An Experimental Study of Index Compression and DAAT Query Processing Methods

Antonio Mallia **Michał Siedlaczek** **Torsten Suel**

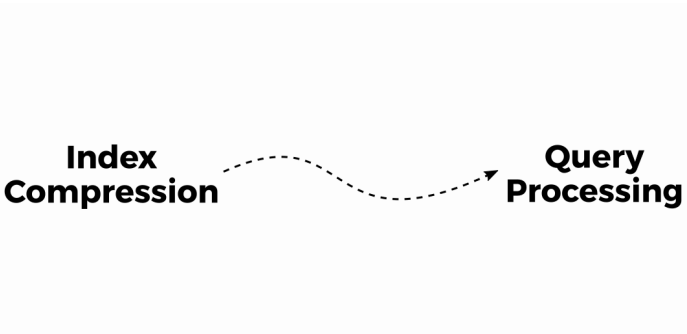
Department of Computer Science and Engineering
Tandon School of Engineering
New York University

April 16th, 2019

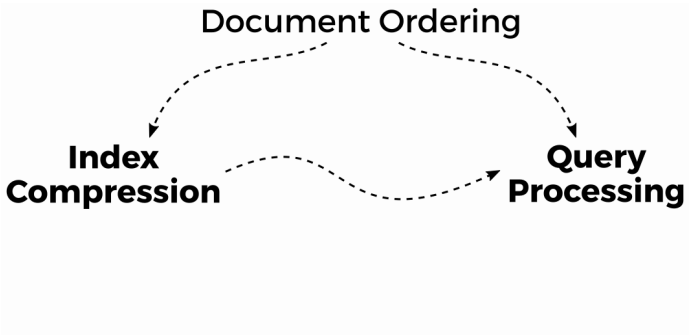
Motivations

Interacting Components

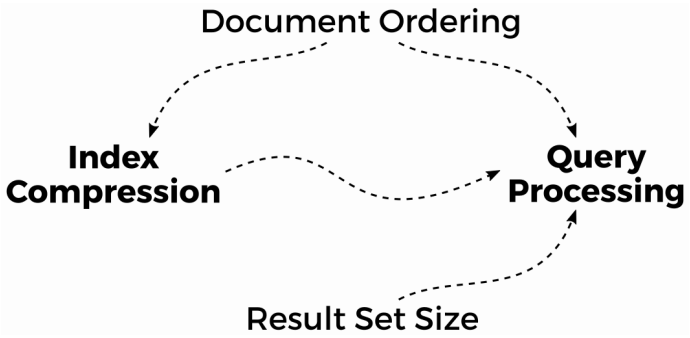
Interacting Components



Interacting Components



Interacting Components



Main Contributions

- Confirmed some established results
- New important insights
- Modern and generic code base

Source Code

`https://github.com/pisa-engine/pisa`

Main Contributions

- Confirmed some established results
- New important insights
- Modern and generic code base

Source Code

`https://github.com/pisa-engine/pisa`

Main Contributions

NYU

- Confirmed some established results
- New important insights
- Modern and generic code base

Source Code

`https://github.com/pisa-engine/pisa`

Index Compression

NYU

- Variable Byte Methods:
  - **VarintGB** [Dean 2009]
  - **Varint-G8IU** [Stepanov et al. 2011]
  - **StreamVByte** [Lemire et al. 2018]
- Word-Aligned Methods:
  - **Simple16** [Zhang et al. 2008]
  - **Simple8b** [Anh and Moffat 2010]
  - **SIMD-BP128** [Lemire and Boytsov 2015]
  - **QMX** [Trotman and Lin 2016]
- **OptPForDelta** [Yan et al. 2009]
- **Partitioned Elias-Fano** [Ottaviano and Venturini 2014]
- **Binary Interpolative** [Moffat and Stuiver 2000]
- **Asymetric Numeral Systems** [Moffat and Petri 2018]

Query Processing Algorithms

NYU

Top-k disjunctive Document-at-a-Time query processing algorithms with safe early-termination.

