# Counting occurrences for a finite set of words: an inclusion-exclusion approach

*Pierre Nicodème*

CNRS - LIX, École polytechnique

joint work with *Frédérique Bassino, Julien Clément* and *Julien Fayolle*

# Problem setting

Compute separately the number of occurrences of a non-reduced set of words $\mathcal{U}$ in a random text under Bernoulli (non-uniform) model

**Reduced set:** no word is factor of another word

| Reduced | Non-Reduced |
|---|---|
| $\mathcal{U} = \{aab, ba, bb\}$ | $\mathcal{U} = \{aa, aab, bbaabb\}$ |

**Methods**

- Formal languages manipulations (Régnier-Szpankowski) (**it fails in the non-reduced case**)
- Aho-Corasick (automaton) + Chomsky-Schützenberger
- Inclusion-Exclusion (Goulden-Jackson, Noonan-Zeilberger)

# (Auto)-Correlation Set

**auto-correlation**

$$h = ababa \quad \rightsquigarrow \quad \begin{array}{l} ababa \\ ababa| \\ \quad ababa \\ \quad\quad ababa \end{array} \quad \rightsquigarrow \quad \mathcal{C}_{ababa,ababa} = \{\epsilon, ba, baba\}$$

$$\mathcal{C}_{h,h} = \{\ w, \quad h.w = r.h \quad \text{and} \quad |w| < |h|\ \}$$

**correlation**

$$\mathcal{C}_{h_1,h_2} = \{\ w, \quad h_1.w = r.h_2 \quad \text{and} \quad |w| < |h_2|\ \}$$

$$h_1 = baba,\ \ h_2 = abaaba \quad \longrightarrow \quad \mathcal{C}_{baba,abaaba} = \{aba, baaba\}$$

# Formal Languages Analysis

# (Régnier-Szpankowski)

Right $\mathcal{R} = \{\ t = u.h \quad \text{and} \ \nexists r, s \neq \epsilon, \quad t = r.h.s\ \}$

Minimal $\mathcal{M} \ = \{\ t \neq \epsilon, \quad h.t = u.h \ \ \text{and} \ \nexists r, s, \quad h.t = r.h.s\ \}$

Ultimate $\mathcal{U} = \{t, \quad \nexists r, s, \quad h.t = r.h.s\}$

Not $\mathcal{N} = \overline{\mathcal{A}^\star.h.\mathcal{A}^\star} = \{t, \ \nexists r, s, \quad t = r.h.s\}$

$$\mathcal{A}^\star = \mathcal{N} + \mathcal{R}.\,(\mathcal{M})^\star.\mathcal{U} \quad \Rightarrow \quad \mathcal{L}_x = \mathcal{N} + \mathcal{R}x.\,(\mathcal{M}x)^\star.\mathcal{U}$$

# Equations over the langages

$\mathcal{C} = \mathcal{C}_{h,h} \qquad \pi_h = \Pr(h)$ (Bernoulli model)

(I) $\mathcal{A}^\star = \mathcal{U} + \mathcal{M}\mathcal{A}^\star$ (II) $\mathcal{A}^\star h = \mathcal{R}.\mathcal{C} + \mathcal{R}.\mathcal{A}^\star.h$

(III) $\mathcal{M}^+ = \mathcal{A}^\star.h + \mathcal{C} - \epsilon$ (IV) $\mathcal{N}.\mathcal{A} = \mathcal{R} + \mathcal{N} - \epsilon$

**solving**

$$R(z) = \frac{\pi_h z^{|h|}}{\pi_h z^{|h|} + (1-z)C(z)} \qquad U(z) = \frac{1}{\pi_h z^{|h|} + (1-z)C(z)}$$

$$N(z) = \frac{C(z)}{\pi_h z^{|h|} + (1-z)C(z)} \qquad M(z) = 1 + \frac{z-1}{\pi_h z^{|h|} + (1-z)C(z)}$$

$$L(z,x) = \frac{1}{1 - z + \pi_h z^{|h|} \dfrac{1-x}{x + (1-x)C(z)}}$$

# Reduced sets (Régnier)

$\mathcal{R}_i, \mathcal{M}_{i,j}, \mathcal{U}_i \;\rightsquigarrow\; R_i(z), M_{i,j}(z), U_i(z)$

functions of $C_{h_1,h_1}(z), C_{h_2,h_2}(z), C_{h_1,h_2}(z), C_{h_2,h_1}(z)$

$$F(z, x_1, x_2) = N(z) + (x_1 R_1(z), x_2 R_2(z)) \begin{pmatrix} x_1 M_{1,1}(z) & x_2 M_{1,2}(z) \\ x_1 M_{2,1}(z) & x_2 M_{2,2}(z) \end{pmatrix}^{\star} \begin{pmatrix} U_1(z) \\ U_2(z) \end{pmatrix}$$

**This collapses in case of non-reduced sets**

# Aho-Corasick

- **Input:** non-reduced set of words $\mathcal{U}$.
- **Output:** automaton $\mathcal{A}_{\mathcal{U}}$ recognizing $\mathcal{A}^*\mathcal{U}$.

