Bidirectional Adaptive Compression

Aharon Fruchtman
Shmuel T. Klein
Dana Shapira
Data Compression

- **Static**
  - The model - the distribution of the encoded elements
    - Given in advance
    - Gathered in a first scan
- **Adaptive**
  - The model - learned incrementally.
Data Compression

- Statistical
  - Huffman
  - Arithmetic
- Dictionary Based
  - Lempel - Ziv.
Adaptive Algorithms

- Backward looking:
  - Base the current model on what has already been seen.
  - *The past is a good approximation of the future*

- Forward looking:
  - Exact statistics
  - Uses the model’s knowledge of what is still to come.
Differences

- **Backward:**
  - Increments the frequency
  - “Selfish” behavior

- **Forward:**
  - “Altruistic” approach
  - Decrements the frequency
Backward Looking Example

- Vitter’s dynamic Huffman variant
- NYT - Not Yet Transmitted
Backward

\[ T = \text{BANANAS} \]
Backward

\[ T = \text{BANANAS} \]

ASCII(\(B\))

\[ \varepsilon(T) = 01000010 \]
Backward

\[ T = \text{BANANAS} \]

\[ \text{ASCII(B)} \]

\[ \mathcal{E}(T) = 01000010 \]
Backward

\[ T = \text{BANANAS} \]

\[ \mathbf{T} = 01000010 \]

\[ \mathcal{E}(T) = 01000010 \ 0 \ 01000001 \]
Backward

\( T = \text{BANANAS} \)

NYT ASCII(A)

\( \mathcal{E}(T) = 01000010 0 01000001 \)
Backward

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 01000010 \ 0 \ 01000001 \ 10 \ 01001110 \]

NYT ASCII(N)
Backward

$T = \text{BANANAS}$

NYT ASCII(N)

$\varepsilon(T) = 01000010 0 01000001 10 01001110$

The Prague Stringology Conference (PSC-2019)
Backward

\[ T = \text{BANANAS} \]

\[ \mathcal{E}(T) = 01000010 \ 0 \ 01000001 \ 10 \ 01001110 10 \]

\[ \mathcal{E}(A) \]
Backward

\[ T = \textsc{BANANAS} \]

\[ \mathcal{E}(T) = 01000010 0 01000001 10 01001110 10 \]
Backward

$T = \text{BANANAS}$

$\mathcal{E}(T) = 01000010 0 01000001 10 01001110 10 111$

$\mathcal{E}(N)$
Backward

\[ T = \text{BANANAS} \]

\[ \mathcal{E}(T) = 01000010 0 01000001 10 01001110 10 111 \]

\[ \mathcal{E}(N) \]

The Prague Stringology Conference (PSC-2019)
Backward

\[ T = \text{BANANAS} \]

\[ \mathcal{E}(T) = 01000010\ 0\ 01000001\ 10\ 01001110\ 10\ 111\ 0 \]

\[ \mathcal{E}(A) \]
Backward

$T = \text{BANANAS}$

$\varepsilon(T) = 01000010\ 0\ 01000001\ 10\ 01001110\ 10\ 111\ 0\ 100\ 01010011$
Backward

\[ T = \text{BANANAS} \]

\[ \mathcal{E}(T) = 01000010 0 01000001 10 01001110 10 111 0 100 01010011 \]

NYT ASCII(S)
Forward – previous results

- **Best known bound** for dynamic is $\leq n$ bits + Static

- For a given distribution of frequencies, the average codeword length of FORWARD is at least as good as the average codeword length of STATIC

- **Classic** might produce a file twice the size of Forward
Backward

\[ T = \text{BANANAS} \]
Forward

$T = \text{BANANAS}$
Forward:

\[ T = \text{BANANAS} \]

\[ \mathcal{E}(T) = 101 \]

\[ \mathcal{E}(B) \]
Forward:

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 1010 \]

\[ \varepsilon(A) \]
Forward:

$T = \text{BANANAS}$

$\varepsilon(T) = 101011$

$\varepsilon(N)$
Forward:

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 1010110 \]

\[ \varepsilon(A) \]
Forward:

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 101011011 \]

\[ \varepsilon(N) \]
Forward:

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 1010110110 \]

\[ \varepsilon(A) \]
Forward:

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 1010110110 \]

No Need to transmit \( S \)
Drawbacks of current methods

- **Backward and Forward** - use information about the distribution which isn’t necessarily needed.
- **Static** - frequencies of the characters in the entire text.
- **String of characters** \{a, b, ..., z\} followed by numbers \{0,...,9\}.
A new Hybrid coding

- NYT - Not Yet Transmitted
- Encoding the model
  - Forward - exact frequencies at the beginning of the process
  - Backward - incrementally

