New Entropy Measures for Tries with Applications to the XBWT

11th Workshop on Data Structures in Bioinformatics

Lorenzo Carfagna[1] and **Carlo Tosoni**[2]

1. University of Pisa, Italy
2. Ca' Foscari University of Venice, Italy

Ca' Foscari University of Venice

18/02/2026

What's the Worst-Case Entropy?

Definition: Worst-Case Entropy

Let $\mathcal{U}$ be a set, the **worst-case entropy** $\mathcal{H}^{wc}(\mathcal{U})$ of $\mathcal{U}$ is defined as

$$\mathcal{H}^{wc}(\mathcal{U}) = \log_2 |\mathcal{U}|$$

Example, if $\mathcal{U} = \{\text{dog, cat, bird, mouse}\}$, then $\mathcal{H}^{wc}(\mathcal{U}) = \log_2 |\mathcal{U}| = \log_2 4 = \mathbf{2}$.

What's the Worst-Case Entropy?

If **we assign a unique codeword to every element** of $\mathcal{U}$, then there exists an element of $\mathcal{U}$ having **codeword length** at least $\mathcal{H}^{wc}(\mathcal{U})$.

- dog $\rightarrow$ 0
- cat $\rightarrow$ 1
- bird $\rightarrow$ 01
- mouse $\rightarrow$ 11

Both 'bird' and 'mouse' have a codeword of length $2 = \mathcal{H}^{wc}(\mathcal{U})$

Let us see how this applies to the case of strings!

What's the Worst-Case Entropy?

Consider the string **S = aaaaabaaaaaaaaabaaa**.

- If $\mathcal{U}$ is **the set of strings** of **length** $n = 20$ and **alphabet** $\sigma = 2$:

$$\mathcal{H}^{wc}(\mathcal{U}) = n \log \sigma = 20 \text{ bits}$$

- If $\mathcal{U}$ is the set of strings where **a** and **b appear 18** and **2 times**:

$$\mathcal{H}^{wc}(\mathcal{U}) = \log \binom{20}{2} \approx 7.57 \text{ bits}$$

Worst-case entropy with fixed frequencies can be much smaller!

What's the Worst-Case Entropy?

Consider the string **S** = **aaaaabaaaaaaaaabaaa**.

- If $\mathcal{U}$ is **the set of strings** of **length** $n = 20$ and **alphabet** $\sigma = 2$:

$$\mathcal{H}^{wc}(\mathcal{U}) = n \log \sigma = 20 \text{ bits}$$

- If $\mathcal{U}$ is the set of strings where **a** and **b appear 18** and **2 times**:

$$\mathcal{H}^{wc}(\mathcal{U}) = \log \binom{20}{2} \approx 7.57 \text{ bits}$$

Worst-case entropy with fixed frequencies can be much smaller!

Worst-Case Entropy vs Empirical Entropy

For a string S:

- $n_w = \#$ of characters having context $w \in \Sigma^*$.
- $n_c = \#$ of characters equal to $c \in \Sigma$.

(k-th) **Empirical entropy**: $\mathcal{H}_k(S) = \sum_{w \in \Sigma^k} \sum_{c \in \Sigma} n_{w,c} \log \frac{n_w}{n_{w,c}}$

Worst-Case Entropy vs Empirical Entropy

For a string S:

- $n_w = \#$ of characters having context $w \in \Sigma^*$.
- $n_c = \#$ of characters equal to $c \in \Sigma$.

(k-th) **Empirical entropy**: $\mathcal{H}_k(S) = \sum_{w \in \Sigma^k} \sum_{c \in \Sigma} n_{w,c} \log \frac{n_w}{n_{w,c}}$

Our contributions

1. Extend $\mathcal{H}_k$ and $\mathcal{H}^{wc}$ **from strings to tries!**

   Well-known **worst-case trie entropy** $\log \frac{1}{n}\binom{n\sigma}{n-1}$ [1] without fixed frequencies.