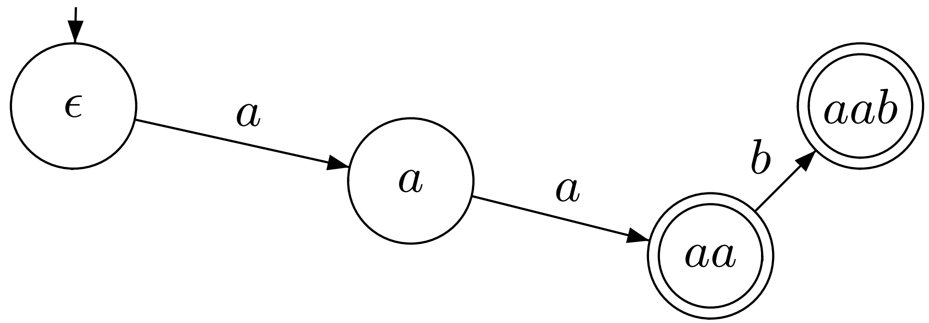
**Algorithm:**

1. build $\mathcal{T}_{\mathcal{U}}$, the ordinary trie representing the set $\mathcal{U}$
2. build $\mathcal{A}_{\mathcal{U}} = (\mathcal{A}, Q, \delta, \epsilon, T)$:
   - $Q = \text{Pref}(\mathcal{U})$
   - $T = \mathcal{A}^*\mathcal{U} \cap \text{Pref}(\mathcal{U})$
   - $\delta(q, x) = \begin{cases} qx & \text{if } qx \in \text{Pref}(\mathcal{U}), \\ \text{Border}(qx) & \text{otherwise,} \end{cases}$

   $\text{Border}(v)$ = the longest proper suffix of $v$ which belongs to $\text{Pref}(\mathcal{U})$ if defined, or $\epsilon$ otherwise.

