

Powerset Viewer: A Datamining Application

Jordan Lee

1

Update

2

Update

- Completed Tools and Features
 - And relevant GUI widgets

3

Update

- Completed Tools and Features
 - And relevant GUI widgets
- Implemented animation between zoom states and automatic zooming

4

Update

- Completed Tools and Features
 - And relevant GUI widgets
- Implemented animation between zoom states and automatic zooming
- Increased alphabet size from 14 to 30
 - Optimized calculations

5

Update

- Completed Tools and Features
 - And relevant GUI widgets
- Implemented animation between zoom states and automatic zooming
- Increased alphabet size from 14 to 30
 - Optimized calculations
- Increased alphabet size from 30 to 45
 - Realized set cardinality is, in practice, low
 - Using max set size of 10

6

Milestones Status Update

- #1 Completion of the basic visualization of a randomized database of small set size (~10)

7

Milestones Status Update

- #1 Completion of the basic visualization of a randomized database of small set size (~10)
- #2 Addition of a single level of "marking".
- #3 Addition of multiple levels of "marking" (6)
- #4 Addition of background marking to demarcate areas of sets containing different amounts of items.

8

Milestones Status Update

- #1 Completion of the basic visualization of a randomized database of small set size (~10)
- #2 Addition of a single level of "marking".
- #3 Addition of multiple levels of "marking" (6)
- #4 Addition of background marking to demarcate areas of sets containing different amounts of items.
- #5 Implement multiple constraints

9

Milestones Status Update

- #1 Completion of the basic visualization of a randomized database of small set size (~10)
- #2 Addition of a single level of "marking".
- #3 Addition of multiple levels of "marking" (6)
- #4 Addition of background marking to demarcate areas of sets containing different amounts of items.
- #5 Implement multiple constraints
- #6 Increase maximum possible dataset size to at least 100.

10

Difficulties

- BigInteger solution to increase maximum alphabet caused massive slow-down
 - Recall: required BigIntegers to support > 30 alphabet size
 - Solution: redesign keys to use integers and create a bridge to map integers to BigInteger positions

11

BEFORE BRIDGE

- Incoming Set (Position = 982) Success!
- Incoming Set (Position = $2^{32} + 1$) CRASH!
 - Integer too large

12

AFTER BRIDGE

- Incoming Set (Position = 982)
 - Encode to Key #1 Success!
- Incoming Set (Position = $2^{32} + 1$)
 - Encode to Key #2 Success!
- Incoming Set (Position = arbitrarily large)
 - Encode to Key #3 Success!

13

Difficulties

- BigInteger solution to increase maximum alphabet caused massive slow-down
 - Recall: required BigIntegers to support > 30 alphabet size
 - Solution: redesign keys to use integers and create a bridge to map integers to BigInteger positions
- Expensive initial costs
- Grid size limited by integer restrictions
 - Solution: create grid on the fly

14

Benchmarks

- Low Cardinality First

MEMORY (MB)	SET COUNT
76	10M
75	1M
74	100,000
73	10,000
58	1,000

15

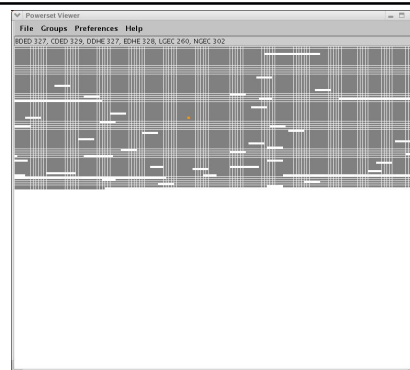


Figure: Low Cardinality (10000 sets) 73 MB

16

Benchmarks (cont'd)

- Random Generated

MEMORY (MB)	SET COUNT
72	263
71	168
70	127
72	30
71	10

17

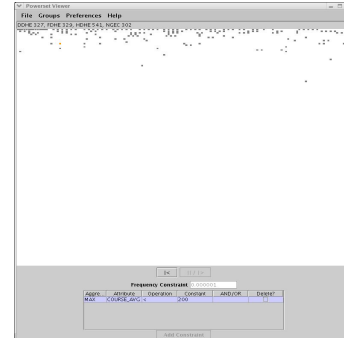


Figure: Random (176 sets) 71 MB

18

Questions and Comments

19