2. Reachability of our empirical entropy with **arithmetic coding**.

3. Comparison between these entropies and other trie measures.

4. BWT of a trie can be **compressed** and **indexed** in $\mathcal{H}_k(\mathcal{T}) + o(n)$.

1. *R. Graham, D. Knuth, and O. Patashnik: Concrete Mathematics. Addison-Wesley. (1994)*

Our contributions

❶ Extend $\mathcal{H}_k$ and $\mathcal{H}^{wc}$ **from strings to tries!**

Well-known **worst-case trie entropy** $\log \frac{1}{n}\binom{n\sigma}{n-1}$ [1] without fixed frequencies.

❷ Reachability of our empirical entropy with **arithmetic coding**.

❸ Comparison between these entropies and other trie measures.

❹ BWT of a trie can be **compressed** and **indexed** in $\mathcal{H}_k(\mathcal{T}) + o(n)$.

1. *R. Graham, D. Knuth, and O. Patashnik: Concrete Mathematics. Addison-Wesley. (1994)*

Our contributions

❶ Extend $\mathcal{H}_k$ and $\mathcal{H}^{wc}$ **from strings to tries!**

Well-known **worst-case trie entropy** $\log \frac{1}{n}\binom{n\sigma}{n-1}$ [1] without fixed frequencies.

❷ Reachability of our empirical entropy with **arithmetic coding**.

❸ Comparison between these entropies and other trie measures.

④ BWT of a trie can be **compressed** and **indexed** in $\mathcal{H}_k(\mathcal{T}) + o(n)$.

1. *R. Graham, D. Knuth, and O. Patashnik: Concrete Mathematics. Addison-Wesley. (1994)*

Our contributions

1. Extend $\mathcal{H}_k$ and $\mathcal{H}^{wc}$ **from strings to tries!**

   Well-known **worst-case trie entropy** $\log \frac{1}{n}\binom{n\sigma}{n-1}$ [1] without fixed frequencies.

2. Reachability of our empirical entropy with **arithmetic coding**.

3. Comparison between these entropies and other trie measures.

4. BWT of a trie can be **compressed** and **indexed** in $\mathcal{H}_k(\mathcal{T}) + o(n)$.

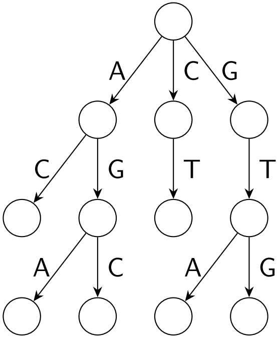
1. *R. Graham, D. Knuth, and O. Patashnik: Concrete Mathematics. Addison-Wesley. (1994)*

Applications

Tries are used in many applications, including: **databases**, **search engines**, **semantic web**, and **NLP**.

In **bioinformatics** they can be used to represent/index genomes or k-mers [2,3].

Compression achieved only if they share long prefixes.



$D = \{AC, AGA, AGC, CT, GTA, GTC\}$

2. *Chikhi et al., Data Structures to Represent a Set of k-long DNA Sequences, ACM Computing Surveys (2022)*

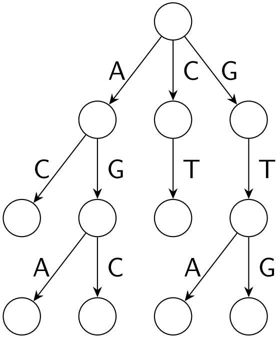
3. *Altschul et al., Basic Local Alignment Search Tool, Journal of Molecular Biology (1990)*

Applications

Tries are used in many applications, including: **databases**, **search engines**, **semantic web**, and **NLP**.

In **bioinformatics** they can be used to represent/index genomes or k-mers [2,3].

Compression achieved only if they share long prefixes.



$$D = \{AC, AGA, AGC, CT, GTA, GTC\}$$

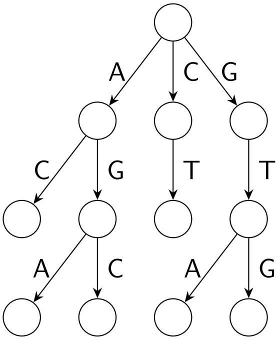
2. *Chikhi et al., Data Structures to Represent a Set of k-long DNA Sequences, ACM Computing Surveys (2022)*

3. *Altschul et al., Basic Local Alignment Search Tool, Journal of Molecular Biology (1990)*

Applications

BWT-based indexes found many applications in bioinformatics. Ex. are the **FM-index** [4] and the **r-index** [5].