# Example

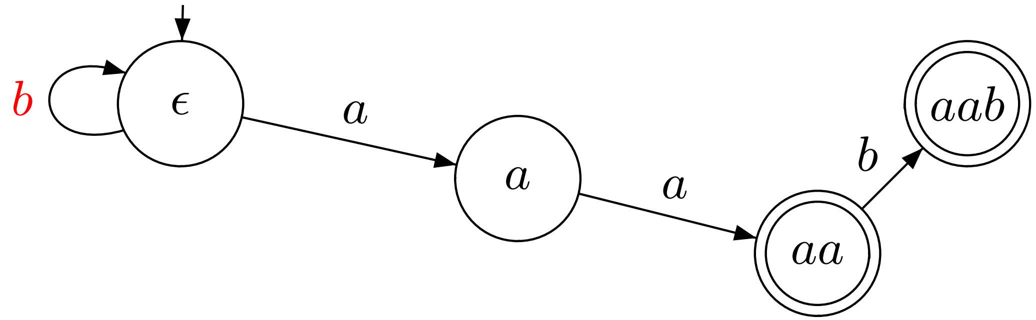
$\mathcal{U} = \{aab, aa\}$




**Trie $\mathcal{T}_{\mathcal{U}}$ of $\mathcal{U}$**

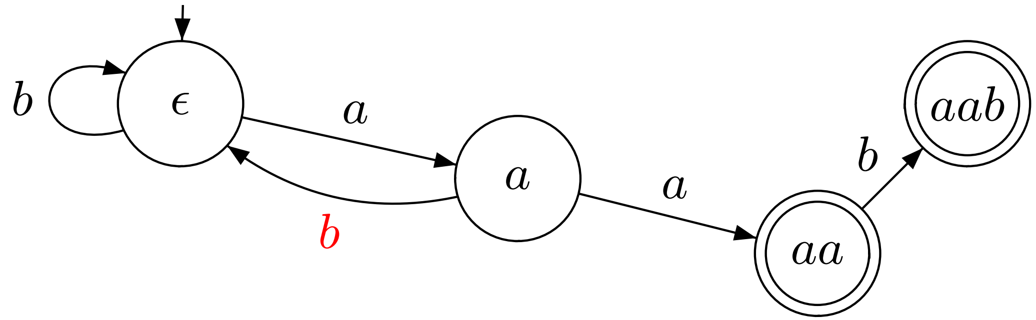
# Example

$\mathcal{U} = \{aab, aa\}$ $\delta(\epsilon, b) = \text{Border}(b) = \epsilon$



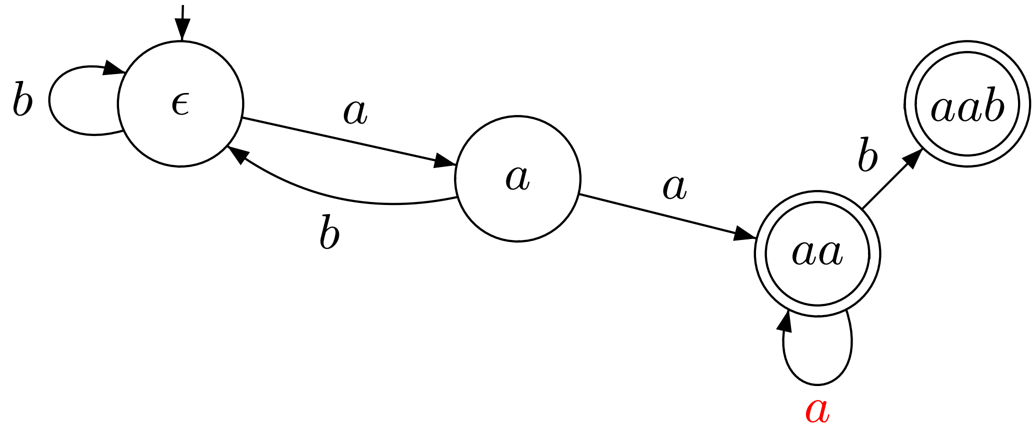
# Example

$\mathcal{U} = \{aab, aa\}$ $\delta(a, b) = \text{Border}(a.b) = \epsilon$



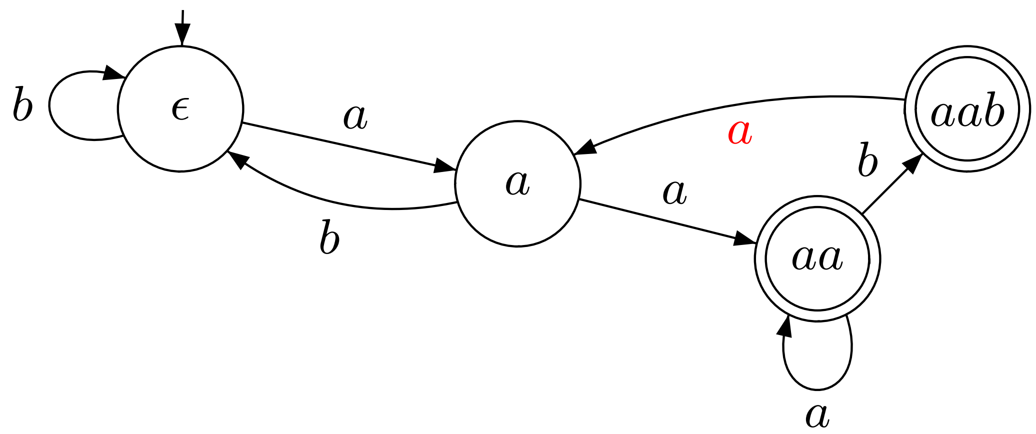
# Example

$\mathcal{U} = \{aab, aa\}$ $\qquad$ $\delta(aa, a) = \text{Border}(aa.a) = aa$



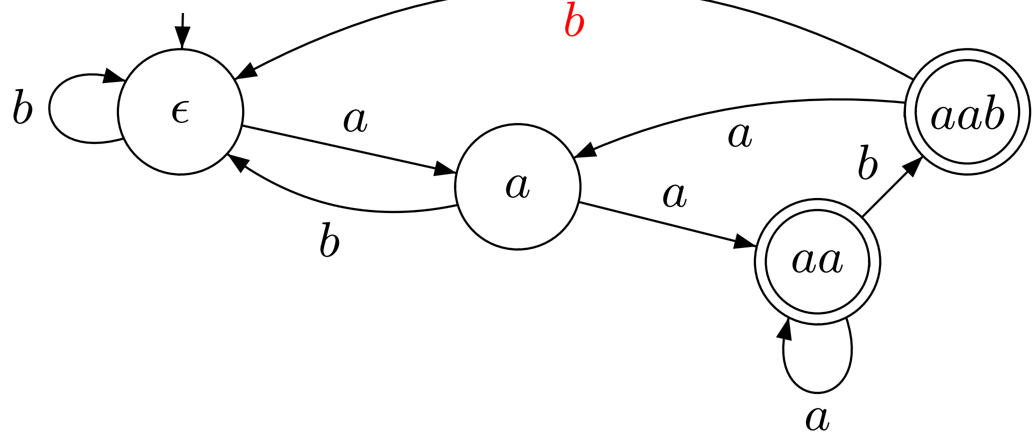
# Example

$\mathcal{U} = \{aab, aa\}$ $\quad$ $\delta(aab, a) = \text{Border}(aab.a) = a$



# Example

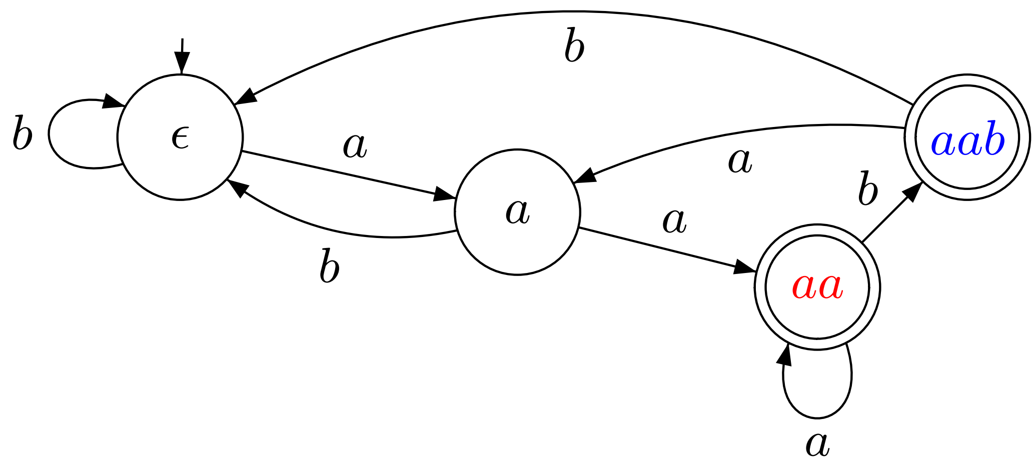
$\mathcal{U} = \{aab, aa\}$ $\delta(aab, b) = \text{Border}(aab.b) = \epsilon$



