Efficient Query Processing Infrastructures

A half-day tutorial at SIGIR 2018

Nicola Tonellotto
ISTI-CNR
Pisa, Italy
nicola.tonellotto@isti.cnr.it

Craig Macdonald
University of Glasgow
Glasgow, Scotland, UK
craig.macdonald@glasgow.ac.uk

ABSTRACT

Typically, techniques that benefit effectiveness of information retrieval (IR) systems have a negative impact on efficiency. Yet, with the large scale of Web search engines, there is a need to deploy efficient query processing techniques to reduce the cost of the infrastructure required. This tutorial aims to provide a detailed overview of the infrastructure of an IR system devoted to the *efficient yet effective processing of user queries*. This tutorial guides the attendees through the main ideas, approaches and algorithms developed in the last 30 years in query processing. In particular, we illustrate, with detailed examples and simplified pseudo-code, the most important query processing strategies adopted in major search engines, with a particular focus on dynamic pruning techniques. Moreover, we present and discuss the state-of-the-art innovations in query processing, such as impact-sorted and blockmax indexes. We also describe how modern search engines exploit such algorithms with learning-to-rank (LtR) models to produce effective results, exploiting new approaches in LtR query processing. Finally, this tutorial introduces query efficiency predictors for dynamic pruning, and discusses their main applications to scheduling, routing, selective processing and parallelisation of query processing, as deployed by a major search engine.

ACM Reference Format:
Nicola Tonellotto and Craig Macdonald. 2018. Efficient Query Processing Infrastructures: A half-day tutorial at SIGIR 2018. In *SIGIR '18: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, July 8–12, 2018, Ann Arbor, MI, USA.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3209978.3210191

INTENDED AUDIENCE

This tutorial is targeted to SIGIR attendees with at least an introductory knowledge in IR or IR-related tasks (e.g., databases, data mining). In particular, the tutorial is of utmost interest to PhD students, researchers and practitioners following a research path on efficiency and infrastructures in IR and Web search. Indeed, anyone working on search and ranking on big data will benefit from this tutorial. Finally, the tutorial is also well suited to lecturers looking for clear and concise examples on state-of-the-art query processing techniques to include in their university IR-related teaching course.

INSTRUCTORS

Dr. Nicola Tonellotto (m) (http://hpc.isti.cnr.it/~khast/) is a Researcher within the High Performance Computing Lab at Inforation Science and Technologies Institute of the National Research Council of Italy. His main research interests include cloud computing and information retrieval, focusing on efficiency aspects of query processing and resource management. Nicola has co-authored more than 60 papers on these topics in peer reviewed international journal and conferences. He lectures on Computer Architectures for BSc students for four years and Distributed Enabling Platforms for MSc students for ten years at the University of Pisa. He co-presented a tutorial on the same topic as this tutorial at ECIR 2017. He was co-recipient of the ACM's SIGIR 2015 Best Paper Award for the paper entitled "QuickScorer: a Fast Algorithm to Rank Documents with Additive Ensembles of Regression Trees".

Dr. Craig Macdonald (m) (http://www.dcs.gla.ac.uk/~craigm/ is a Lecturer within the Information Retrieval Group at the University of Glasgow. He has co-authored 190 publications in information retrieval, including on efficient query processing, as well as on the practical deployments of learning-to-rank approaches. He was co-recipient of the ECIR 2014 Best Paper Award for the paper entitled "On Inverted Index Compression for Search Engine Efficiency". Craig has been joint coordinator of the TREC Blog, Microblog and Web tracks, is lead maintainer of the Terrier.org information retrieval platform, and jointly presented a tutorial at ECIR 2008 entitled "Researching and building IR applications using Terrier", and provided input to the CIKM 2011 tutorial entitled "Large-scale Information Retrieval Experimentation with Terrier". He presented a session on Information Retrieval Infrastructures at the 2015 edition of the European Summer School on Information Retrieval, and co-presented a tutorial on the same topic as this tutorial at ECIR 2017. He lectures on Database Systems, Text Analytics and Information Retrieval at the University of Glasgow, across cohorts from 1st year- to BSc/Master-level.

