cpi00/
* Succinct Data Structure Library - https://github.com/simongog/sdsl-lite
* Ternary Search Tree - http://www.cs.princeton.edu/~rs/strings/
* Minimal Perfect Hashing - https://github.com/ot/emphf

**References**

D. Belazzougui, R. Venturini: Compressed String Dictionary Look-Up with Edit Distance One. CPM 280-292, 2012.

I. Chegrane, D. Belazzougui: Simple, compact and robust approximate string dictionary. CoRR abs/1312.4678, 2013.

Lez 1. Trie + Ternary Search Tree - Rossano

Lez 2. Rank/Select + Succinct tree encodings - Rossano

Lez 3. FMI + SA (no construction) - Paolo

Lez 4. q-grammi + (compressed) permuterm + (min-)perfect hash - Paolo

Lez 5. Discussion on the project - Rossano

**Compressing large collections of Web pages**

**(in collaboration with Tiscali)**

The problem consists of grouping Web pages and then compressing each group individually via Lzma2 compressor. The goal is to find the grouping strategy that minimizes the achieved compression. This problem has important applications in storage of text collections in search engines.

The project consists of designing and implementing an efficient clustering solution which can scale to Million of pages. Then, the goal is to test the proposed solution on the provided dataset by measuring the compressed space with Lzma2, and the overall time to compress the whole dataset. The students have to profile and optimize their code and provide measurements of its time/space efficiency. As a baseline for comparison the students can compare their solution with the one that sort Web pages by their URLs.

**Datasets**

Gov2 crawl - 25M Web pages. 81 Gb gzipped.

**Software and Libraries**

* Lzma2 compressor - http://www.7-zip.org/sdk.html
* Sorting large datasets: Unix sort or STXXL (http://stxxl.sourceforge.net/)
* For LSH implementation: <http://www.mit.edu/~andoni/LSH/>
* For shingling: <http://codingplayground.blogspot.it/2008/06/shingling-and-text-clustering.html> or <http://research.microsoft.com/en-us/downloads/4e0d0535-ff4c-4259-99fa-ab34f3f57d67/>
* Weka - http://www.cs.waikato.ac.nz/~ml/weka/ implements k-means and k-means++ in class SimpleKMeans with proper options.

**References**

A. Z. Broder, S. C. Glassman, M. S. Manasse, G. Zweig: Syntactic Clustering of the Web. Computer Networks 29(8-13), 1157-1166, 1997.

M. R. Henzinger: Finding near-duplicate web pages: a large-scale evaluation of algorithms. SIGIR, 284-291, 2006.

P. Ferragina, G. Manzini: On compressing the textual web. ACM WSDM, 391-400, 2010.

D. Arthur, S. Vassilvitskii: K-means++. ACM-SIAM SODA, 1027-1035, 2007. (<http://theory.stanford.edu/~sergei/papers/kMeansPP-soda.pdf>) *First 2 sections no proof.*

Lez 6. Sorting and Permuting - Paolo

Lez 7. Shingling, Min-Hash and LSH - Paolo

Lez 8. Clustering - Paolo

Lez. 9. Compressors LZ77 and Lzma, bounded window and others - Rossano

Lez 10. Discussion on the project - Paolo