Faster WAH Compression Querying through the Use of Metadata

Miguel Velez and Jason Sawin

University of St. Thomas. St. Paul, MN

College of Arts and Sciences
Computer & Information Sciences

Bitmaps

- Matrix data structure made up of bit arrays
- Coarse approximation of data
- A record has an attribute if its value is 1

Students	A	B	C	D	F	Master's	Ph.D.
Bob Johnson	0	0	0	1	0	1	0
Katie Morrison	0	1	0	0	0	0	1
John Taylor	0	0	1	0	0	1	0
Jackie Williams	1	0	0	0	0	0	1

- Can be compressed for faster querying

WAH

- Word- Aligned Hybrid code [1] encodes runs into a single word

Uncompressed bitmap (1426 bits)

1001101010…101101101 | 1111111111…1111111111111111

31 bit literal | $31 \cdot 45$ (45 runs)

WAH compressed bitmap (64 bits)

01001101010…101101101 **11**00000000…00000000101101

32 | 32

Flag bit | Fill bit | Flag bit | Length

- WAH provides high compression ratio and rapid query processing

Querying

SELECT * **FROM** Students **WHERE** Degree='Ph.D.' **AND** Grade='B';

B		Ph.D.		Result
0		0		0
1	&	1	=	1
0		0		0
0		1		0

Leverage bitwise operations (NOT, AND, OR, XOR), which are atomic and fast, to obtain fast querying speeds [2]

Can take advantage of certain data configurations to obtain even faster querying executions [3].

Verbose

- Motivation: Jump as many words in the queried columns as possible
- Encode the length of a run and the number of literals following the run
- General approach that does not alter the compression scheme

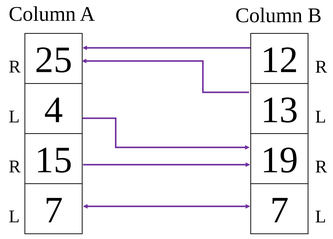
Bitmap column

run length 11
literal
literal
literal
run length 3
literal

→ Metadata

11
3
3
1

- Example



Do not have to read certain words since they fit into the currently read word

- Improvement over regular query
- Complicated algorithm
- Noticed in general that the word with the run had to be read to check its value

Succinct

- Motivation: Encode necessary information
- Encode the number of literals following a run

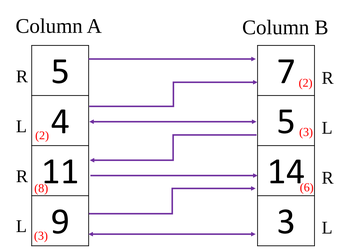
Bitmap column

run length 11
literal
literal
literal
run length 3
literal

→ Metadata

3
1

Example

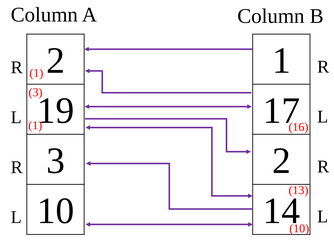


More realistic scenario where knowing the number of runs is not beneficial

- Improvement over regular query and Verbose
- Simpler algorithm
- Reduced size of metadata by 50%

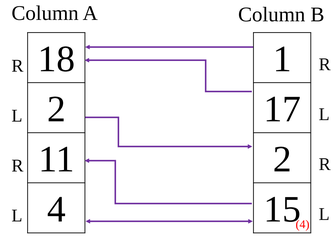
Heuristic

- Theoretically, columns that have many runs queried with columns containing many literals will produce faster execution



In the case of two columns with few runs and many literals, our approaches are not able to jump many words

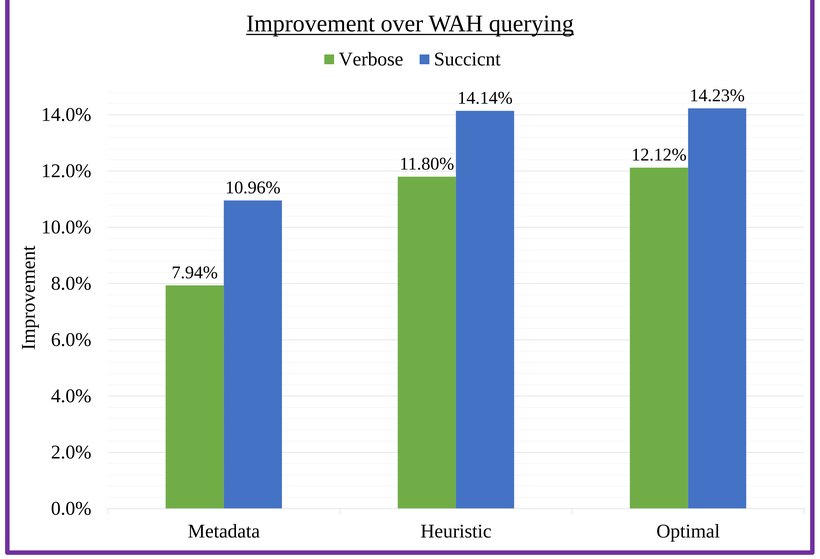
In the case of a column with many runs and a column with many literals, our approaches are exploited to their maximum



- Empirical study showed that columns with 30% or more runs juxtaposed with 26% or more literals had ~4% improvement

Results

- Both approaches improved querying speed
- Even faster improvement with heuristic



Bibliography

[1] K. Wu, E. J. Otoo, and A.Shoshani. Optimizing bitmap indices with efficient compression. *ACM Transactions on Database Systems (TODS)*, 31(1):1{38, 2006.

[2] K. Wu, E. J. Otoo, and A.Shoshani. Compressing bitmap indexes for faster search operations. In *SSDBM'02*, pages 99-108.

[3] K. Wu, E. J. Otoo, A. Shoshani, and H. Nordberg. Notes on design and implementation of compressed bit vectors. *Lawrence Berkeley National Laboratory, Tech. Rep*, 2001.