SIGIR '18, July 8–12, 2018, Ann Arbor, MI, USA

ACM ISBN 978-1-4503-5657-2/18/07.
https://doi.org/10.1145/3209978.3210191

EXTENDED ABSTRACT

1 MOTIVATION

Within large IR systems such as Web search engines, there is a need to constantly deal with two main problems: (1) the rapid and continuous growth of the number of Web pages and (2) the requirement of having sub-second response times. Even if architectural and engineering advances are tremendously improving the responsiveness of such systems, latency has still a major impact on IR systems users. Amazon has reported that every 100 ms of latency results in a 1% drop in sales. Similarly Google has reported that an extra 0.5 seconds in search page generation time resulted in a 20% drop in traffic. Moreover, almost all information retrieval techniques that increase effectiveness (e.g. query rewriting/expansion; learning-to-rank) do so at the loss of efficiency. Effectiveness is of course the other key requirement for any IR system, forming a natural trade-off with efficiency.

To reduce the cost of deploying such techniques (e.g. increasing hardware costs), the efficiency of the overall search engine is a key concern for operational reasons. As a consequence, several changes and optimizations have been introduced in recent years to cope with these two problems, without negatively impacting the quality of the results returned to users for their queries.

The aim of the tutorial is two-fold. Firstly, we aim at providing a detailed overview of the state-of-the-art query processing techniques adopted in Web information retrieval along with clear examples that can help attendees to understand and implement basic and advanced solutions for query processing in research frameworks and how such techniques have been adopted and employed in real-world search engines. Secondly we aim at providing attendees with the knowledge of complete multi-stage query processing pipeline composed by ranked retrieval and learning-to-rank stages and how their interactions affect both efficiency and effectiveness of the IR systems as a whole. We expect that attendees obtain a thorough knowledge about implementing, deploying and analysing efficient yet effective learning-to-rank solutions in IR systems.

2 TUTORIAL INTENDED LEARNING OUTCOMES

The overall goal of this tutorial is to give attendees a detailed overview of the architecture of an IR system devoted to the *efficient yet effective processing of user queries.* The tutorial guides the attendees through the main ideas, approaches and algorithms developed in the last 30 years in query processing, and, in particular, tutorial attendees attain the following intended learning outcomes:

ILO 1. learn about the fundamentals of query processing in information retrieval in terms of the data structures and algorithms involved (e.g. Document-at-a-Time/Term-at-a-time/Score-at-a-time);

ILO 2. learn about the state-of-the-art of dynamic pruning (query processing) techniques (e.g., TAAT optimizations, impacts, MaxScore, WAND, blockmax indexes, top-hitlists), illustrated through extensive use of pseudo-code and examples.

ILO 3. understand how such query processing fits into a multi-tier search architecture, involving learning-to-rank and distributed query processing, including the efficient application of state-of-the-art learning-to-rank models.

ILO 4. understand about *safeness* in dynamic pruning techniques, and about the nature of the tradeoffs between efficiency and effectiveness in query processing.

ILO 5. learn about query efficiency prediction and its applications to efficient yet effective web search and to operational costs reduction, including a description of Bing's use of QEP.

Other tutorials cover material outwith the scope of this tutorial, or in varying level of detail. The "Scalability and Efficiency Challenges in Large-Scale Web" tutorial by B. Barla Cambazoglu, Ricardo Baeza-Yates, which has been presented several times over the last few years (e.g., SIGIR 2014, CIKM 2015, SIGIR 2016), portrays a wider viewpoint on efficiency in large-scale distributed information retrieval, covering a wider range of infrastructure topics, from crawling & indexing to query processing in distributed settings. In contrast, this tutorial is more focussed on the core query processing data structures and algorithms in further detail, which allows attendees understanding of the algorithms and their limitations, and assist anyone planning their implementation. Moreover, the tutorial covers new material (across ILO 2, 3, 4 & 5) that are not covered by Cambazoglu & Baeza-Yates; the SIGIR 2016 "Succinct Data Structures in Information Retrieval: Theory and Practice" tutorial by Simon Gog and Rossano Venturini is a more focussed examination of posting list compression, which this tutorial only touches upon within ILO 1.

For these reasons, we believe that a tutorial on efficient query processing – giving insights into new techniques such as BlockMaxWAND and query efficiency prediction — is both timely and an appropriate addition to SIGIR 2018, and nicely complements the contents of those two previous tutorials.

