Managing Tail Latency in Large Scale Information Retrieval Systems

Joel M. Mackenzie
RMIT University, Melbourne, Australia
joel.mackenzie@rmit.edu.au

Abstract

As both the availability of internet access and the prominence of smart devices continue to increase, data is being generated at a rate faster than ever before. This massive increase in data production comes with many challenges, including efficiency concerns for the storage and retrieval of such large-scale data. However, users have grown to expect the sub-second response times that are common in most modern search engines, creating a problem — how can such large amounts of data continue to be served efficiently enough to satisfy end users?

This dissertation investigates several issues regarding *tail latency* in large-scale information retrieval systems. Tail latency corresponds to the high percentile latency that is observed from a system — in the case of search, this latency typically corresponds to how long it takes for a query to be processed. In particular, keeping tail latency as low as possible translates to a good experience for all users, as tail latency is directly related to the worst-case latency and hence, the worst possible user experience. The key idea in targeting tail latency is to move from questions such as *"what is the median latency of our search engine?"* to questions which more accurately capture user experience such as *"how many queries take more than* 200*ms to return answers?"* or *"what is the worst case latency that a user may be subject to, and how often might it occur?"*

While various strategies exist for efficiently processing queries over large textual corpora, prior research has focused almost entirely on improvements to the *average* processing time or cost of search systems. As a first contribution, we examine some state-of-the-art retrieval algorithms for two popular index organizations, and discuss the trade-offs between them, paying special attention to the notion of tail latency. This research uncovers a number of observations that are subsequently leveraged for improved search efficiency and effectiveness.

We then propose and solve a new problem, which involves processing a number of related query variations together, known as *multi-queries*, to yield higher quality search results. We experiment with a number of algorithmic approaches to efficiently process these multi-queries, and report on the cost, efficiency, and effectiveness trade-offs present with each.

Finally, we examine how *predictive models* can be used to improve the tail latency and end-to-end cost of a commonly used *multi-stage* retrieval architecture *without* impacting result effectiveness. By combining ideas from numerous areas of information retrieval, we propose a prediction framework which can be used for training and evaluating several efficiency/effectiveness trade-off parameters, resulting in improved trade-offs between cost, result quality, and tail latency.

Awarded by: RMIT University, Melbourne, Australia.
Supervised by: J. Shane Culpepper and Falk Scholer.
Available at: https://jmmackenzie.io/pdf/jmm-phd-thesis.pdf.

Selected Publications

M. Crane, J. S. Culpepper, J. Lin, J. Mackenzie, and A. Trotman. A comparison of Document-at-a-Time and Score-at-a-Time query evaluation. In *Proc. ACM Int. Conf. on Web Search and Data Mining (WSDM)*, pages 201–210, 2017.

J. Mackenzie. Managing tail latencies in large scale IR systems. In *Proc. ACM Int. Conf. on Research and Development in Information Retrieval (SIGIR)*, page 1369, 2017.

J. Mackenzie, F. Scholer, and J. S. Culpepper. Early termination heuristics for Score-at-a-Time index traversal. In *Proc. Australasian Document Computing Symp. (ADCS)*, pages 8.1–8.8, 2017.

J. Mackenzie, J. S. Culpepper, R. Blanco, M. Crane, C. L. A. Clarke, and J. Lin. Query driven algorithm selection in early stage retrieval. In *Proc. ACM Int. Conf. on Web Search and Data Mining (WSDM)*, pages 396–404, 2018.

R. Benham, J. Mackenzie, A. Moffat, and J. S. Culpepper. Boosting search performance using query variations. *ACM Trans. on Information Systems*, 37(4):41.1–41.25, 2019.