Our index is a **generalisation of the FM-index to tries**, since it compresses to the empirical entropy.



$D = \{AC, AGA, AGC, CT, GTA, GTC\}$

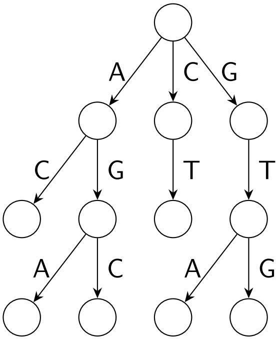
4. *P. Ferragina and G. Manzini, Opportunistic Data Structures with Applications, FOCS (2000)*

5. *Gagie et al., Fully Functional ST and Optimal Text Searching in BWT-Runs Bounded Space, J. ACM (2020)*

Applications

BWT-based indexes found many applications in bioinformatics. Ex. are the **FM-index** [4] and the **r-index** [5].

Our index is a **generalisation of the FM-index to tries**, since it compresses to the empirical entropy.

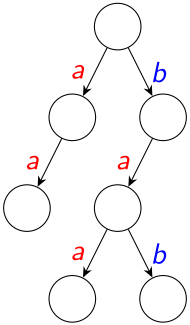


$D = \{AC, AGA, AGC, CT, GTA, GTC\}$

4. *P. Ferragina and G. Manzini, Opportunistic Data Structures with Applications, FOCS (2000)*

5. *Gagie et al., Fully Functional ST and Optimal Text Searching in BWT-Runs Bounded Space, J. ACM (2020)*

Worst-Case Entropy with Fixed Frequencies



We consider a **fixed distribution of edge-labels**.

Example: trie has 4 edges labeled by a and 2 edges labeled by b

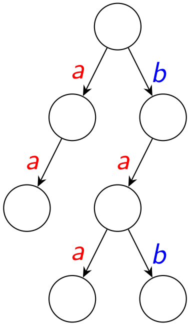
Number of tries $|\mathcal{U}|$ with a given symbol distribution is:

$$|\mathcal{U}| = \frac{1}{n} \prod_{c \in \Sigma} \binom{n}{n_c}$$

$n \leftarrow \#$ of nodes, $n_c \leftarrow \#$ of edges labeled by c

Example: the tries with 7 nodes, 4 edges labeled by a and 2 edges labeled by b are $\frac{1}{7}\binom{7}{4}\binom{7}{2} = 105$.

Worst-Case Entropy with Fixed Frequencies



We consider a **fixed distribution of edge-labels**.

Example: trie has 4 edges labeled by *a* and 2 edges labeled by *b*

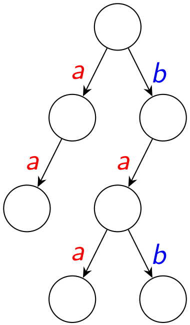
Number of tries $|\mathcal{U}|$ with a given symbol distribution is:

$$|\mathcal{U}| = \frac{1}{n} \prod_{c \in \Sigma} \binom{n}{n_c}$$

$n \leftarrow \#$ of nodes, $n_c \leftarrow \#$ of edges labeled by c

Example: the tries with 7 nodes, 4 edges labeled by *a* and 2 edges labeled by *b* are $\frac{1}{7}\binom{7}{4}\binom{7}{2} = 105$.

Worst-Case Entropy with Fixed Frequencies



Therefore, the **worst-case number of bits** to encode a trie with a fixed symbol distribution is:

$$\mathcal{H}^{wc}(\mathcal{U}) = \log|\mathcal{U}| = \sum_{c\in\Sigma} \log \binom{n}{n_c} - \log n$$

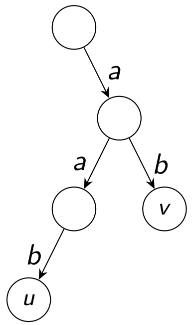
Example: To encode a tries with 7 nodes, 4 edges labeled by a and 2 edges labeled by b we need $\log \binom{7}{4} + \log \binom{7}{2} - \log 7 \approx 7$ bits in the worst-case!

