Kun Jiang, Yue-xiang Yang, 2016. Efficient dynamic pruning on largest scores first (LSF) retrieval. *Frontiers of Information Technology & Electronic Engineering*, **17**(1):1-14. http://dx.doi.org/10.1631/FITEE.1500190

Efficient dynamic pruning on largest scores first (LSF) retrieval

Key words: Inverted index, Index traversal, Largest scores first (LSF) retrieval, Dynamic pruning

Contact: Kun Jiang

E-mail: jk_365@126.com

ORCID: http://orcid.org/0000-0003-1316-5237

Motivation/Main ideas

- **Motivation:**
  - Index traversal techniques have been studied in addressing the query processing performance challenges of web search engines, but still leave much room for improvement.
  - We focus on the inverted index traversal on document-sorted indexes and the optimization technique called dynamic pruning, which can efficiently reduce the hardware computational resources required.
- **Main ideas:**
  - We propose the largest scores first (LSF) retrieval, in which the candidates are first selected in the posting list with the largest estimated score and then fully scored with the contribution of the remaining query terms.
  - We present efficient rank safe dynamic pruning techniques based on LSF retrieval, i.e., list omitting (LSF_LO) and partial scoring (LSF_PS) that fully make use of the query term importance.

LSF retrieval

- Consider the candidates in the posting list with the largest estimated score

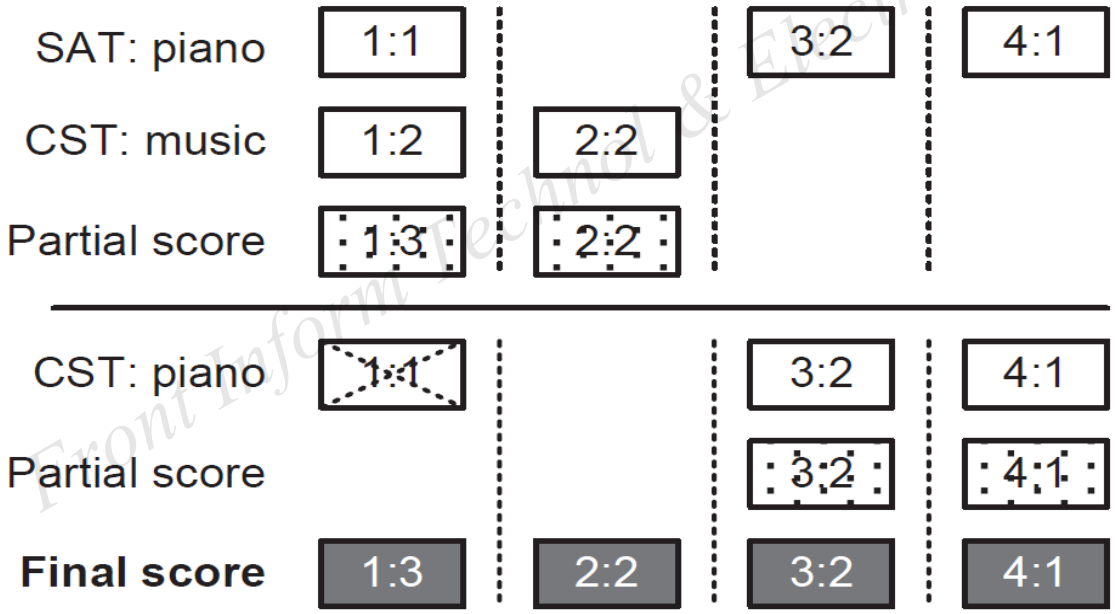

Fig. 3 An example showing the index traversal procedure of LSF with the query terms 'piano' and 'music'.

LSF-based pruning

- Two dynamic pruning techniques on LSF:

LSF_LO: Define $\rho(\{t_i\})$ as the largest score that the posting could achieve in the posting list for term t_i. Every time we finish one posting list of a given term, we check if the threshold $\theta > \rho(q_r)$, where $\rho(q_r)$ is the largest achievable score of new postings in the remaining terms q_r.

LSF_PS: Perform immediate checking of the achievable score of every posting. We first evaluate the score of this posting in the current list. If the sum of the score plus the upper bounds of the remaining lists is less than threshold θ, then we can abort the evaluation of the posting, which avoids large amounts of skipping.

Major results (I)

Table 1 Average query latency of the exhaustive index traversal techniques with different numbers of results

Algorithm	Average query latency (ms)		
	k=10	k=100	k=1000
OR_TAAT	1290.1	1250.8	1275.1
OR_DAAT	231.6	232.6	237.9
OR_LSF	279.6	281.2	284.8
AND_TAAT	238.7	238.3	238.4
AND_DAAT	24.1	24.1	24.3
AND_LSF	34.2	34.3	34.5

Major results (II)

Table 3 Average query latency of different dynamic pruning techniques for different numbers of query terms

Algorithm	Average query latency (ms)					
	Average	2	3	4	5	> 5
WAND	47.3	30.6	44.6	58.4	68.3	103.7
MaxScore	35.6	26.7	33.3	38.8	47.7	63.1
LSF_LO	61.1	38.9	56.9	70.2	89.9	140.3
LSF_PS	34.4	28.3	33.2	37.8	45.9	58.6

Conclusions

- We propose a novel inverted index traversal technique called LSF retrieval, which can effectively reduce the memory consumption of TAAT and the candidate selection cost of existing DAAT retrieval.

- To further reduce the number of postings processed, we present efficient rank safe dynamic pruning techniques based on LSF retrieval, i.e., list omitting (LSF_LO) and partial scoring (LSF_PS) that make full use of the query term importance.

- Experiments show that our approaches reduce the query latency by almost 27% over the WAND and are slightly better than the MaxScore.