# Example

$\mathcal{U} = \{aab, aa\}$

$$\mathbb{T}(x_1, x_2) = \begin{pmatrix} b & a & 0 & 0 \\ b & 0 & ax_2 & 0 \\ 0 & 0 & ax_2 & bx_1 \\ b & a & 0 & 0 \end{pmatrix},$$

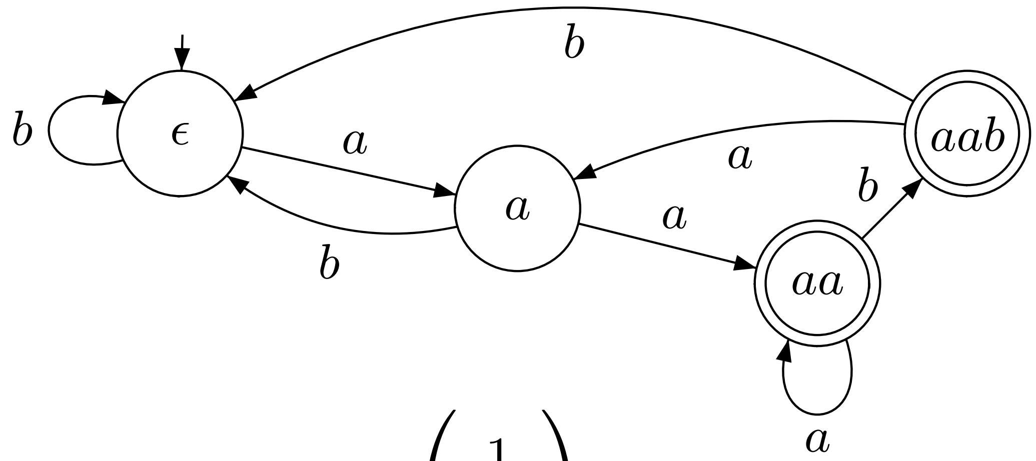



$x_1, x_2$ marks for $aab, aa$

# Example

$$\mathcal{U} = \{aab, aa\}$$

$$\mathbb{T}(x_1, x_2) = \begin{pmatrix} b & a & 0 & 0 \\ b & 0 & ax_2 & 0 \\ 0 & 0 & ax_2 & bx_1 \\ b & a & 0 & 0 \end{pmatrix},$$



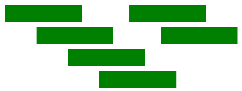

$$F(a, b, x_1, x_2) = (1, 0, 0, 0)(\mathbb{I} - \mathbb{T}(a, b, x_1, x_2))^{-1} \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

$$= \frac{1 - a(x_2 - 1)}{1 - ax_2 - b + ab(x_2 - 1) - a^2 b x_2 (x_1 - 1)^2}.$$

# Inclusion-Exclusion: one word

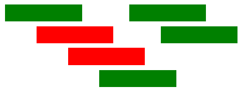
word $\color{red}{aaa}$ $\quad$ $p(x)$: unknown p.g.f of counts of $aaa$

$bbbbbaaaaaaaabbbbb$

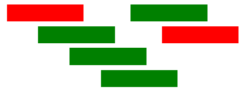


each occurrence is marked or not (flip-flop)

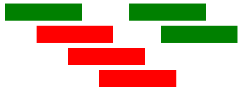
$bbbbbaaaaaaaabbbbb$



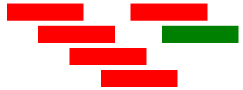
$bbbbbaaaaaaaabbbbb$

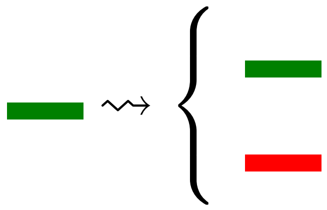


$bbbbbaaaaaaaabbbbb$



$bbbbbaaaaaaaabbbbb$





$$\color{green}{x} \leadsto \begin{cases} \color{green}{1} \\ \color{red}{+x} \end{cases} \qquad \begin{aligned} p(x) &\leadsto p(\color{blue}{1+x}) = \phi(\color{blue}{x}) \\ &\leadsto p(\color{red}{x}) = \phi(\color{red}{x-1}) \end{aligned}$$

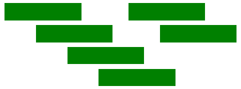
**computing easier $\phi(\color{red}{t})$ and substituting $\color{red}{t} \leadsto \color{red}{x-1}$**

**gives harder $\color{red}{p(x)}$**

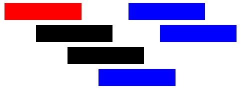
# One word - Clusters

word $aaa$

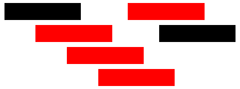
$bbbbbaaaaaaaabbbbb$



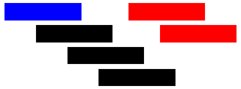
$bbbbbaaaaaaaabbbbb$



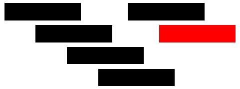
$bbbbbaaaaaaaabbbbb$



$bbbbbaaaaaaaabbbbb$



$bbbbbaaaaaaaabbbbb$



## clusters $\mathfrak{C}$

$$\mathfrak{C} = w + \mathfrak{C}.(\mathcal{C}_{w,w} - \epsilon) \Longrightarrow \qquad \mathfrak{C}(z,x) = \frac{x\pi_w z^{|w|}}{1 - x(C(z) - 1)}$$