Empirical Entropy for Tries

We also extended the **empirical entropy to tries**, our measure:

1. Encodes the **labels and the topology simultaneously**.
2. Use information on the **edge-labels distribution**.
3. Exploits the **k-length context** of the nodes to achieve compression.

Example: In figure, **nodes u** and **v** have **2-length context** equal to **ab**.

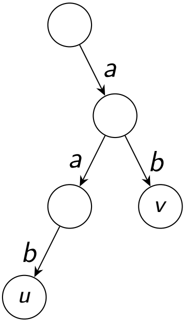


Empirical Entropy for Tries

We also extended the **empirical entropy to tries**, our measure:

1. Encodes the **labels and the topology simultaneously**.
2. Use information on the **edge-labels distribution**.
3. Exploits the **k-length context** of the nodes to achieve compression.

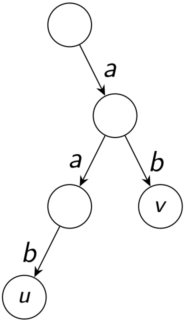
Example: In figure, **nodes u** and **v** have **2-length context** equal to **ab**.



Empirical Entropy for Tries

We also extended the **empirical entropy to tries**, our measure:

1. Encodes the **labels and the topology simultaneously**.
2. Use information on the **edge-labels distribution**.
3. Exploits the **k-length context** of the nodes to achieve compression.

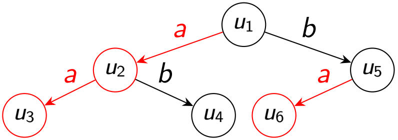


Example: In figure, **nodes u** and **v** have **2-length context** equal to **ab**.

Formula Empirical Entropy for Tries

For $w \in \Sigma^k$ and $c \in \Sigma$, consider the integers $\mathbf{n_w}$ and $\mathbf{n_{w,c}}$:

- $\mathbf{n_w} = |\{u \in V \mid u \text{ has context } w\}|$
- $\mathbf{n_{w,c}} = |\{u \in V \mid u \text{ has context } w \text{ and there exists } u \xrightarrow{c} v\}|$



Example: In figure, $\mathbf{n_a = 3}$.

Indeed, u_2, u_3, and u_6 are reached by the string a.

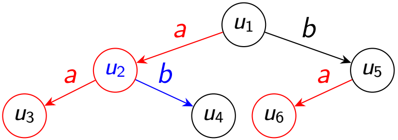
Definition: k-th order empirical entropy $\mathcal{H}_k(\mathcal{T})$

$$\mathcal{H}_k(\mathcal{T}) = \sum_{c \in \Sigma} \sum_{w \in \Sigma^k} n_{w,c} \log\left(\frac{n_w}{n_{w,c}}\right) + (n_w - n_{w,c}) \log\left(\frac{n_w}{n_w - n_{w,c}}\right)$$

Formula Empirical Entropy for Tries

For $w \in \Sigma^k$ and $c \in \Sigma$, consider the integers $\mathbf{n_w}$ and $\mathbf{n_{w,c}}$:

- $\mathbf{n_w} = |\{u \in V \mid u \text{ has context } w\}|$
- $\mathbf{n_{w,c}} = |\{u \in V \mid u \text{ has context } w \text{ and there exists } u \xrightarrow{c} v\}|$



Example: In figure, $\mathbf{n_{a,b} = 1}$.

Among the nodes reached by a, only u_2 has an outgoing edge labeled by b.