3 TUTORIAL HISTORY

This is the second edition of this tutorial, presented for the first time at ECIR 2017, with 25 attendees. Moreover, it builds upon a long-standing collaboration between the presenters and their host institutions. The collaboration was initiated in 2008 thanks to a research funding grant by Royal Society (UK), and has to-date resulted in 15 publications covering the areas of efficiency and Green IR, across venues such as SIGIR, WSDM, CIKM, ICTIR & TOIS. Three graduated PhD students have benefitted from this collaboration.

The tutorial leverages material on IR infrastructures presented at the European Summer School in Information Retrieval (ESSIR) 2015 (51 participants), and a cut-down version presented to the BSc/Master-level IR course taught at the University of Glasgow (80 students).

4 FULL DESCRIPTION OF TOPICS

In this tutorial we tackle the "foundations" and "recent" aspects of efficient yet effective query processing both from the point of view of the data structures and algorithms involved, and of the infrastructure support required to execute such algorithms on textual collections, with a focus on some industry-level approaches. In particular, the tutorial presents and discuss the following topics:

- A general view and referenced description of the system components and data structures involved in query processing (e.g., inverted index, ranking models, compression) and basic query processing algorithms: boolean vs. ranked retrieval, conjunctive vs. disjunctive processing, Term-at-a-time (TAAT) vs. Document-at-a-time (DAAT) [9, 10, 14, 31, 35, 41].
- The most effective optimization techniques introduced in TAAT and DAAT strategies to improve query processing efficiency: (1) techniques aiming at dynamically skipping the scoring of documents stored in the inverted index that have a low chance to make the top final results (e.g., TAAT optimizations, MaxScore, WAND and their variants) [8, 9, 17, 21, 30, 31, 34, 35], (2) an alternative organization of the inverted index, where the documents in the posting lists are sorted not according to document identifiers, but according to some measure of their contributions to the relevance score of the documents to user queries (i.e., impacts), and some special query processing algorithms to deal with these new indexes [1–5, 32], (3) the most recent index organization based on block max-scores, leveraging a measure of the contribution of posting list portions to the relevance of their documents to user queries, without altering the inverted index, but by introducing a new component, and improved query processing algorithms exploiting this new index component (e.g., BlockMaxWAND and its variants) [12, 15, 16, 29, 33]. Moreover, we discuss how some query processing techniques (such as MaxScore and WAND) can be configured to have no negative impact on effectiveness (also known as safe-to-rank-*K*), or to potentially trade effectiveness to attain efficiency gains.
- Many IR systems work in a cascading manner: a ranking of documents is iteratively truncated at a rank cutoff, and then re-ranked by the application of a more refined ranking technique. Later stages are based upon learning-to-rank. We illustrate how this is typically deployed and applied, including: (1) which features and models are used in learning-to-rank in IR [7, 23, 26, 27], and (2) how such models are efficiently applied in query processing (e.g., Struct+, PRED, VPRED, Quickscorer, selective pruning) [6, 11, 22, 25, 36, 38, 39].
- Query processing techniques such as MaxScore and WAND can prune the scoring of postings, such as the time they take to execute a query is not just dependent on the length of the query term's postings lists. We introduce query efficiency prediction (QEP) techniques, which predict the duration of a query before it is executed. QEP techniques have a number of applications: For instance, arriving queries can be routed within a distributed setup to the query server with the shortest queue duration; We describe applications of QEP, namely: (1) the selective adjustment of dynamic pruning safeness and (2) selectively applying multiple threads to the execution of queries that are predicted to be slow – as performed by the Bing search engine. In doing so, a significant gain in server capacity can be achieved, resulting in financial and/or energy savings [18–20, 24, 37, 40].

In particular, we highlight the updating of the ECIR 2017 tutorial content to encompass recently published material, including VariableBMW [29], inclusion of efficiency-aware query rewriting into the query processing pipeline [28], and cost-sensitive learning-to-rank techniques [13].

5 TUTORIAL FORMAT

This is a half-day format tutorial. Indeed, experience by the co-authors in recent organising large conferences (CIKM 2011, SIGIR 2016) found a growing preference among both attendees and tutorial organisers in using the half-day format, which allows for a short, focussed session of ~130 slides, without tiring out the attendees.

