Online Infix Probability Computation for Probabilistic Finite Automata

Marco Cognetta Yo-Sub Han Soon Chan Kwon
Yonsei University, Seoul, Republic of Korea
http://toc.yonsei.ac.kr

Problem

We compute infix probabilities for probabilistic finite automata *faster* and *online*.

$$\underbrace{\mathcal{P}(\Sigma^* w \Sigma^*)}_{\text{Infix probability of } w} = \sum_{x \in \Sigma^* w \Sigma^*} \overbrace{\mathcal{P}(x)}^{\text{Probability of } x}$$

Use the infix probability of w to compute the infix probability of wa.

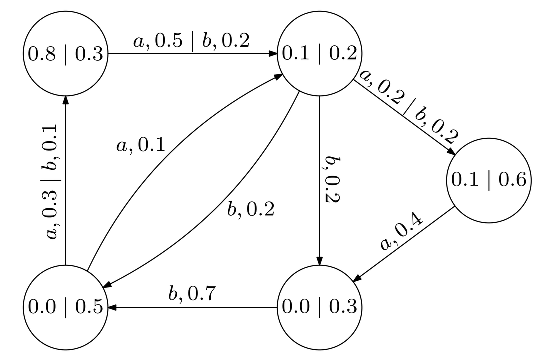
$$\mathcal{P}(\Sigma^* w \Sigma^*) \to \mathcal{P}(\Sigma^* wa \Sigma^*)$$

We consider the case where w is given as a stream.

Probabilistic Finite Automata

PFA $\mathcal{P} = (Q_\mathcal{P}, \Sigma, \{\mathbb{M}_\mathcal{P}(c)\}_{c \in \Sigma}, \mathbb{I}_\mathcal{P}, \mathbb{F}_\mathcal{P})$

- $\mathbb{M}_\mathcal{P}(c)$ — $|Q_\mathcal{P}| \times |Q_\mathcal{P}|$ transition matrix
- $\mathbb{I}_\mathcal{P}$ — $1 \times |Q_\mathcal{P}|$ initial weight vector
- $\mathbb{F}_\mathcal{P}$ — $|Q_\mathcal{P}| \times 1$ final weight vector



$$\mathcal{P}(w) = \mathbb{I}_\mathcal{P} \prod_{i=1}^{|w|} \mathbb{M}_\mathcal{P}(w_i) \mathbb{F}_\mathcal{P}$$

Previous Approach (Our EMNLP'18 Paper)

- Construct DFA $\mathcal{D}$ for $\Sigma^* w \Sigma^*$
- Perform state elimination:

$$\alpha_{i,j}^k = \alpha_{i,j}^{k-1} + \alpha_{i,k}^{k-1}(\alpha_{k,k}^{k-1})^* \alpha_{k,j}^{k-1}$$

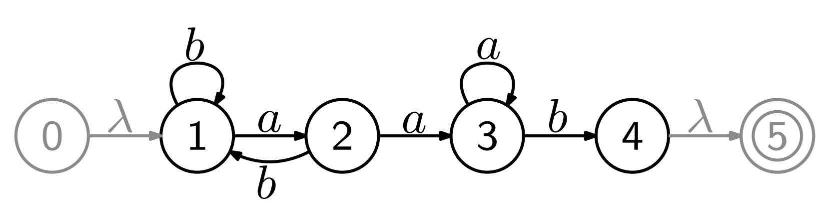
- Extract $\alpha_{0,k+1}^k$ to get $\mathcal{P}(\Sigma^* w_1 w_2 \dots w_k \Sigma^*)$

Regex	Matrix	Regex	Matrix
$\emptyset$	0	$R \cup S$	$\mathbb{M}_\mathcal{P}(R) + \mathbb{M}_\mathcal{P}(S)$
λ	1	RS	$\mathbb{M}_\mathcal{P}(R)\mathbb{M}_\mathcal{P}(S)$
c	$\mathbb{M}_\mathcal{P}(c)$	R^*	$(1 - \mathbb{M}_\mathcal{P}(R))^{-1}$

$$\mathbb{I}_\mathcal{P} \mathbb{M}_\mathcal{P}(R) \mathbb{F}_\mathcal{P} = \sum_{w \in R} \mathcal{P}(w)$$

Basic Incremental Algorithm

$\alpha_{i,j}^k$ = paths from q_i to q_j that don't visit states $> q_k$.



