Incremental Clustering-Based Compression

Student: Luboš Krčál
Supervisor: Jan Holub

Faculty of Information Technology
Czech Technical University in Prague

Archiver

• It is possible to store more information about the deduplication, simhashing, clustering and grouping - the archivadata
• This allows for convenient CRUD operations over files in the archive

Compressing groups are compressed using standard compressors, such as Deflate, BZip2, etc.

Example of metadata implementation:

```
// magic number 0000: 0000 0123
// number of compression groups and chunks 0004: 0000 ICIC 002D 40AC
// original data size 0014: 0000 0061 224F
// offsets of the 1. group to compressed and orig data 001C: 0000 0882 0013 6002
// offset of the 1. chunk ICIC: 0003 2421
```

Compression

• Compression groups are compressed using standard compressors, such as Deflate, BZip2, etc.
• Metadata is then added, representing all the chunks and compression groups

Grouping & Reordering

• Clustering is gradually processed and compression groups are determined
• Compression groups are represented with a single cluster

Clustering

• Hierarchical clustering - binary tree, leaves represent chunks
• Bottom-up - new node inserted from the bottom
• Randomized KD-trees used to find the nearest neighbor
• Simhash calculated for parent nodes - as linear combination of children
• Preserves heap property on inter-cluster distances (with simhash distances)

Deduplication

• Uses standard hashes to find duplicate chunks
• Removing duplicate chunks is more effective than compressing them together
• Most effective for small chunks

Chunking

• Binary data is split into content-defined chunks
• Resistant to character inserts or shifts
• Uses partial pattern matches to mark delimiters

SIMILARITY HASHING

• Features: n-gram, compression features, etc. Every feature is hashed
• Standard simhash: merges feature hashes based on majority value per hash position (vector over $Z_2$, same length as feature hashes)
• Extended simhash: feature hashes are combined into a vector and normalized (vector over $Z_2$, of arbitrary length)
• Vector norms are used as distance measures

Example of binary simhash for the string "SWISS MISS".

```
hash("S"") = 111001001011001000101101101101101
hash("W"") = 100010101000111110110010111000110
hash("I"") = 10000101111011011101011011101000
hash("S"") = 00001011100111111001101110001111
hash("M"") = 011010001010011011111111010000111
simhash("SWISS MISS") = 000010001001111110111011010001111
```

At the end, the hash distance is calculated between the chunks and the new cluster is inserted between c1 and c2 as a sibling of c2

```
3
4
5
6
```

Disbalanced clustering:

• Optional extra balancing (in scenarios where heap property is not inherent)
• Optional deep distance (use multiple levels of descendants)
• Optional representatives (use only selected descendants)

• New approach to compression
• Extends current near-deduplication systems
• Overcomes locality-based redundancy removal failures of standard compression algorithms
• Has the potential of being an ultimate archiver

Compression ratio

Redundant datasets

• Compression ratio significantly improved for very redundant datasets
• For non-redundant dataset, the ratio was almost the same. This is a success, since the files were now compressed in very small batches
• Using weaker compression on appropriately preprocessed data results in both faster and better compression than using a strong compressor as it is