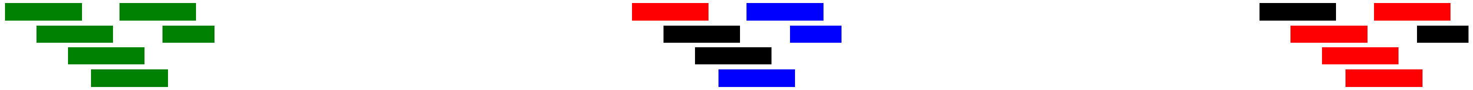
$$\mathcal{T} = \mathrm{Seq}(\mathcal{A} + \mathfrak{C}) \Longrightarrow \qquad \Phi(z,x) = \frac{1}{1 - z - \mathfrak{C}(z,x)}$$

$$F(z,x) = \Phi(z, x-1)$$

# Three words - Clusters (Goulden-Jackson)

$\mathcal{U} = \{aba, bab, aa\}$

$bbbbbababaabbbbb$ $\quad$ $bbbbbababaabbbbb$ $\quad$ $bbbbbababaabbbbb$



**clusters** $\mathfrak{C}_{i,j}$ begin with $w_i$ and finish with $w_j$

$$\mathfrak{C}_{i,j} = w_i \mathcal{C}_{w_i,w_j} + \sum_{1 \le k \le 3} \mathfrak{C}_{i,k}.(\mathcal{C}_{w_k,w_j} - \delta_{kj}\epsilon)$$

$$\mathfrak{C} = (w_1 {\color{blue}\bullet}, w_2 {\color{red}\bullet}, w_3 {\color{teal}\bullet}) \left( \mathbf{I} - \begin{pmatrix} \mathcal{C}_{w_1,w_1}{\color{blue}\bullet} - \epsilon & \mathcal{C}_{w_1,w_2}{\color{red}\bullet} & \mathcal{C}_{w_1,w_3}{\color{teal}\bullet} \\ \mathcal{C}_{w_2,w_1}{\color{blue}\bullet} & \mathcal{C}_{w_2,w_2}{\color{red}\bullet} - \epsilon & \mathcal{C}_{w_2,w_3}{\color{teal}\bullet} \\ \mathcal{C}_{w_3,w_1}{\color{blue}\bullet} & \mathcal{C}_{w_3,w_2}{\color{red}\bullet} & \mathcal{C}_{w_3,w_3}{\color{teal}\bullet} - \epsilon \end{pmatrix} \right)^{-1} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$$

$$\mathcal{T} = \mathrm{Seq}(\mathcal{A} + \mathfrak{C}) \Longrightarrow \Phi(z, x_1, x_2, x_3) = \frac{1}{1 - z - \mathfrak{C}(z, x_1, x_2, x_3)}$$

$$F(z, x_1, x_2, x_3) = \Phi(z, x_1 - 1, x_2 - 1, x_3 - 1) = \frac{1}{1 - z - \mathfrak{C}(z, x_1 - 1, x_2 - 1, x_3 - 1)}$$

# General Case: Non Reduced Set of Words

$\mathcal{U} = \{aa, ab, baaaab\}$

Consider clusters I and II

```
I           II
aaaabbbbbbabaaaabbb
aa          aa
 aa          aa
  aa          aa
   ab          ab
```

– Cluster I: as in the one word or reduced case: "flip-flop" counting

– Cluster II: there are supplementary matches: "sticky" counting, attached to a right extension of one word to another word (here $ab$ to $baaaab$).

Main idea: consider the **"reduced" backbone** by counting **flip-flop** occurrences, add the **sticky** occurrences during the **right extensions** from one word to another (generalization of correlation)