- **MaxScore** [Turtle and Flood 1995]
- **WAND** [Broder et al. 2003]
- **Block-Max MaxScore** [Chakrabarti et al. 2011]
- **Block-Max WAND** [Ding and Suel 2011]
- **Variable Block-Max WAND** [Mallia et al. 2017]

Document Ordering

NYU

The impact that document ID assignment has on index compression and query efficiency.

- **Random** – baseline
- **URL** [Silvestri 2007]
- **Recursive Graph Bisection (BP)** [Dhulipala et al. 2016]

Result Set Size

NYU

- Typically small k in past top-k search studies
- Can be significantly larger for candidate retrieval for cascade ranking
- Recently shown that large k slow down retrieval [Crane et al. 2017]
- Thus, we experiment with values of k between 10 and 10,000

Experimental Setup

Implementation

NYU

Source Code

- https://github.com/pisa-engine/pisa
- Fork of ds2i: https://github.com/ot/ds2i

Third Party Libraries

- https://github.com/lemire/FastPFor
- https://github.com/andrewtrotman/JASSv2
- https://github.com/mpetri/partitioned_ef_ans

Testing Environment

NYU

- Implemented in C++17 and compiled with GCC 7.3 on highest optimization level
- Intel Core i7-4770 quad-core 3.40GHz CPU
- Haswell micro architecture supporting AVX2 instruction set
- CPUs L1, L2, and L3 cache sizes are 32KB, 256KB, and 8MB, respectively
- 32GiB RAM

Data Sets

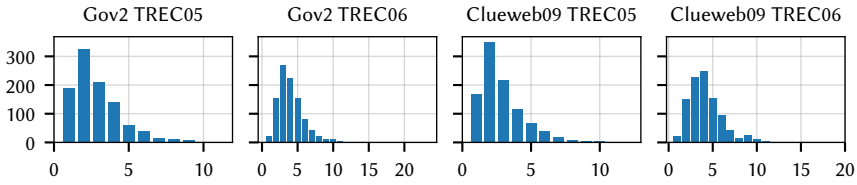
NYU

	Documents	Terms	Postings
GOV2	24,622,347	35,636,425	5,742,630,292
Clueweb09B	50,131,015	92,094,694	15,857,983,641

- HTML content parsed with Apache Tika
- Words stemmed with Porter2
- Stopwords kept

Queries

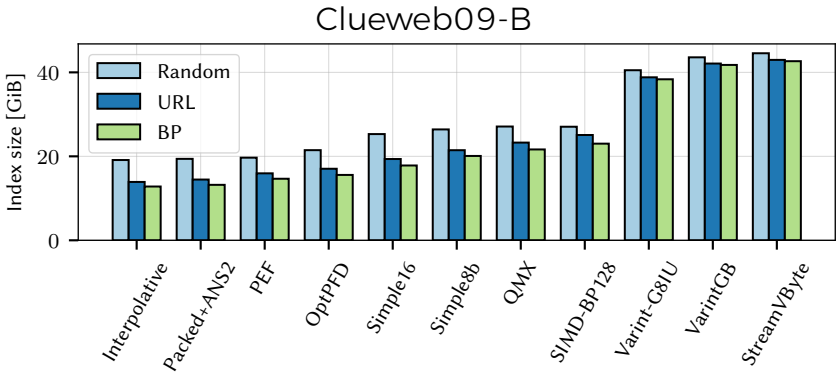
- TREC 2005 TREC 2006 from Terabyte Track Efficiency Task
- Queries with non-existent terms removed
- Initially sampled 1,000 queries for each query set and collection