α^0

	0	1	2	3	4	5
0	$\emptyset$	λ	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$
1	$\emptyset$	b	a	$\emptyset$	$\emptyset$	$\emptyset$
2	$\emptyset$	b	$\emptyset$	a	$\emptyset$	$\emptyset$
3	$\emptyset$	$\emptyset$	$\emptyset$	a	b	$\emptyset$
4	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	λ
5	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

α^1

	0	1	2	3	4	5
0	$\emptyset$	$\lambda + b^*b$	b^*a	$\emptyset$	$\emptyset$	$\emptyset$
1	$\emptyset$	$b + bb^*b$	$a + bb^*a$	$\emptyset$	$\emptyset$	$\emptyset$
2	$\emptyset$	$b + bb^*b$	bb^*a	a	$\emptyset$	$\emptyset$
3	$\emptyset$	$\emptyset$	$\emptyset$	a	b	$\emptyset$
4	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	λ
5	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

$\alpha_{0,2}^1 = b^*a$ $\alpha_{0,3}^2 = b^*a(bb^*a)^*a$ $\alpha_{0,4}^3 = b^*a(bb^*a)^*aa^*b$

Algorithm 1: Incremental Infix Probability

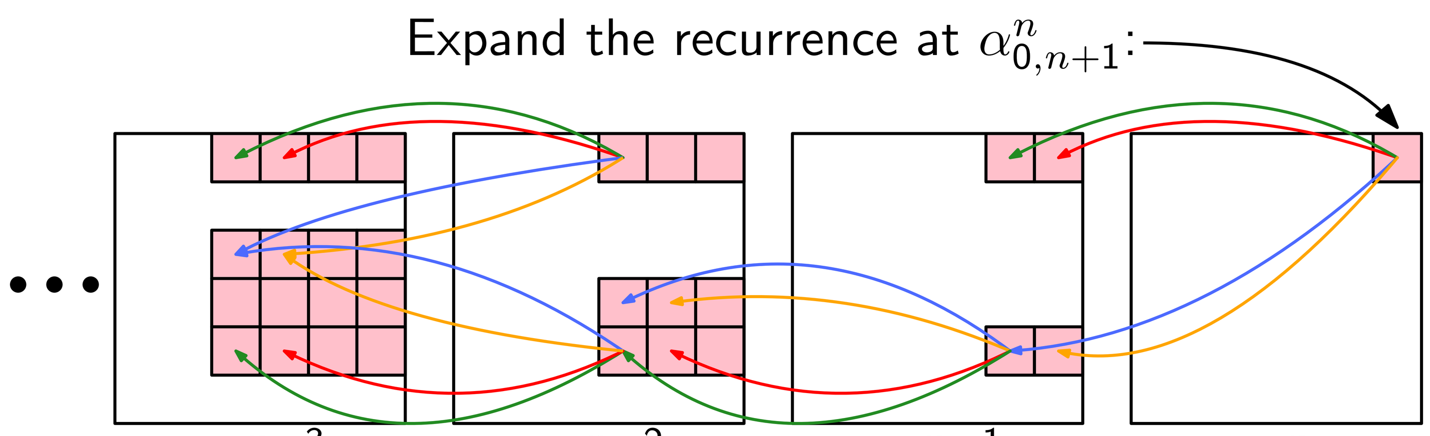
```
Data: PFA 𝒫, String w
𝒟 ← KMP DFA for w
n ← |Q_𝒟|
T, T' ← (n+2) × (n+2) table
for i, j ∈ [0, n+1] do
    if (i = 0 ∧ j = 1) ∨ q_i ∈ F ∧ j = n+1 then
        T_{i,j} = 1
    else
        for c such that δ(q_i, c) = q_j do
            T_{i,j} = T_{i,j} + M_𝒫(c)
V ← I_𝒫
for k ∈ [1, n] do
    V ← V(T_{k,k})* T_{k,k+1}
    yield V M_𝒫(Σ*) F_𝒫
    for i, j ∈ [0, n+1] do
        T'_{i,j} = T_{i,j} + T_{i,k}(T_{k,k})* T_{k,j}
    T ← T'
```