Hybrid - NYT+ASCII+freq
Hybrid

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 01000010 1 \]

ASCII(B) \( c_\delta(1) \)
Hybrid

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 01000010 1 01000001 0101 \]

ASCII(A) \( c_\delta(3) \)
Hybrid

$T = \text{BANANAS}$

$\epsilon(T) = 01000010101000010101$

ASCII(A) $c_8(3)$
Hybrid

\[ T = \text{BANANAS} \]

\[ \varepsilon(T) = 01000010 \ 1 \ 01000001 \ 0101 \ 0 \ 01001110 \ 0100 \]

NYT ASCII(N) c_8(2)
Hybrid

$T = \text{BANANAS}$

$\mathcal{E}(T) = 01000010 \ 1 \ 01000001 \ 0101 \ 0 \ 01001110 \ 0100 \ 0$

$\mathcal{E}(A)$
Hybrid

\[ T = \text{BANANAS} \]

\[ \mathcal{E}(T) = 01000010 \ 1 \ 01000001 \ 0101 \ 0 \ 01001110 \ 0100 \ 0 \ 11 \]

\[ \mathcal{E}(N) \]
Hybrid

\( T = \text{BANANAS} \)

\[ \mathcal{E}(T) = 01000010 1 01000001 0101 0 01001110 0100 0 11 0 \]
Hybrid

\[ T = \text{BANANAS} \]

\[ \mathcal{E}(T) = 01000010 \ 1 \ 01000001 \ 0101 \ 0 \ 01001110 \ 0100 \ 0 \ 11 \ 0 \]

\[ \mathcal{E}(A) \]
Hybrid

\( T = \text{BANANAS} \)

\[ \varepsilon(T) = 01000010 \ 1 \ 01000001 \ 0101 \ 0 \ 01001110 \ 0100 \ 0 \ 11 \ 0 \ 01010011 \ 1 \]

ASCII(S) \( c_\delta(1) \)
Generic Hybrid-ENCODE\((T = x_1 \cdots x_n)\)

Preprocess \(T\) to get \(\text{freq}(\sigma_i), \ \forall \sigma_i \in \Sigma\)

Initialize the model with \(\text{NYT}\) with \(\text{freq}(\text{NYT}) \leftarrow |\Sigma|\)

Encode \(\text{freq}(\text{NYT})\)

for \(i \leftarrow 1\) to \(n\) do

if \(x_i\) has already appeared earlier then

encode \(x_i\) according to current model

\(\text{freq}(x_i) \leftarrow \text{freq}(x_i)-1\)

else

encode \(\text{NYT}\) according to current model

\(\text{freq}(\text{NYT}) \leftarrow \text{freq}(\text{NYT})-1\)

output \(\text{ASCII}(x_i)\)

encode \(\text{freq}(x_i)\)

Update the model with \(x_i, \text{freq}(x_i)\) and \(\text{freq}(\text{NYT})\)
Theorem

- The expected performance of HYBRID is at least as good as FORWARD

\[ \Sigma_r \subset \Sigma \]

\[ h_j = f_j \]
Theorem

- The expected performance of HYBRID is at least as good as FORWARD

\[ \Sigma_r \subset \Sigma \quad h_j = f_j \]

\[
\sum_{\sigma \in \Sigma_r} p_\sigma h_\sigma \leq \sum_{\sigma \in \Sigma_r} p_\sigma f_\sigma
\]

Huffman code built for \( P = \{ p_\sigma | \sigma \in \Sigma_r \} \)

The Prague Stringology Conference (PSC-2019)
Remarks

- Not necessarily true that $h_j \leq f_j$ for $j < t$
- Moderate expected savings by using HYBRID instead of FORWARD
  - $J$ and $Q$ appear with probability 0.002

Main contribution: improve a method which already seems better than one considered “optimal”
## Empirical Results

<table>
<thead>
<tr>
<th>File</th>
<th>Full Size</th>
<th>Size of Encoded File</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Static</td>
</tr>
<tr>
<td>ebib</td>
<td>3,711,020</td>
<td>1,940,573</td>
</tr>
<tr>
<td>exe</td>
<td>48,640</td>
<td>31,296</td>
</tr>
<tr>
<td>ftxt</td>
<td>7,648,930</td>
<td>4,443,525</td>
</tr>
<tr>
<td>eng</td>
<td>52,428,800</td>
<td>29,914,197</td>
</tr>
<tr>
<td>dig – ch</td>
<td>3,726,683</td>
<td>1,969,884</td>
</tr>
</tbody>
</table>

The Prague Stringology Conference (PSC-2019)
“Never touch the screen while you’re compressing a file!”

Copyright 1996 Randy Glasbergen. www.glasbergen.com