Max-Planck-Institut für Informatik

Databases and Information Systems Group (AG 5)

Prof. Dr.-Ing. G. Weikum Dipl.-Inform. Hanglin Pan
Dipl.-Inform. Sergej Sizov
Dipl.-Inform. Stefan Siersdorfer

Selected Topics in Web Information Retrieval and Mining (WS 03/04)

Assignment 5

Handout on: Thursday, December 4, 2003

Due on: solutions will be discussed on Friday, Dec 12, 2003

Exercise 5.1:

Prove the correctness of the authority-based index pruning technique with fancy lists.

Exercise 5.2:

Consider queries with 2 keywords t1, t2, and suppose that these constitute the vast majority of the workload so that the index management should be optimized for this case. Discuss the advantages and disadvantages of keeping all index lists L(i) sorted by si(ti,dj) + r(dj)/2 in descending order (without any fancy lists).
What other techniques for optimizing this particular case can you conceive?

Exercise 5.3:

Consider again the case of 2-keyword queries, so that exactly two index lists are relevant in the query processing.
Assume we use the authority-based pruning with fancy lists. We would like to prune more aggressively by estimating the probability that a document whose score is not yet completely computed qualifies for the current top k.

a) Design a simple probabilistic model for this purpose (e.g., assuming that all si values that have not yet been seen are uniformly distributed between si_bound and 0).
Develop a decision procedure for terminating the scan of L(i) if the probability of missing a top k result drops below x percent (where x would typically be 1 or 5).

b) Try to improve the probabilistic model by taking into account statistics about the distribution of si values that you have already seen in the current scan (i.e., values in F(i) and from the start of L(i) to pos(i)).

Exercise 5.4:

Consider a cluster of computers with inverted index lists distributed across the cluster's nodes. Discuss how you can parallelize the processing of conjunctive keyword queries with the goal of linear scalability (i.e., a cluster with n times more nodes can achieve n times higher throughput).

Exercise 5.5:

Consider the index-based query processing of simple (conjunctive) keyword queries in a peer-to-peer system. Suppose that every peer has index lists for all keywords, but different peers cover different documents (but there is not necessarily a disjoint partitioning).
Design an efficient query processor for this setting (assuming that locating other peers is solved).