Suffix Trees: definition & size

Ben Langmead

Please sign guestbook ([www.langmead-lab.org/teaching-materials](http://www.langmead-lab.org/teaching-materials)) to tell me briefly how you are using the slides. For original Keynote files, email me ([ben.langmead@gmail.com](mailto:ben.langmead@gmail.com)).
Suffix trie

Suffix trie grows *quadratically* with string

Human genome is $3 \cdot 10^9$ bases long

If $m = 3 \cdot 10^9$, $m^2$ is far beyond what we can store in memory
Suffix trie: making it smaller

\[ T = \text{abaaba}\$ \]
Suffix trie: making it smaller

$T = \text{abaaba}$
Idea 1: Coalesce non-branching paths into a *single edge* with a *string* label
Suffix trie: making it smaller

$T = \text{abaaba}$

Idea 1: Coalesce non-branching paths into a *single edge* with a *string* label

Reduces # nodes, edges

Guarantees non-leaf nodes have >1 child
Suffix trie: making it smaller

\[ T = \text{abaaba}\$ \]
Suffix trie: making it smaller

$T = abaaba$
Suffix trie: making it smaller

$T = abaaba$
Suffix trie: making it smaller

$T = \text{abaaba}$

Diagram showing the construction or modification of a suffix trie.
Suffix tree

When no node has an "only child," we can bound the total # nodes in terms of the # leaves, $m$.
Suffix tree

In a *binary* tree, where each non-leaf has *exactly* 2 children, the # of non-leaves = $m - 1$
In a binary tree, where each non-leaf has exactly 2 children, the number of non-leaves = \( m - 1 \)
In a **binary** tree, where each non-leaf has **exactly** 2 children, the # of non-leaves = $m - 1$
If we allow non-leaves to have >2 children, then the number of leaves only increases relative to non-leaves.
Suffix tree

If we allow non-leaves to have >2 children, then the number of leaves only increases relative to non-leaves.
Suffix tree

If we allow non-leaves to have >2 children, then the number of leaves only increases relative to non-leaves.

# leaves $m$
# non-leaf nodes
If we allow non-leaves to have >2 children, then the number of leaves only increases relative to non-leaves.

Call this the "no-only-child" principle.
Suffix tree

\[ T = \text{abaaba$} \quad |T| = m \]

\[ \text{No-only-child principle} \]

- \# leaves? \( m \)
- \# non-leaf nodes \( \leq m - 1 \)
- \( \leq 2m - 1 \) nodes total — \( O(m) \)

Is total space \( O(m) \) now? \( \text{No: total length of edge labels grows with } m^2 \)
Idea 2: Store $T$ itself in addition to the tree. Convert edge labels to (offset, length) pairs with respect to $T$. 

$T = \text{abaaba}\$
**Suffix tree**

**Idea 2:** Store $T$ itself in addition to the tree. Convert edge labels to (offset, length) pairs with respect to $T$.

$T = \text{abaaba}\$
**Idea 2:** Store $T$ itself in addition to the tree. Convert edge labels to (offset, length) pairs with respect to $T$.

$T = \text{abaaba}$
Idea 2: Store $T$ itself in addition to the tree. Convert edge labels to (offset, length) pairs with respect to $T$. 

$T = \text{abaaba}$
**Suffix tree**

**Idea 2:** Store $T$ itself in addition to the tree. Convert edge labels to (offset, length) pairs with respect to $T$.

Total space is now $O(m)$

Suffix trie was $O(m^2)$!
Suffix tree: leaves hold offsets
Suffix tree: leaves hold offsets
Suffix tree: leaves hold offsets
Suffix tree

$T = abaaba$
Suffix tree

\[ T = \text{abaaba$} \]

Label = “aaba$”, the suffix at offset 2
Two notions of depth:

- **Node** depth: # edges from root to node
- **Label** depth: total length of edge labels from root to node
Suffix tree: depth & labels

Two notions of depth:

• **Node** depth: \# edges from root to node

• **Label** depth: total length of edge labels from root to node

Node depth = 2
Label depth = 2 + 4 = 6
Suffix tree: actual growth

Suffix trees built for first 500 prefixes of the lambda phage virus genome

Remember suffix trie plot:

123 K nodes
Suffix tree: actual growth

Suffix trees built for first 500 prefixes of the lambda phage virus genome

Remember suffix trie plot:

123 K nodes
Suffix trie

>100K nodes

Length prefix over which suffix trie was built

# suffix trie nodes

m(m+1)/2
actual
2m+2

Suffix tree

<1K nodes

Length prefix over which suffix tree was built

# suffix tree nodes

2m
actual
m