Time Complexity: $O(|w|^3 |Q_\mathcal{P}|^m)$

Recurrence Reanalysis

Not all cells need to be computed at each iteration.

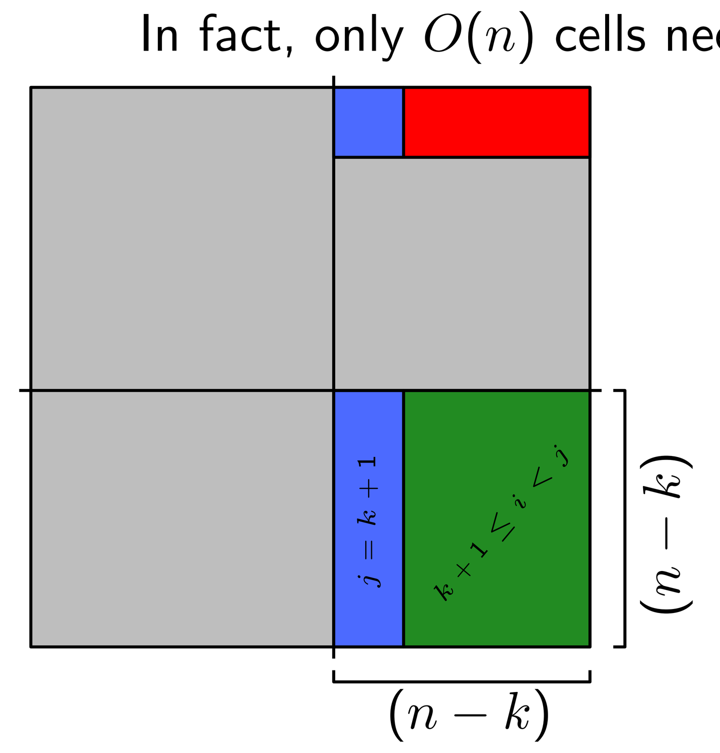
Expand the recurrence at $\alpha_{0,n+1}^n$:



$$\alpha_{i,j}^k = \alpha_{i,j}^{k-1} + \alpha_{i,k}^{k-1}(\alpha_{k,k}^{k-1})^* \alpha_{k,j}^{k-1}$$

We only need to consider cells that eventually contribute to $\alpha_{0,n+1}^n$.

In fact, only $O(n)$ cells need to be evaluated at each step.



