SBoost: Boosting Data Filtering in Columnar Store

THE UNIVERSITY OF CHICAGO

Hao Jiang, Aaron J. Elmore

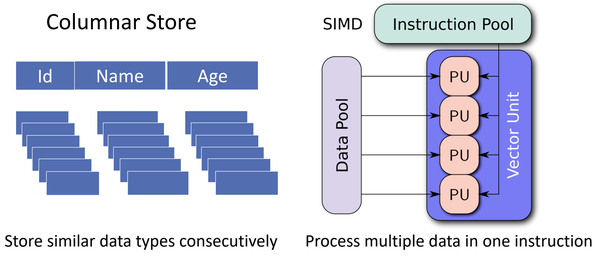
Abstract

Lightweight encodings are widely used in columnar stores. They are fast and allow in-situ queries [2]

- Bit-Packed Encoding [3]
- Delta Encoding [5]
- Run-Length Encoding
- Dictionary Encoding

We build SBoost, a columnar store based on Parquet [1] that uses innovative SIMD algorithms to speed up data filtering on encoded data by up to 100x.

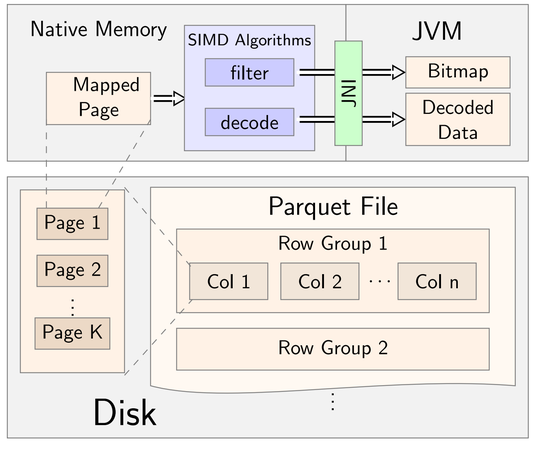
Background



Store similar data types consecutively

Process multiple data in one instruction

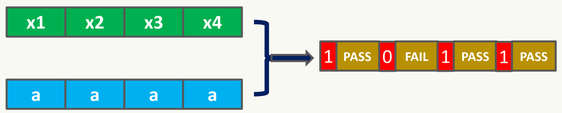
SBoost Architecture



Algorithms

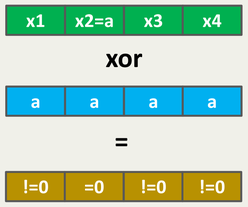
Data Filter on Bit-Packed Encoded Data

Idea: store the comparison result in the most significant bit

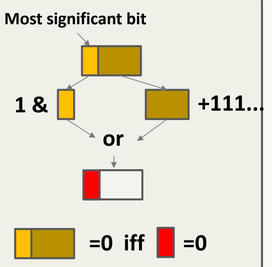


Equality

Use **xor** to check equality



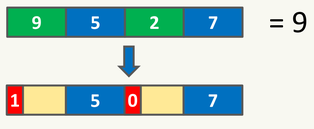
Most significant bit



Less

Data Filter on Run-Length Encoded Data

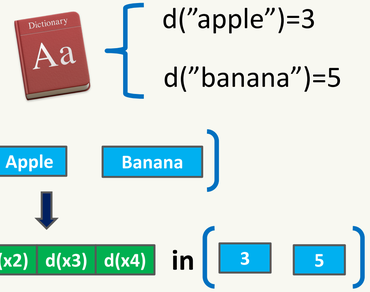
Use bit-packed filter algorithm to generate a run-length bitmap



Algorithms

Data Filter on Dictionary Encoded Data

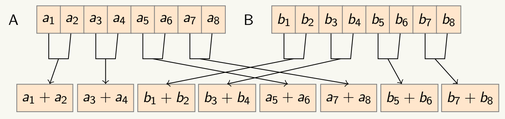
Use Order-Preserving Dictionary [4] to rewrite the query, then use bit-packed algorithm



Data Filter on Delta Encoded Data

Delta encoding stores delta between adjacent numbers. To decode, we need to compute the cumulative sum on a entry list.

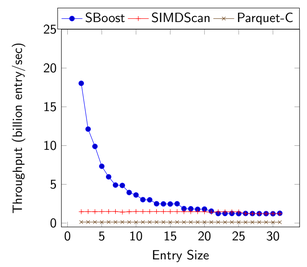
SIMD **hadd** performs multiple add in parallel

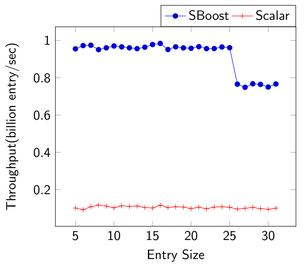


Use **hadd** to compute cumulative sum

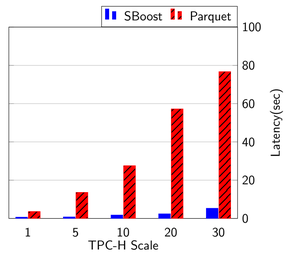
$b =$	b_7	b_6	b_5	b_4	b_3	b_2	b_1	b_0
$bp = shift256(b, 32)$	b_6	b_5	b_4	b_3	b_2	b_1	b_0	0
$s1 = hadd(b, bp)$	b_{56}	b_{34}	b_{67}	b_{45}	b_{12}	b_0	b_{23}	b_{01}
$s2 = permute(s1)$	b_{34}	b_{12}	b_{45}	b_{23}	0	b_0	0	b_{01}
$s3 = hadd(s1, s2)$	b_{14}	b_{25}	b_{36}	b_{47}	b_0	b_{01}	b_{02}	b_{03}
$s4 = permute(s3)$	b_0	b_{01}	b_{02}	b_{03}	0	0	0	0
$result = add\_32(s3, s4)$	b_{04}	b_{05}	b_{06}	b_{07}	b_0	b_{01}	b_{02}	b_{03}
$reorder(result)$	b_{07}	b_{06}	b_{05}	b_{04}	b_{03}	b_{02}	b_{01}	b_0

Experimental Result



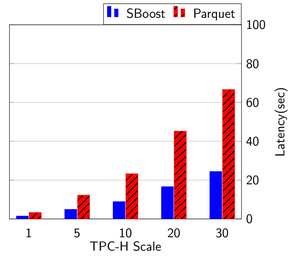


Throughput on Bit-Packed /RLE/ Dictionary



TPC-H Q1

Throughput on Delta



TPC-H Q6

Conclusion

- SBoost provides innovative SIMD algorithms on Lightweight Encodings
- Speed up Data Filtering on Columnar Stores
- Fully Compatible with existing Datasets
- Next Step: Explore more encoding schemes (on Double Data Type)

Reference

1. Apache Parquet. https://parquet.apache.org/
2. D. Abadi, SIGMOD’06, 671-682
3. T. Willhalm, VLDB’ 09, 385-394
4. C. Binnig, SIGMOD’09, 283-296
5. D. Lemire, Soft Prat Expr. Jan 15, 1-29