Assignment C – On Paper

Eirik Isene

March 8, 2016

1 Query evaluation

1.1 What are skiplists? How do they work?

A skiplist is a datastructure that allows us to search quickly within an ordered collection of elements. It works by having a layered linked list, where the lowest level is the ordered collection of elements. If we go up a layer the linked list found there will skip some elements of the ordered collection, and layers above will skip a few more than the previous, etc. This way one can start at the top-most layer and skip many elements, and move down a layer when the next skip is to far. Continuing to do so will land you at the correct spot at the bottom (original) layer at the location for your search/insertion/deletion/etc.

1.1.1 Are skiplists necessarily always beneficial for performance?

If the skip pointers are placed poorly, resulting in few successful skips, it won't be so beneficial for performance. However, a well working skiplist, with well placed skip pointers will shave off a lot of time by giving searches an "express lane" to the desired location without having to iterate through the entire list.

1.2 What's the difference between term-at-a-time and document-at-a-time evaluation

Basically, a term-at-a-time evaluation will update the score per term by iterating through all the documents in the terms document-list, where a document-at-a-time evaluation will update the score per document by iterating through all the terms in the documents posting-list.

1.2.1 Which approach would you use for an index where the posting lists are impact-ordered, and why?

If the posting lists are impact-ordered it is better to use a term-at-a-time evaluation since the posting-lists for each term are not sorted by some common ordering, thus making it har for us to update score document by document, but doing it term by term as in term-at-a-time won't pose any problems of the posting-lists aren't sorted by a common ordering.

1.3 Why is it beneficial to order posting lists according to static quality score $g(d)$?

When doing a term-at-a-time evaluation where posting-lists are impact-ordered by static quality scores, we can use this ordering so that the pages with the best quality are considered first. If we set a "quality threshold" we can assure the user that all search results found are from good pages according to our quality score implementation.