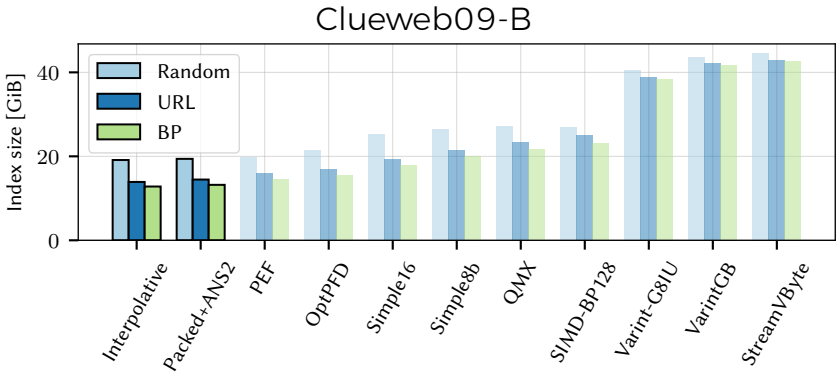
- Further sampled 1,000 queries for each query length from 2 to 6+

Results and Discussion

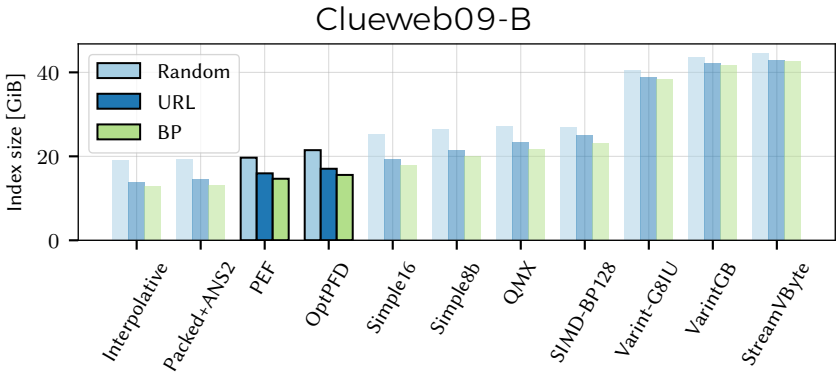
Compression



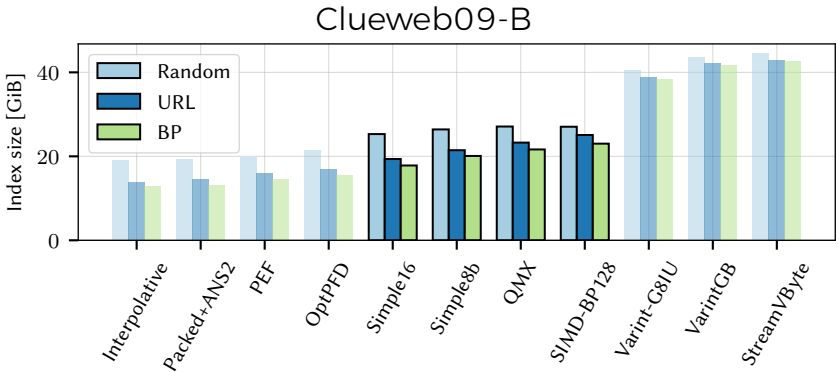
Compression



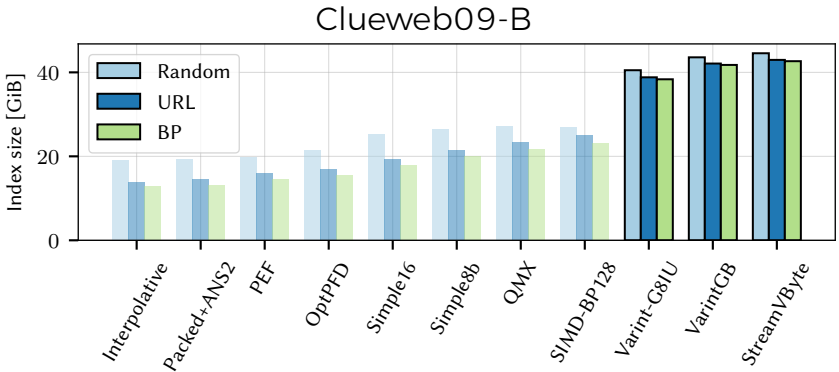
Compression



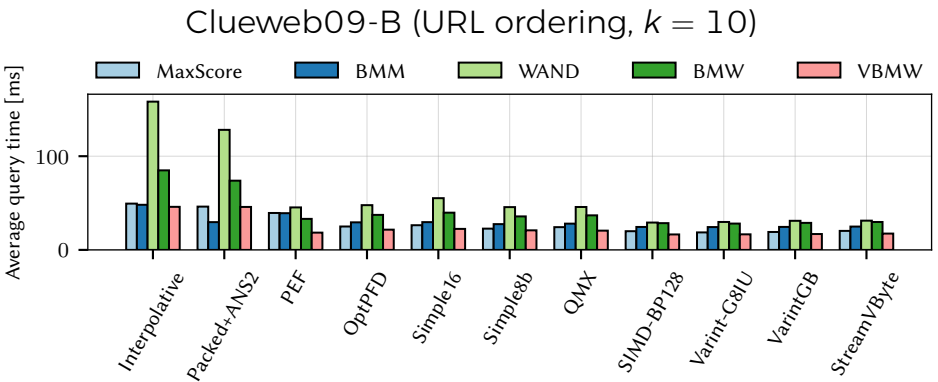