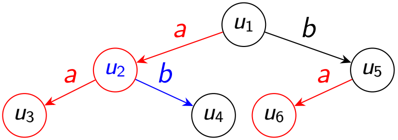
Definition: **k-th order empirical entropy $\mathcal{H}_k(\mathcal{T})$**

$$\mathcal{H}_k(\mathcal{T}) = \sum_{c \in \Sigma} \sum_{w \in \Sigma^k} n_{w,c} \log\left(\frac{n_w}{n_{w,c}}\right) + (n_w - n_{w,c}) \log\left(\frac{n_w}{n_w - n_{w,c}}\right)$$

Formula Empirical Entropy for Tries

For $w \in \Sigma^k$ and $c \in \Sigma$, consider the integers $\mathbf{n_w}$ and $\mathbf{n_{w,c}}$:

- $\mathbf{n_w} = |\{u \in V \mid u \text{ has context } w\}|$
- $\mathbf{n_{w,c}} = |\{u \in V \mid u \text{ has context } w \text{ and there exists } u \xrightarrow{c} v\}|$



Example: In figure, $\mathbf{n_{a,b} = 1}$.

Among the nodes reached by a, only u_2 has an outgoing edge labeled by b.

Definition: k-th order empirical entropy $\mathcal{H}_k(\mathcal{T})$

$$\mathcal{H}_k(\mathcal{T}) = \sum_{c\in\Sigma} \sum_{w\in\Sigma^k} n_{w,c} \log\left(\frac{n_w}{n_{w,c}}\right) + (n_w - n_{w,c}) \log\left(\frac{n_w}{n_w - n_{w,c}}\right)$$

Properties and Reachability

Properties analogous to the string entropies:

1. $\mathcal{H}_0(\mathcal{T}) = \mathcal{H}^{wc}(\mathcal{T}) + O(\sigma \log n)$
2. $\mathcal{H}_{k+1}(\mathcal{T}) \leq \mathcal{H}_k(\mathcal{T})$, for every $k \geq 0$

Theorem

Every trie $\mathcal{T}$ can be stored in $\mathcal{H}_k(\mathcal{T}) + (\sigma + 1)\sigma^k \log n$ bits

Idea: extend **arithmetic coding** to tries! Let's see an example.

Properties and Reachability

Properties analogous to the string entropies:

1. $\mathcal{H}_0(\mathcal{T}) = \mathcal{H}^{wc}(\mathcal{T}) + O(\sigma \log n)$
2. $\mathcal{H}_{k+1}(\mathcal{T}) \leq \mathcal{H}_k(\mathcal{T})$, for every $k \geq 0$

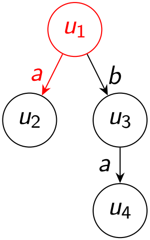
Theorem

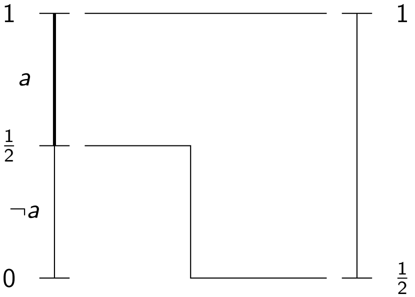
Every trie $\mathcal{T}$ can be stored in $\mathcal{H}_k(\mathcal{T}) + (\sigma + 1)\sigma^k \log n$ bits

Idea: extend **arithmetic coding** to tries! Let's see an example.

Reachability

We iterate over the nodes based on a **preorder visit**: u_1, u_2, u_3, u_4



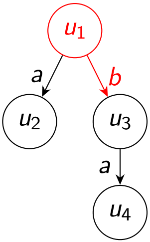


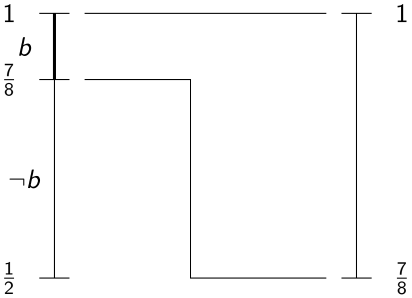
Initial interval $[0, 1)$, **probability of outgoing edge labeled by a**: $\dfrac{n_a}{n} = \dfrac{1}{2}$.

the new interval becomes $[\frac{1}{2}, 1)$.

