Built for Speed: Using Bloom Filters for File Identification

Doug White

NIST United States Department of Commerce
National Institute of Standards and Technology

Disclaimer

Trade names and company products are mentioned in the text or identified. In no case does such identification imply recommendation or endorsement by the National Institute of Standards and Technology, nor does it imply that the products are necessarily the best available for the purpose.

Statement of Disclosure

This research was funded by the National Institute of Standards and Technology Office of Law Enforcement Standards, the Department of Justice National Institute of Justice, the Federal Bureau of Investigation and the National Archives and Records Administration.

Issues Identified

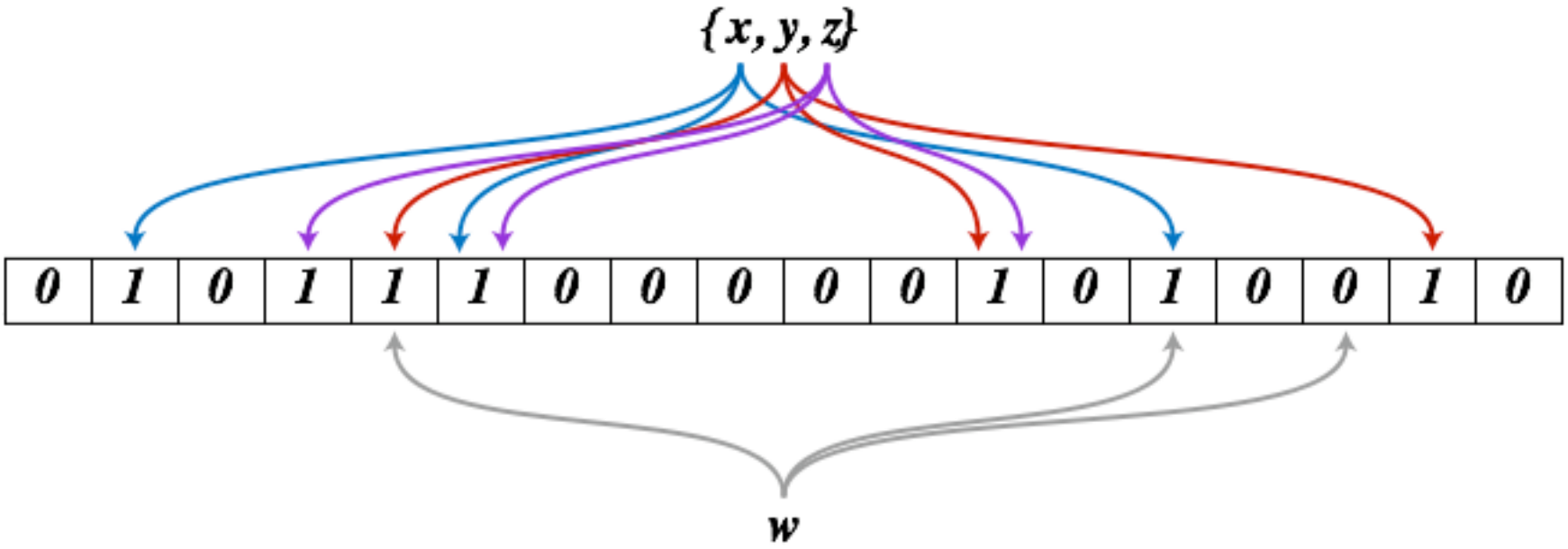
- Storage and distribution of tens of millions of hash values
- Storage and distribution of hundreds of millions of block hash values
- Speed of testing acquired hash values
- Interagency awareness without information release

Bloom Filter

A Bloom filter is a data structure that is used to test whether an element is a member of a set.

- False positives
- Elements can not removed

Most implementations are dynamic, growing as data is added to a back-end storage system.



Items x, y, and z have been added to the Bloom filter.

A search for item w yields a negative result.

Storage Space

Bloom filters have an advantage over other data structures which require storing at least the data items themselves.

A Bloom filter with 1% false positive rate requires only about 9.6 bits per element regardless of element size.

The false positive rate can be reduced by a factor of ten each time 4.8 bits per element are added.

Storage Space

NSRL investigated values of m = 2^32 and n = 10^8, which equates to a 512MB bit array containing 100,000,000 items.

A value of k = 16 allows a false positive rate of 0.000001%.

This compares favorably to 1.6GB needed for 100,000,000 MD5 hashes.

Storage Space

NSRL investigated values of m = 2^35 and n = 10^9, which equates to a 4GB bit array containing 1 billion items.

A value of k = 16 allows a false positive rate of 0.000014%.

This compares favorably to 16GB needed for 1 billion MD5 hashes.

Speed of Access

Bloom filters have the property that the time needed to add items or test set membership is a fixed constant, O(k), independent of the number of items in the set.

No other constant-space set data structure has this property.

The k lookups in a Bloom filter are independent and can be parallelized.

NSRL Implementation

Fixed size files allow use of stable vector algorithms.

Fixed size files with stable algorithms reduce two of three variables to constants when computing false positive rate.

Code and example filters are available at http://www.nsrl.nist.gov/RDS/rds_2.13/bloom

NSRL File Structure

- 512 Byte header
  - File signature
  - Agency information
  - Bloom parameters (bits, items, keys)
  - SHA1 and MD5 of data section
  - Text description of contents
- 512MB data (2^32 bits)
- Unbounded trailer

Measurements

Experiments focused on a 512MB filter, as math could be performed with 32 bit integers and 512MB was easily held in RAM.

Using a 2GHz intel Core 2 Duo, 10 million MD5 values can be added to a 512MB filter in less than 10 seconds.

Average query speed is on the order of 15,000 results per second.

Query speed increases as the ratio of unknown items increases.

Information Distribution

Bloom filter distribution can be as simple as a bitwise-or process for updates.

Filters can be built for specific query universes.

Data items are not distributed.

Next Steps

- Investigation of 4GiB, billion item sets
- Investigation of k value / false positive rate tradeoffs in larger scales
- Prototype disk block imager
- Publicly available prototype for feedback

Contacts

Douglas White
www.nsrl.nist.gov
nsrl@nist.gov

Barbara Guttman
Software Diagnostics & Conformance Testing Division
barbara.guttman@nist.gov

Sue Ballou, Office of Law Enforcement Standards
Rep. For State/Local Law Enforcement
susan.ballou@nist.gov