Compression



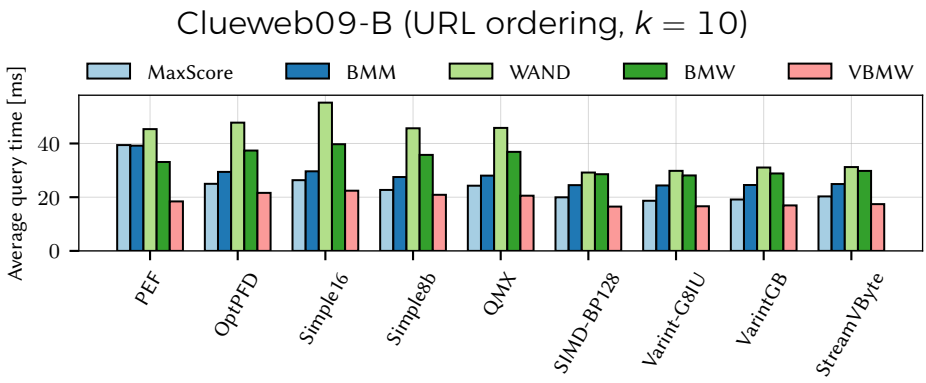
Compression



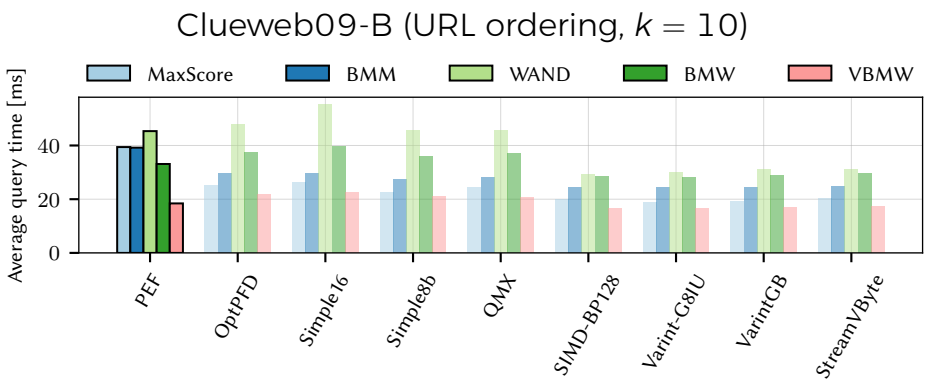
Query Speed


Clueweb09-B (URL ordering, $k = 10$)
MaxScore
BMM
WAND
BMW
VBMW
Average query time [ms]
100
0
Interpolative
Packed+ANS2
PEF
OptPFD
Simple16
Simple8b
QMX
SIMD-BP128
Varint-G8IU
VarintGB
StreamVByte

