Practical Data Compression Presentation

**What is Data Compression?**

* Bits – Computer storage units (can take the value of 0 or 1);
* Lossless compression – full dataset is retained;
* Lossy compression – partial dataset is retained;

**Huffman Coding**

* Symbol concatenation 🡪 Huffman tree creation 🡪 Code generation;
* Right paths must be the same, left paths must be the same
* Produces variable-length, prefix codes.

**Lempel-Ziv-77 (LZ77)**

* Sliding Window – Search buffer**|**Look-Ahead window;
* Match look-ahead data with search buffer data;
* Store data as tokens

**Lempel-Ziv-4 (LZ4)**

* Variant of LZ77
* Store data as blocks - **DATA BLOCK:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Token | Symbol Length+ | Symbols | Offset | Match Length+ |

* Symbol Length+ and Match Length+ are overflow fields;

**Implementations**

* Produced Encoder and Decoder for LZ4 and Huffman Coding;
* Compared against three commercial/open-source compressors;
* Achieved 52.4% compression overall;

**The Next Step**

* Encoder runtimes could be improved;
* Currently, shortest average runtime is 14.4s**;**
* OS/Commercial compressors <1s average runtime.