Reachability

Before moving to u_2, we redo this operation for the **character b**



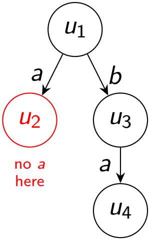


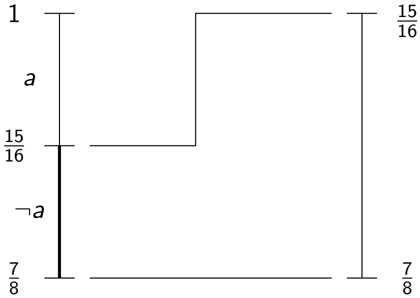
Probability of outgoing edge labeled by b: $\frac{n_b}{n} = \frac{1}{4}$.

Size of the interval shrinks to $\frac{1}{4}$, the new interval becomes $[\frac{7}{8}, 1)$.

Reachability

We move to the **next node in preorder**, i.e., u_2



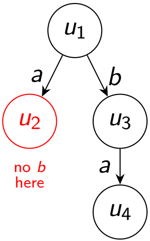


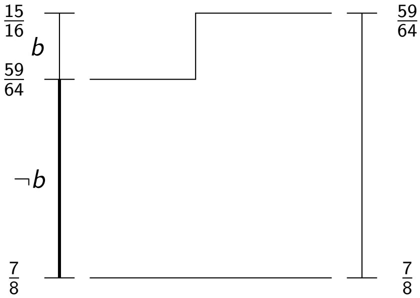
Probability of NO outgoing edge labeled by a: $1 - \frac{n_a}{n} = \frac{1}{2}$.

Halve the interval size, the new interval becomes $[\frac{7}{8}, \frac{15}{16})$.

Reachability

And again we redo the same operation for the **character b**



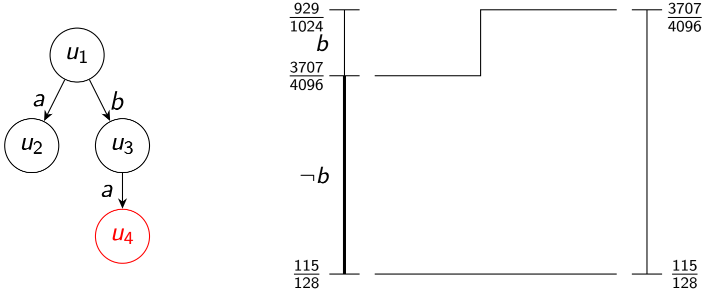


Probability of NO outgoing edge labeled by b: $1 - \frac{n_b}{n} = \frac{3}{4}$.

The interval size decreases by $\frac{3}{4}$, so it becomes $[\frac{7}{8}, \frac{15}{16})$.

Reachability

We continue in this way **until the last node**



The final interval is $[115/2^7,\ 3707/2^{12})$

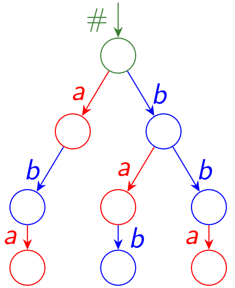
We can **store a point in this interval** in at most $\mathcal{H}_0(\mathcal{T}) + 2$ bits.

Comparison with the label entropy

Definition: Label entropy $\mathcal{H}_k^{label}$ [6]

$cover(w)$ = labels outgoing from nodes having context w

$\mathcal{H}_k^{label}(\mathcal{T}) = \sum_{w \in \Sigma^k} |cover(w)| \mathcal{H}_0(cover(w))$



Example:

$cover(\#) \leftarrow ab \quad \mathcal{H}_0(ab) = 1$

$cover(a) \leftarrow bb \quad \mathcal{H}_0(bb) = 0$

$cover(b) \leftarrow abaa \quad \mathcal{H}_0(abaa) \approx 0.811$

$\mathcal{H}_1^{label}(\mathcal{T}) = 1 * 2 + 0 * 2 + 4 * 0.811 \approx 5.245$

6. *Ferragina et al., Structuring labeled trees for optimal succinctness, and beyond, FOCS (2005)*