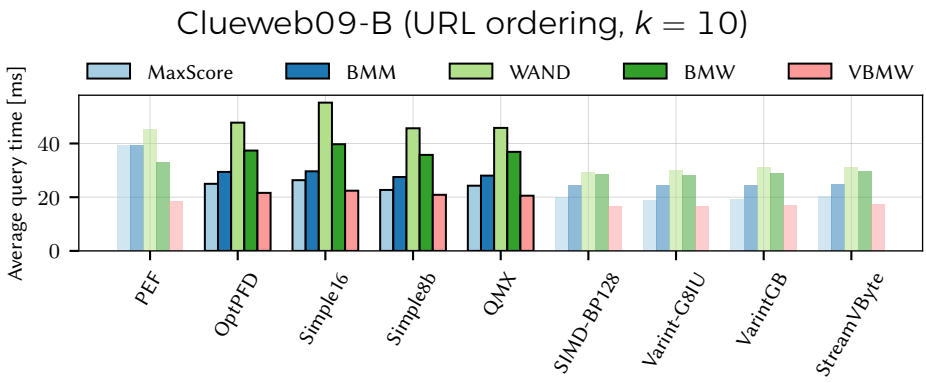
Query Speed


Clueweb09-B (URL ordering, $k = 10$)
MaxScore
BMM
WAND
BMW
VBMW
Average query time [ms]
0
20
40
PEF
OptPFD
Simple16
Simple8b
QMX
SIMD-BP128
Varint-G8IU
VarintGB
StreamVByte

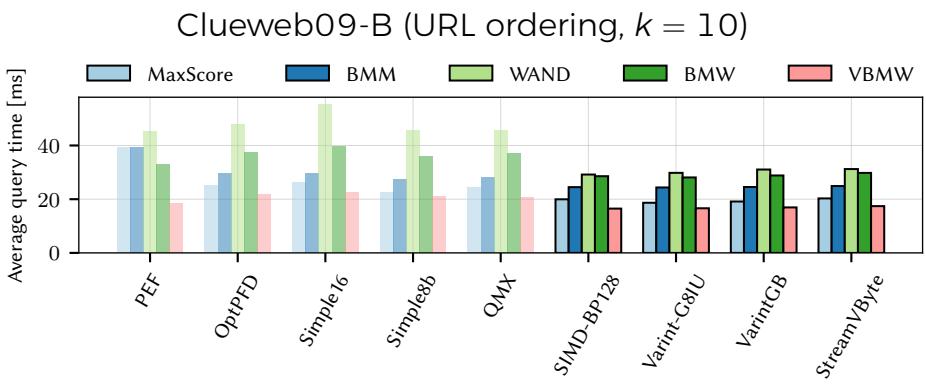
Query Speed


Clueweb09-B (URL ordering, $k = 10$)
MaxScore
BMM
WAND
BMW
VBMW
Average query time [ms]
0
20
40
PEF
OptPFD
Simple16
Simple8b
QMX
SIMD-BP128
Varint-G8IU
VarintGB
StreamVByte