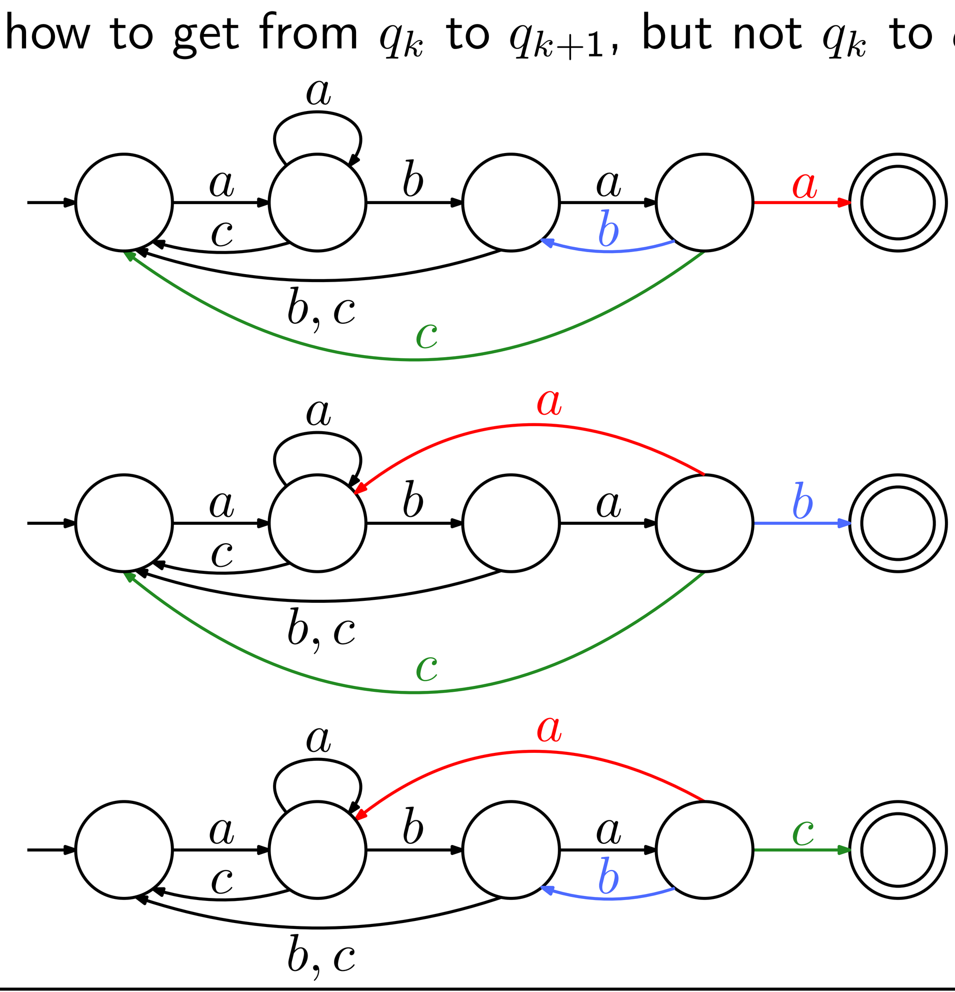
- $\alpha_{i,j}^k$: Ignore
- $\alpha_{i,j}^k$: Compute normally
- $\alpha_{i,j}^k := \alpha_{i,j}^{k-1}$
- $\alpha_{i,j}^k$: Always $\emptyset$
- New runtime: $O(|w|^2 |Q_\mathcal{P}|^m)$