# "Flip-Flop" versus "Sticky"

$\mathcal{U} = \{u_1 = aa, u_2 = ab, u_3 = baaaab\}$

I II

$aaaab\,bbbbb\,abaaaab\,bbb$

$aa$ $aa$

$aa$ $aa$

$aa$ $aa$

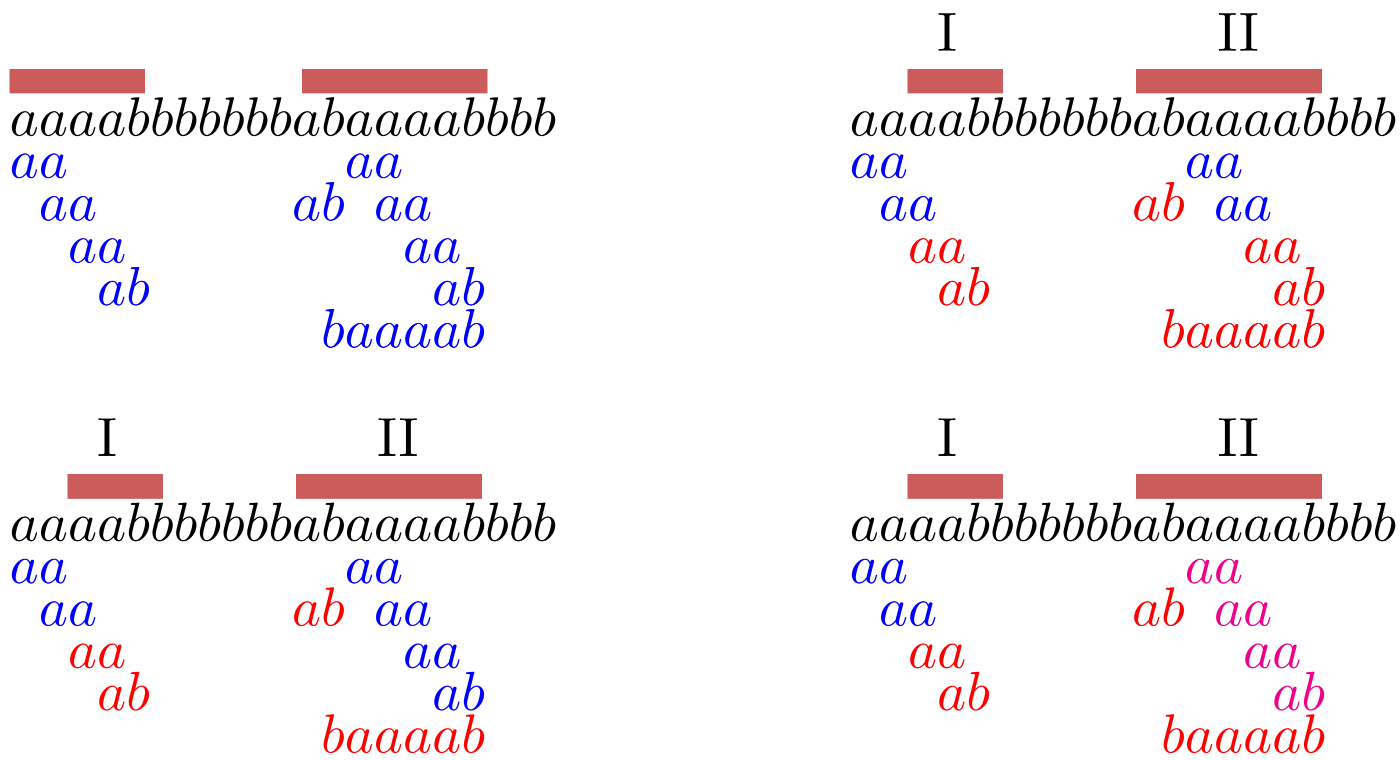
$ab$ $ab$

– **Cluster I "flip-flop"**

– **Cluster II contains "sticky" words**

# Inclusion-Exclusion: Non-Reduced Case

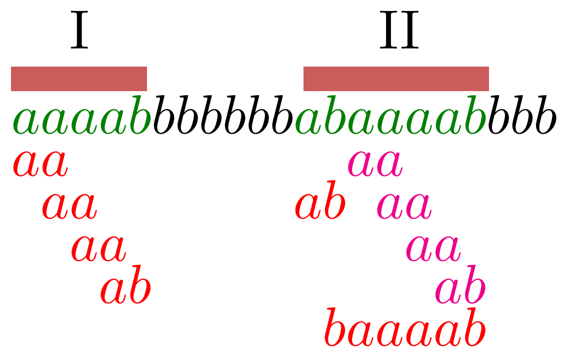
$\mathcal{U} = \{u_1 = aa, u_2 = ab, u_3 = baaaab\}$




1. **flip-flop** all occurrences giving **clusters**
2. **forget factor** occurrences
3. **add sticky occurrences**

# Counting "Flip-Flop" versus "Sticky"

$\mathcal{U} = \{u_1 = aa, u_2 = ab, u_3 = baaaab\}$




- **Cluster I "flip-flop":** $t_i \rightsquigarrow x_i - 1$
- **Cluster II**
  1. **"flip-flop"**: $t_i \rightsquigarrow x_i - 1$
  2. **"sticky"**: $v_i \rightsquigarrow x_i$

Remark: counting the sticky ocurrences by $v_i = 1 + t_i$ and doing $t_i \rightsquigarrow x_i - 1$ is correct

# Right Extension Sets and Matrices

**Right Extension Set** of a pair of words $(h_1, h_2)$

$\mathcal{E}_{h_1,h_2} = \{\, e \quad | \quad$ there exists $e' \in \mathcal{A}^+$ such that $h_1 e = e' h_2$ with $0 < |e| < |h_2|\}$.

if $h_1 \neq h_2$ have no factor relation, $\mathcal{E}_{h_1,h_2} = \mathcal{C}_{h_1,h_2}$ but $\mathcal{E}_{h,h} = \mathcal{C}_h - \epsilon$

**Right Extension Matrix** of a vector of words $\mathbf{u} = (u_1, \ldots, u_r)$

$$\mathcal{E}_{\mathbf{u}} = \left(\mathcal{E}_{u_i,u_j}\right)_{1 \le i,j \le r}.$$

**Examples**

$$\mathbf{u}_1 = (aba, ab) \;\Rightarrow\; \mathcal{E}_{\mathbf{u}_1} = \begin{pmatrix} ba & b \\ \emptyset & \emptyset \end{pmatrix} \qquad \mathcal{E}_{ab,aba} = \emptyset \quad \begin{cases} aba = |aba \\ e' = \epsilon \notin \mathcal{A}^+ \end{cases}$$

$$\mathbf{u}_2 = (aaaa, aaa) \;\Rightarrow\; \mathcal{E}_{\mathbf{u}_2} = \begin{pmatrix} a+a^2+a^3 & a+a^2 \\ a^2+a^3 & a+a^2 \end{pmatrix} \quad \begin{cases} a \notin \mathcal{E}_{aaa,aaaa} & aaa.a = |aaaa \\ aa \in \mathcal{E}_{aaa,aaaa} & aaa.aa = a.aaaa \end{cases}$$