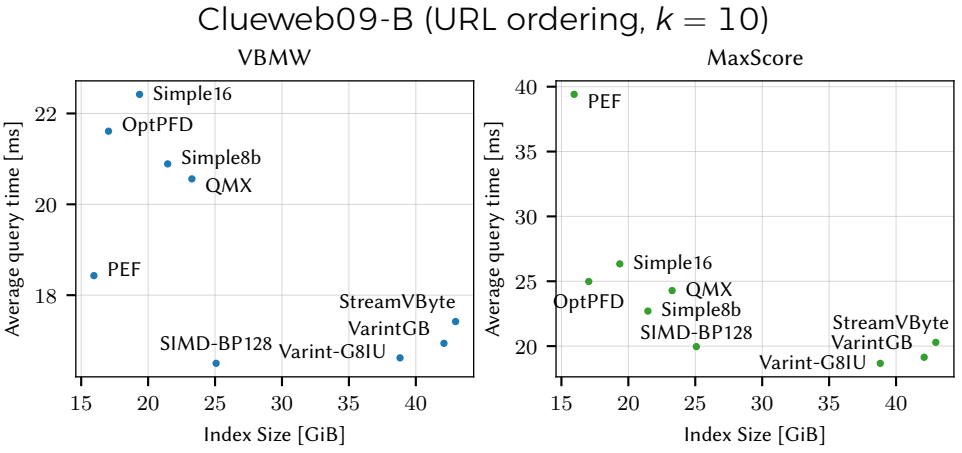
Query Speed


Clueweb09-B (URL ordering, $k = 10$)
MaxScore
BMM
WAND
BMW
VBMW
Average query time [ms]
0
20
40
PEF
OptPFD
Simple16
Simple8b
QMX
SIMD-BP128
Varint-G8IU
VarintGB
StreamVByte

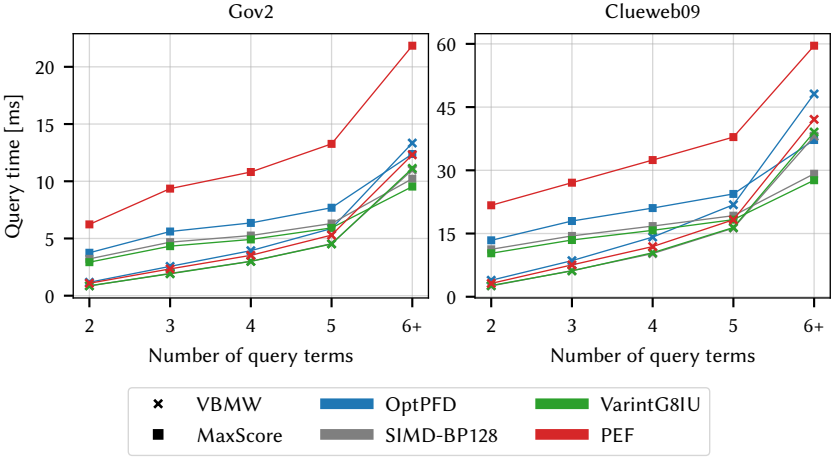
Query Speed


Clueweb09-B (URL ordering, $k = 10$)
MaxScore
BMM
WAND
BMW
VBMW
Average query time [ms]
0
20
40
PEF
OptPFD
Simple16
Simple8b
QMX
SIMD-BP128
Varint-G8IU
VarintGB
StreamVByte

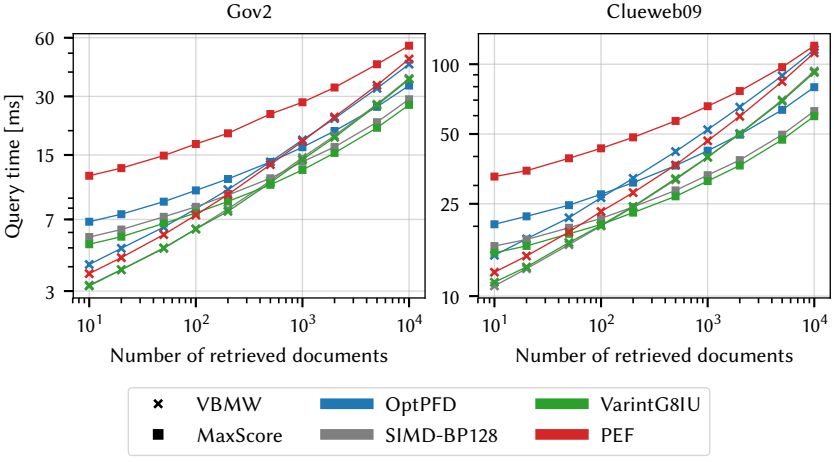
Query Speed v. Index Size



Query Length



Result Set Size



Conclusions

NYU

- Clear trade-off between speed and size
- Interesting compression insights
  - SIMD-BP128 matches speed of Varint methods while improving compression ratio
  - PEF's speed competitive when using VBMW
- Significant slowdown for large k
- MaxScore competitive with VBMW under certain circumstances
- Recursive Graph Bisection improves both compression and speed over URL ordering

Thank you for your time.