Always $\emptyset$

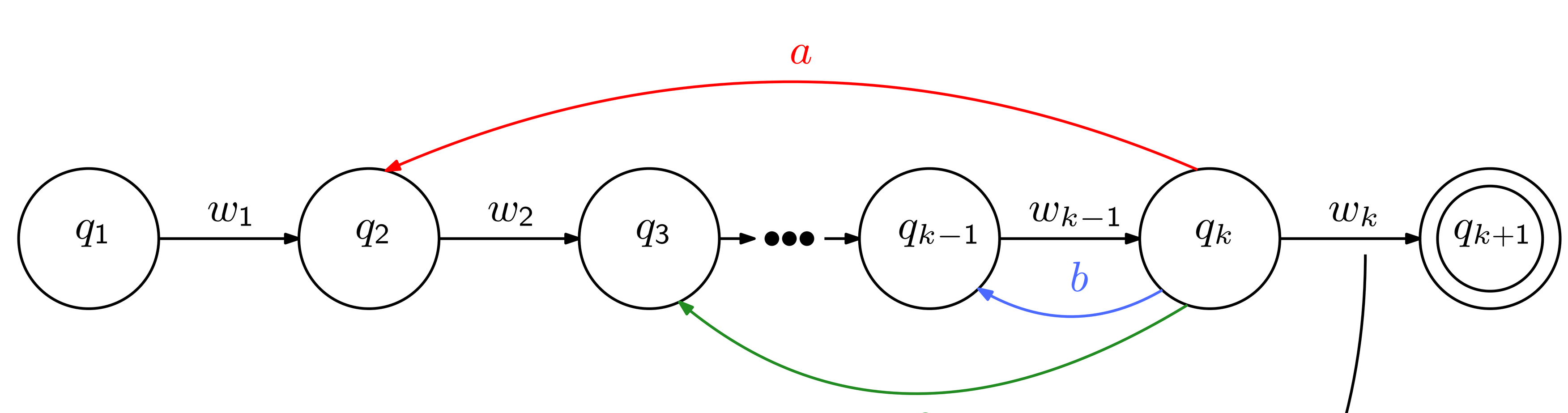
$$\underbrace{\alpha_{k+x,k+y}^k}_{0<x<y} = \alpha_{k+x,k+y}^{k-1} + \alpha_{k+x,k}^{k-1}(\alpha_{k,k}^{k-1})^* \alpha_{k,k+y}^{k-1}$$

Online Algorithm

Issue: We can't predict where back-transitions will go. At step $k+1$, we know how to get from q_1 to q_k and how to get from q_k to q_{k+1}, but not q_k to q_k.



Key idea: Prepend $c \in \Sigma - w_k$ to each path from state $\delta(q_{k+1}, c)$ to k. Now we can reconstruct all $q_k \to q_k$ paths without ever knowing the out-transitions beforehand.



$$\alpha_{0,k+1}^k = \alpha_{0,k+1}^{k-1} + \alpha_{0,k}^{k-1}(\alpha_{k,k}^{k-1})^* \alpha_{k,k+1}^{k-1}$$

Always $\emptyset$

Known from previous iteration

$$\alpha_{k,k}^{k-1} = a(\alpha_{q_2,k}^{k-1}) + b(\alpha_{q_{k-1},k}^{k-1}) + c(\alpha_{q_3,k}^{k-1})$$

Online iteration: $O((|\Sigma| + k)|Q_\mathcal{P}|^m)$

Experimental Timings

$\|Q\|, \|\Sigma\|$ / Alg / $\|w\|$	1500, 26 Alg 1	1500, 26 Faster	1500, 26 Online	1500, 100 Alg 1	1500, 100 Faster	1500, 100 Online
1	13.396	1.780	1.201	13.371	1.720	1.605
2	13.382	1.649	1.320	13.382	1.570	1.750
3	13.154	1.446	1.459	13.290	1.447	1.849
4	13.333	1.295	1.609	13.342	1.273	1.986
5	13.378	1.161	1.763	13.319	1.143	2.135
6	14.352	1.002	1.898	13.282	0.994	2.254
7	14.287	0.869	2.056	13.571	0.832	2.368
8	14.330	0.735	2.189	13.614	0.702	2.479
9	14.673	0.591	2.367	13.661	0.568	2.679
10	13.847	0.447	1.596	13.627	0.445	1.507
Total	137.947	10.976	1.365	134.462	10.694	20.615

An important use case:
Given a string w, find a such that the infix of wa is maximized.

... be or not to ... $\to$ *... be or not to be ...*

Consider the $|Q| = 1500$, $|\Sigma| = 100$ case with $|w| = 9$.
Old method $\approx 134.462 * 100 = 13446.2$ seconds
Faster method $\approx 10.694 * 100 = 1069.4$ seconds
Online method $\approx (20.615 - 1.507) + 1.507 * 100 = 169.808$ seconds

This work was supported by the Institute for Information & Communications Technology Promotion (IITP) grant funded by the Korea government (MSIP) (2018-0-00247).