# Counting Sticky Words

$\mathcal{U} = \{u_1 = aa, u_2 = baaaabaaaab\}$ $\quad$ $\mathcal{E}_{u_2,u_2} = \{aaaab, aaaabaaaab\}$

$$\begin{array}{l} baaaabaaaabaaaab \\ \qquad\quad baaaabaaaabaaaab \end{array} \qquad N_{2,1}(6) = 9 - 6 = 3$$

$$\begin{array}{l} baaaabaaaabaaaab \\ \qquad\qquad\quad baaaabaaaabaaaab \end{array} \qquad N_{2,1}(11) = 9 - 3 = 6$$

$$N_{i,j}(k) = \big|u_i\big|_j - \big|u_i[1\ldots|u_i| - k]\big|_j.$$

$$\langle\mathcal{E}_{u_2,u_2}\rangle_2 = \pi_a^4\pi_b z^5(t_1+1)^3 t_2 + \pi_a^8\pi_b^2 z^{10}(t_1+1)^6 t_2$$

# Formal Setting

$N_{i,j}(k)$ counts the number of occurrences of $u_j$ in $u_i$ ending in the last $k$ positions

$$N_{i,j}(k) = \left|u_i\right|_j - \left|u_i[1 \ldots |u_i| - k]\right|_j.$$

$\langle s \rangle_i$ **formal weight** of a **suffix** of word $u_i$

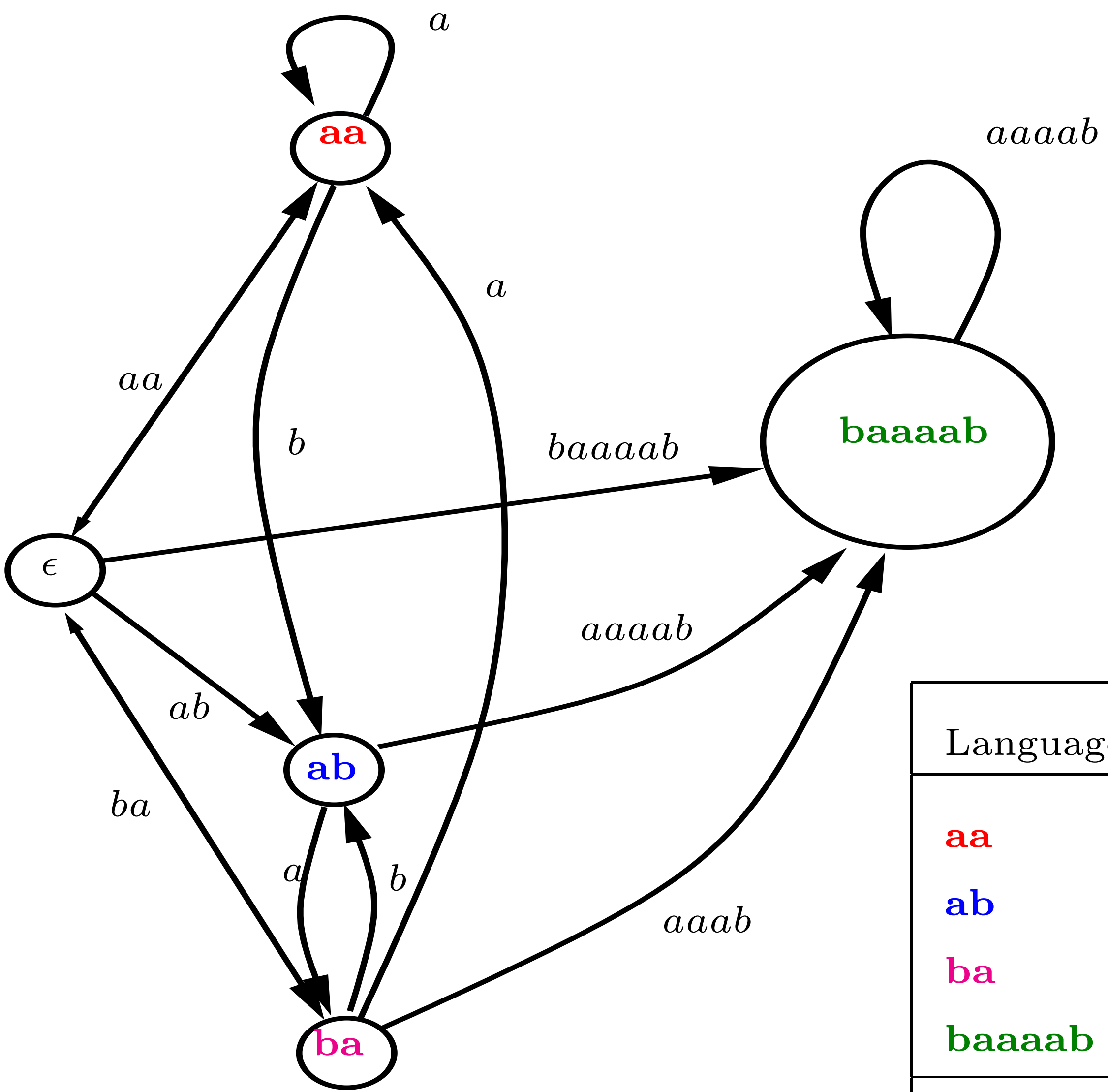
$$\langle s \rangle_i = \pi(s) z^{|s|} t_i \prod_{m \neq i} (t_m + 1)^{N_{i,m}(|s|)}.$$

extension to a set of words $S$ which are suffixes of $u_i$

$$\langle S \rangle_i = \sum_{s \in S} \langle s \rangle_i.$$

$$\mathcal{E}_{i,j} \quad \rightsquigarrow \quad \langle \mathcal{E}_{i,j} \rangle_j$$