Comparison with the label entropy

Theorem

For every $k \geq 0$ and trie $\mathcal{T}$:

$$\mathcal{H}_k(\mathcal{T}) \leq \mathcal{H}_k^{label}(\mathcal{T}) + 1.443n$$

- $\mathcal{H}_k^{label}(\mathcal{T})$ accounts only for the labels, **not the tree topology!**
- To encode the tree topology we need $\mathbf{2n - \Theta(\log n)}$ **bits** [7]

$\mathcal{H}_k(\mathcal{T})$ **always smaller** than $\mathcal{H}_k^{label}(\mathcal{T}) + 2n - \Theta(\log n)$

For some family of tries $\mathcal{H}_k(\mathcal{T}) = 0$ and $\mathcal{H}_k^{label}(\mathcal{T}) = \Omega(n)$!

7. *G. Navarro, Compact Data Structures - A Practical Approach, Cambridge University Press (2016)*

Comparison with the label entropy

Theorem

For every $k \geq 0$ and trie $\mathcal{T}$:

$$\mathcal{H}_k(\mathcal{T}) \leq \mathcal{H}_k^{label}(\mathcal{T}) + 1.443n$$

- $\mathcal{H}_k^{label}(\mathcal{T})$ accounts only for the labels, **not the tree topology!**
- To encode the tree topology we need $\mathbf{2n - \Theta(\log n)}$ **bits** [7]

$\mathcal{H}_k(\mathcal{T})$ **always smaller** than $\mathcal{H}_k^{label}(\mathcal{T}) + 2n - \Theta(\log n)$

For some family of tries $\mathcal{H}_k(\mathcal{T}) = 0$ and $\mathcal{H}_k^{label}(\mathcal{T}) = \Omega(n)$!

7. *G. Navarro, Compact Data Structures - A Practical Approach, Cambridge University Press (2016)*

Comparison with the label entropy

Theorem

For every $k \geq 0$ and trie $\mathcal{T}$:

$$\mathcal{H}_k(\mathcal{T}) \leq \mathcal{H}_k^{label}(\mathcal{T}) + 1.443n$$

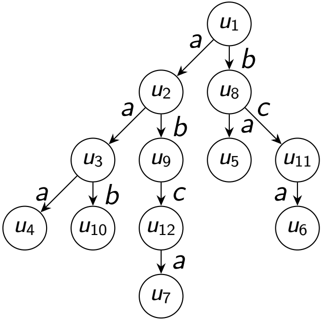
- $\mathcal{H}_k^{label}(\mathcal{T})$ accounts only for the labels, **not the tree topology!**
- To encode the tree topology we need $\mathbf{2n - \Theta(\log n)}$ **bits** [7]

$\mathcal{H}_k(\mathcal{T})$ **always smaller** than $\mathcal{H}_k^{label}(\mathcal{T}) + 2n - \Theta(\log n)$

For some family of tries $\mathcal{H}_k(\mathcal{T}) = 0$ and $\mathcal{H}_k^{label}(\mathcal{T}) = \Omega(n)$!

7. *G. Navarro, Compact Data Structures - A Practical Approach, Cambridge University Press (2016)*

XBWT of a trie



$out(u) \leftarrow$ set of outgoing labels of u

$u_1, u_2, \ldots, u_n \leftarrow$ nodes sorted **co-lexicographically**

Definition: XBWT [4]

$$\text{XBWT}(\mathcal{T}) = out(u_1), out(u_2), \ldots, out(u_n)$$

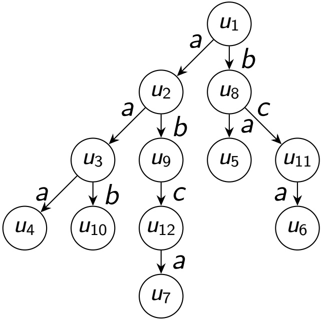
We can **compress** and **index** a trie in:

$\mathcal{H}_k(\mathcal{T}) + o(n)$, $\forall k = \max\{0, \alpha \log_\sigma n - 2\}$ s.t. $\alpha < 1$

co-lex	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}
XBWT	a b	a b	a b					a c	c		a	a