In terms of content,for structure purposes, we list an approximate number of slides for each topic of the tutorial:

1. **Introduction (15 slides): Motivations for efficient query processing**
2. **Index Structures & Compression (18 slides)** – ILO 1
   - 3 slides: Data Structures
   - 10 slides: Compression
   - 3 slides: Skipping
3. **Query Evaluation inc. Dynamic Pruning (55 slides)** – ILOs 1, 2 & 4
   - 7 slides: Basic Query Processing: DAAT vs. TAAT;
   - 12 slides: Dynamic Pruning concepts
   - 15 slides: DAAT MaxScore & WAND; DAAT Safeness
   - 8 slides: Blockmax indexes, BMW & VBMW
   - 8 slides: Impact-sorted indexes
4. **Cascading & Learning Infrastructure (20 slides)** – ILO 3
   - 8 slides: architectural design, LTR features
   - 6 slides: types of LTR models
   - 5 slides: efficiency/effectiveness tradeoffs in LtR
   - 10 slides: LtR and query processing: early-exit strategies, VPRED, Quickscorer
5. **Efficiency Prediction and Applications (20 slides)** – ILO 4 & 5
   - 6 slides: QEP Predictors
   - 4 slides: Application – Selective Pruning
   - 3 slides: Application – Selective Rewriting
   - 4 slides: Application – Distributed query scheduling
   - 6 slides: Application – Selective Parallelisation, as deployed by Bing.
6. **Wrap-Up & Reference List (5 slides)**

REFERENCES

[1] Anh, V.N., de Kretser, O., Moffat, A.: Vector-space ranking with effective early termination. In Proc. SIGIR (2001) 35–42

[2] Anh, V.N., Moffat, A.: Impact transformation: effective and efficient web retrieval. In Proc. SIGIR (2002) 3–10

[3] Anh, V.N., Moffat, A.: Simplified similarity scoring using term ranks. In Proc. SIGIR (2005) 226–233

[4] Anh, V.N., Moffat, A.: Pruning strategies for mixed-mode querying. In Proc. CIKM (2006) 190–197

[5] Anh, V.N., Moffat, A.: Pruned query evaluation using pre-computed impacts. In Proc. SIGIR (2006) 372–379

[6] Asadi, N., Lin, J., de Vries, A.P.: Runtime optimizations for tree-based machine learning models. IEEE Trans. Knowl. Data Eng. **26**(9) (2014) 2281–2292

[7] Bendersky, M., Croft, W.B., Diao, Y.: Quality-biased ranking of web documents. In Proc. WSDM (2011) 95–104

[8] Broder, A.Z., Carmel, D., Herscovici, M., Soffer, A., Zien, J.: Efficient query evaluation using a two-level retrieval process. In Proc. CIKM (2003) 426–434

[9] Buckley, C., Lewit, A.F.: Optimization of inverted vector searches. In Proc. SIGIR (1985) 97–110

[10] Cambazoglu, B.B., Baeza-Yates, R.A.: Scalability Challenges in Web Search Engines. Synthesis Lectures on Information Concepts, Retrieval, and Services. Morgan & Claypool Publishers (2015)

[11] Cambazoglu, B.B., Zaragoza, H., Chapelle, O., Chen, J., Liao, C., Zheng, Z., Degenhardt, J.: Early exit optimizations for additive machine learned ranking systems. In Proc. WSDM (2010) 411–420

[12] Chakrabarti, K., Chaudhuri, S., Ganti, V.: Interval-based pruning for top-k processing over compressed lists. In Proc. ICDE (2011) 709–720

[13] Chen, R.-C., Gallagher, L., Blanco, R., Culpepper, J.S.: Efficient cost-aware cascade ranking in multi-stage retrieval. In Proc. SIGIR (2017) 445–454

[14] Culpepper, J.S., Moffat, A.: Efficient set intersection for inverted indexing. ACM Trans. Inf. Syst. **29**(1) (2010) 1:1–1:25

[15] Dimopoulos, C., Nepomnyachiy, S., Suel, T.: Optimizing top-k document retrieval strategies for block-max indexes. In Proc. WSDM (2013) 113–122

[16] Ding, S., Suel, T.: Faster top-k document retrieval using block-max indexes. In Proc. SIGIR (2011) 993–1002