Any questions?

References I


Anh, V. N. and Moffat, A. (2010).
Index compression using 64-bit words.
Software: Practice and Experience, 40(2):131–147.


Broder, A. Z., Carmel, D., Herscovici, M., Soffer, A., and Zien, J. (2003).
Efficient query evaluation using a two-level retrieval process.
In *Proc. of the 12th Intl. Conf. on Information and Knowledge Management*, pages 426–434.


Chakrabarti, K., Chaudhuri, S., and Ganti, V. (2011).
Interval-based pruning for top-k processing over compressed lists.
In *Proc. of the 2011 IEEE 27th Intl. Conf. on Data Engineering*, pages 709–720.


Crane, M., Culpepper, J. S., Lin, J., Mackenzie, J., and Trotman, A. (2017).
A comparison of document-at-a-time and score-at-a-time query evaluation.
In *Proc. of the 10th ACM Intl. Conf. on Web Search and Data Mining*, pages 201–210.


Dean, J. (2009).
Challenges in building large-scale information retrieval systems: invited talk.
In *Proc. of the 2nd ACM Intl. Conf. on Web Search and Data Mining*, pages 1–1.


Dhulipala, L., Kabiljo, I., Karrer, B., Ottaviano, G., Pupyrev, S., and Shalita, A. (2016).
Compressing graphs and indexes with recursive graph bisection.
In *Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1535–1544.

References II

Ding, S. and Suel, T. (2011).
Faster top-k document retrieval using block-max indexes.
In *Proc. of the 34th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 993–1002.

Lemire, D. and Boytsov, L. (2015).
Decoding billions of integers per second through vectorization.
Softw. Pract. Exper., 45(1):1–29.

Lemire, D., Kurz, N., and Rupp, C. (2018).
Stream vbyte: Faster byte-oriented integer compression.
Information Processing Letters, 130:1–6.

Mallia, A., Ottaviano, G., Porciani, E., Tonellotto, N., and Venturini, R. (2017).
Faster blockmax WAND with variable-sized blocks.
In *Proc. of the 40th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 625–634.

Moffat, A. and Petri, M. (2018).
Index compression using byte-aligned ANS coding and two-dimensional contexts.
In *Proc. of the 11th ACM Intl. Conf. on Web Search and Data Mining*, pages 405–413.

Moffat, A. and Stuiver, L. (2000).
Binary interpolative coding for effective index compression.
Inf. Retr., 3(1):25–47.

References III


Ottaviano, G. and Venturini, R. (2014).
Partitioned elias-fano indexes.
In *Proc. of the 37th Ann. Intl. ACM SIGIR Conf. on Research and Development in Information Retrieval*, pages 273–282.


Silvestri, F. (2007).
Sorting out the document identifier assignment problem.
In *Proc. of the 29th European Conf. on IR Research*, pages 101–112.


Stepanov, A. A., Gangolli, A. R., Rose, D. E., Ernst, R. J., and Oberoi, P. S. (2011).
SIMD-based decoding of posting lists.
In *Proc. of the 20th Intl. Conf. on Information and Knowledge Management*, pages 317–326.


Trotman, A. and Lin, J. (2016).
In vacuo and in situ evaluation of SIMD codecs.
In *Proc. of the 21st Australasian Document Computing Symposium*, pages 1–8.


Turtle, H. and Flood, J. (1995).
Query evaluation: Strategies and optimizations.
Information Processing & Management, 31(6):831–850.


Yan, H., Ding, S., and Suel, T. (2009).
Inverted index compression and query processing with optimized document ordering.
In *Proc. of the 18th Intl. Conf. on World Wide Web*, pages 401–410.

References IV


Zhang, J., Long, X., and Suel, T. (2008).
Performance of compressed inverted list caching in search engines.
In *Proc. of the 17th Intl. Conf. on World Wide Web*, pages 387–396.