8. P. Ferragina et al. Compressing and Indexing Labeled Trees, with Applications. J. ACM (2009)

XBWT of a trie



$out(u) \leftarrow$ set of outgoing labels of u

$u_1, u_2, \ldots, u_n \leftarrow$ nodes sorted **co-lexicographically**

Definition: XBWT [4]

$$\text{XBWT}(\mathcal{T}) = out(u_1), out(u_2), \ldots, out(u_n)$$

We can **compress** and **index** a trie in:

$$\mathcal{H}_k(\mathcal{T}) + o(n),\ \forall k = \max\{0, \alpha \log_\sigma n - 2\} \text{ s.t. } \alpha < 1$$

co-lex	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9	u_{10}	u_{11}	u_{12}
XBWT	a b	a b	a b					a c	c		a	a

8. P. Ferragina et al. Compressing and Indexing Labeled Trees, with Applications. J. ACM (2009)

Space usage

Original article about the XBWT $\rightarrow \mathcal{H}_k^{label}(\mathcal{T}) + 2n + o(n)$ [8] bits, $\mathcal{H}_\mathbf{k}(\mathcal{T}) + \mathbf{o(n)}$ always smaller!

Theorem Succinctness

If no character appears in more than $n/2$ edges, then this index is **succinct**, i.e, its space usage is at most:

$$\mathcal{H}^{wc}(\mathcal{T}) + o(\mathcal{H}^{wc}(\mathcal{T}))$$

$\mathcal{H}^{wc}(\mathcal{T})$ is our new worst-case entropy.

8. P. Ferragina et al. Compressing and Indexing Labeled Trees, with Applications. J. ACM (2009)

Space usage

Original article about the XBWT $\rightarrow \mathcal{H}_k^{label}(\mathcal{T}) + 2n + o(n)$ [8] bits,
$\mathcal{H}_{\mathbf{k}}(\mathcal{T}) + \mathbf{o}(\mathbf{n})$ always smaller!

Theorem Succinctness

If no character appears in more than $n/2$ edges, then this index is **succinct**, i.e, its space usage is at most:

$$\mathcal{H}^{wc}(\mathcal{T}) + o(\mathcal{H}^{wc}(\mathcal{T}))$$

$\mathcal{H}^{wc}(\mathcal{T})$ is our new worst-case entropy.

8. P. Ferragina et al. Compressing and Indexing Labeled Trees, with Applications. J. ACM (2009)

Supported operations

These operations are supported **directly on the compressed format**.

	polylog alphabets	**arbitrary alphabets**
`subpath_query`(p) nodes reached by string p	$O(\lvert p \rvert)$	$O(\lvert p \rvert(\log \sigma + \log \log n))$
`parent`(u) parent node of u	$O(1)$	$O(1)$
`c-child`(u, c) child of u labeled by c	$O(1)$	$O(1)$
`j-child`(u, j) j-th child of u	$O(\sigma)$	$O(\sigma)$

Supported operations

These operations are supported **directly on the compressed format**.

	polylog alphabets	**arbitrary alphabets**
`subpath_query`(p) nodes reached by string p	$O(\|p\|)$	$O(\|p\|(\log\sigma + \log\log n))$
`parent`(u) parent node of u	$O(1)$	$O(1)$
`c-child`(u, c) child of u labeled by c	$O(1)$	$O(1)$
`j-child`(u, j) j-th child of u	$O(\sigma)$ **!!!**	$O(\sigma)$ **!!!**

This operation is still slow, must be sped up!!!

Thank you for your attention ☺

1. Extend $\mathcal{H}_k$ and $\mathcal{H}^{wc}$ **from strings to tries!**
2. Reachability of our empirical entropy with **arithmetic coding**.
3. Comparison between these entropies and other trie measures.
4. BWT of a trie can be **compressed** and **indexed** in $\mathcal{H}_k(\mathcal{T}) + o(n)$.

Preprint at: https://arxiv.org/abs/2512.11618