# Right Extension Graph




$\mathcal{U} = \{$**aa**, **ab**, **ba**, **baaaab**$\}$

| Language | G. F. |
|---|---|
| **aa** | $t_1 z^2$ |
| **ab** | $t_2 z^2$ |
| **ba** | $t_3 z^2$ |
| **baaaab** | $t_4 z^6$ |
| $\mathcal{E}_{ab,ba} = \{a\}$ | $t_3 z$ |
| $\mathcal{E}_{ba,baaaab} = \{aaab\}$ | $(1+t_1)^2(1+t_2)t_4 z^4$ |
| $\mathcal{E}_{baaaab,baaaab} = \{aaaab\}$ | $(1+t_1)^3(1+t_2)t_4 z^5$ |

# Putting Things Together

$$\text{Let} \quad \langle \mathbf{u} \rangle = (\langle u_1 \rangle_1, \ldots, \langle u_r \rangle_r) \quad \text{and} \quad \langle \mathcal{E}_{\mathbf{u}} \rangle = \begin{pmatrix} \ldots & \ldots & \ldots \\ \ldots & \langle \mathcal{E}_{i,j} \rangle_j & \ldots \\ \ldots & \ldots & \ldots \end{pmatrix}$$

**Proposition I.** *The generating function $\mathfrak{C}(z, \mathbf{t})$ of clusters built from the set $\mathcal{U} = \{u_1, \ldots, u_r\}$ is given by*

$$\mathfrak{C}(z, \mathbf{t}) = \langle \mathbf{u} \rangle \cdot \left( \mathbb{I} - \langle \mathcal{E}_{\mathbf{u}} \rangle \right)^{-1} \cdot \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix},$$

*where $\mathbf{u} = (u_1, \ldots, u_r)$, $\mathbf{t} = (t_1, \ldots, t_r)$*

**Proposition II.** *The generating function $F(x, \mathbf{x})$ counting matches of a non-reduced set of words is*

$$F(z, \mathbf{x}) = \frac{1}{1 - z - \mathfrak{C}(z, \mathbf{x} - \mathbf{1})}$$

# Examples

$\mathcal{U} = \{u\}$

$$\mathfrak{C}(z,t) = \frac{t\langle u\rangle}{1 - t\langle \mathcal{E}_u\rangle} = \frac{t\pi(u)z^{|u|}}{1 - t(C(z) - 1)}$$

$\mathcal{U} = \{u_1, u_2\}$

$$\mathfrak{C}(z, t_1, t_2)$$

$$= \frac{t_1\langle u_1\rangle_1 + t_2\langle u_2\rangle_2 - t_1 t_2\big(\langle u_1\rangle_1\big[\langle \mathcal{E}_{2,2}\rangle_2 - \langle \mathcal{E}_{1,2}\rangle_2\big] + \langle u_2\rangle_2\big[\langle \mathcal{E}_{1,1}\rangle_1 - \langle \mathcal{E}_{2,1}\rangle_1\big]\big)}{1 - t_2\langle \mathcal{E}_{2,2}\rangle_2 - t_1\langle \mathcal{E}_{1,1}\rangle_1 + t_1 t_2\big(\langle \mathcal{E}_{1,1}\rangle_1\langle \mathcal{E}_{2,2}\rangle_2 - \langle \mathcal{E}_{2,1}\rangle_1\langle \mathcal{E}_{1,2}\rangle_2\big)}$$

# Algorithmic computation

```
Init(A_U)
for i ← 1 to r do
        f_i(u_i) ← 1
for w ∈ Pref(U) by a postorder traversal of the tree do
        for i ← 1 to r do
                for α ∈ A such that w · α ∈ Pref(u_i) do
                        f_i(w) ← π(α) z f_i(w · α) ∏_{j≠i} (1 + t_j)^{[[u_j suffix of w · α]]}
return (f_i)_{1≤i≤r}
```

```
Build-Extension-Matrix(A_U)
▷ Initialize the matrix (E_{i,j})_{1≤i,j≤r}
for i ← 1 to r do
        for j ← 1 to r do
                E_{i,j} ← 0
▷ Compute the maps (f_i(w)) for i = 1..r and w ∈ Pref(U)
(f_i)_{1≤i≤r} ← Init(A_U)
▷ Main loop
for i ← 1 to r do
        v ← u_i
        do      for j ← 1 to r do
                        E_{i,j} ← E_{i,j} + f_j(v)
                        v ← Border(v)
        while v ≠ ε
return E
```

Time complexity of the main loop $O(s \times r^2)$, where $r$ is the number of words and $s$ is the length of the longest suffix chain

(sequence $(u_1 = u, u_2 = \mathrm{Border}(u_1), u_3 = \mathrm{Border}(u_2), \ldots, u_s = \mathrm{Border}(u_{s-1}) = \epsilon)$)