[17] Fontoura, M., Josifovski, V., Liu, J., Venkatesan, S., Zhu, X., Zien, J.: Evaluation strategies for top-k queries over memory-resident inverted indexes. In Proc. VLDB **4**(12) (2011) 1213–1224

[18] Freire, A., Macdonald, C., Tonellotto, N., Ounis, I., Cacheda, F.: A self-adapting latency/power tradeoff model for replicated search engines. In Proc. WSDM (2014) 13–22

[19] Jeon, M., Kim, S., Hwang, S.w., He, Y., Elnikety, S., Cox, A.L., Rixner, S.: Predictive parallelization: Taming tail latencies in web search. In Proc. SIGIR (2014) 253–262

[20] Kim, S., He, Y., Hwang, S.w., Elnikety, S., Choi, S.: Delayed-dynamic-selective (dds) prediction for reducing extreme tail latency in web search. In Proc. SIGIR (2015) 7–16

[21] Lucarella, D.: A document retrieval system based on nearest neighbour searching. Journal of Information Science **14**(1) (1988) 25–33

[22] Lucchese, C., Nardini, F.M., Orlando, S., Perego, R., Tonellotto, N., Venturini, R.: Quickscorer: A fast algorithm to rank documents with additive ensembles of regression trees. In Proc. SIGIR (2015) 73–82

[23] Lucchese, C., Nardini, F.M., Orlando, S., Perego, R., Tonellotto, N.: Speeding up document ranking with rank-based features. In Proc. SIGIR (2015) 895–898

[24] Macdonald, C., Tonellotto, N., Ounis, I.: Learning to predict response times for online query scheduling. In Proc. SIGIR (2012) 621–630

[25] Macdonald, C., Santos, R.L.T., Ounis, I.: The whens and hows of learning to rank for web search. Information Retrieval **16**(5) (2012) 584–628

[26] Macdonald, C., Santos, R.L., Ounis, I.: On the usefulness of query features for learning to rank. In Proc. CIKM (2012) 2559–2562

[27] Macdonald, C., Santos, R.L., Ounis, I., He, B.: About learning models with multiple query-dependent features. ACM Trans. Inf. Syst. **31**(3) (2013) 11:1–11:39

[28] Macdonald, C., Tonellotto, N., Ounis, I.: Efficient & Effective Selective Query Rewriting with Efficiency Predictions. In Proc. SIGIR (2017) 495–504

[29] Mallia, A., Ottaviano, G., Porciani, E., Tonellotto, N., Venturini, R.: Faster Block-Max WAND with Variable-sized Blocks. In Proc. SIGIR (2017) 625–634

[30] Moffat, A., Zobel, J.: Fast ranking in limited space. In Proc. ICDE (1994) 428–437

[31] Moffat, A., Zobel, J.: Self-indexing inverted files for fast text retrieval. ACM Trans. Inf. Syst. **14**(4) (1996) 349–379

[32] Persin, M.: Document filtering for fast ranking. In Proc. SIGIR (1994) 339–348

[33] Shan, D., Ding, S., He, J., Yan, H., Li, X.: Optimized top-k processing with global page scores on block-max indexes. In Proc. WSDM (2012) 423–432

[34] Strohman, T., Turtle, H., Croft, W.B.: Optimization strategies for complex queries. In Proc. SIGIR (2005) 219–225

[35] Turtle, H., Flood, J.: Query evaluation: strategies and optimizations. Information Processing and Management **31**(6) (1995) 831–850

[36] Tang, X., Jin, X., Yang, T.: Cache-conscious runtime optimization for ranking ensembles. In Proc. SIGIR (2014) 1123–1126

[37] Tonellotto, N., Macdonald, C., Ounis, I.: Efficient and effective retrieval using selective pruning. In Proc. WSDM (2013) 63–72

[38] Wang, L., Lin, J., Metzler, D.: Learning to efficiently rank. In Proc. SIGIR (2010) 138–145

[39] Wang, L., Lin, J., Metzler, D.: A cascade ranking model for efficient ranked retrieval. In Proc. SIGIR (2011) 105–114

[40] Wu, H., Fang, H.: Analytical performance modeling for top-k query processing. In Proc. SIGIR (2014) 1619–1628

[41] Zobel, J., Moffat, A.: Inverted files for text search engines. ACM Computing Surveys **38**(2) (2006)