Минимизация автомата Ахо-Корасик. Приложение к вычислению *P*-значения вхождений образцов

Фурлетова Е.И.

ИМПБ РАН, Пущино, Россия

октябрь 2022г.

Введение: Автомат Ахо-Корасик

Автомат Ахо-Корасик предложен в 1975г. Альфредом Ахо и Маргарет Корасик [1] для поиска вхождений слов из заданного набора в строке.

Применение:

- Компьютерная лингвистика
- Компьютерная безопасность
- Биоинформатика
- и др. области компьютерных наук

Применение в биоинформатике. *P*-значение вхождений образцов

Автомат Ахо-Корасик используют в алгоритмах нахождения *P*-значения вхождений слов.

***P*-значение** – вероятность встретить слова из данного набора не менее заданного числа раз в случайной последовательности заданной длины.

P-значение применяется при оценке достоверности обнаружения кластеров функционально-значимых фрагментов в биологических последовательностях.

Например, сайтов связывания факторов регуляции транскрипции.

Автомат Ахо-Корасик

Пусть Σ – конечный алфавит; $H = \{h_1, ..., h_n\}$ – набор слов в Σ.

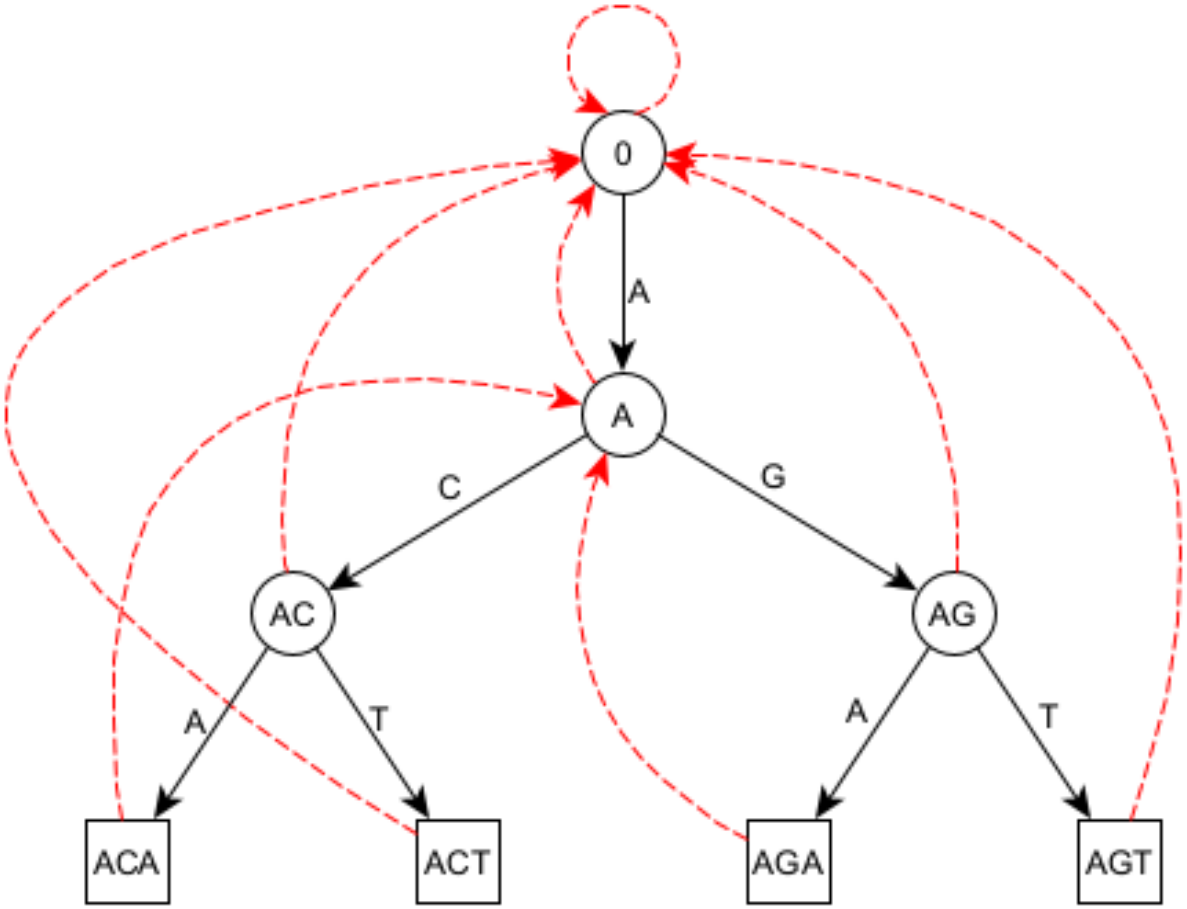
Автомат Ахо-Корасик – пятерка $(Q, \Sigma, \delta, q_0, F)$, где

- Q – множество состояний, состоящее из всех префиксов слов из H;
- q_0 – начальное состояние;
- F – множество допускающих состояний;
- δ – функция переходов; $\delta(q,a) = qa$, если $qa \in Q$ (прямой переход); иначе, $\delta(q,a) = sl(qa)$ (обратный переход).

$sl(x)$ (суффиксная ссылка) – максимальный суффикс x, который является префиксом некоторого слова из H.

Автомат Ахо-Корасик распознаёт язык $\Sigma^* \cdot H$.

Автомат Ахо-корасик для *H* = A{C,G}{A,T}

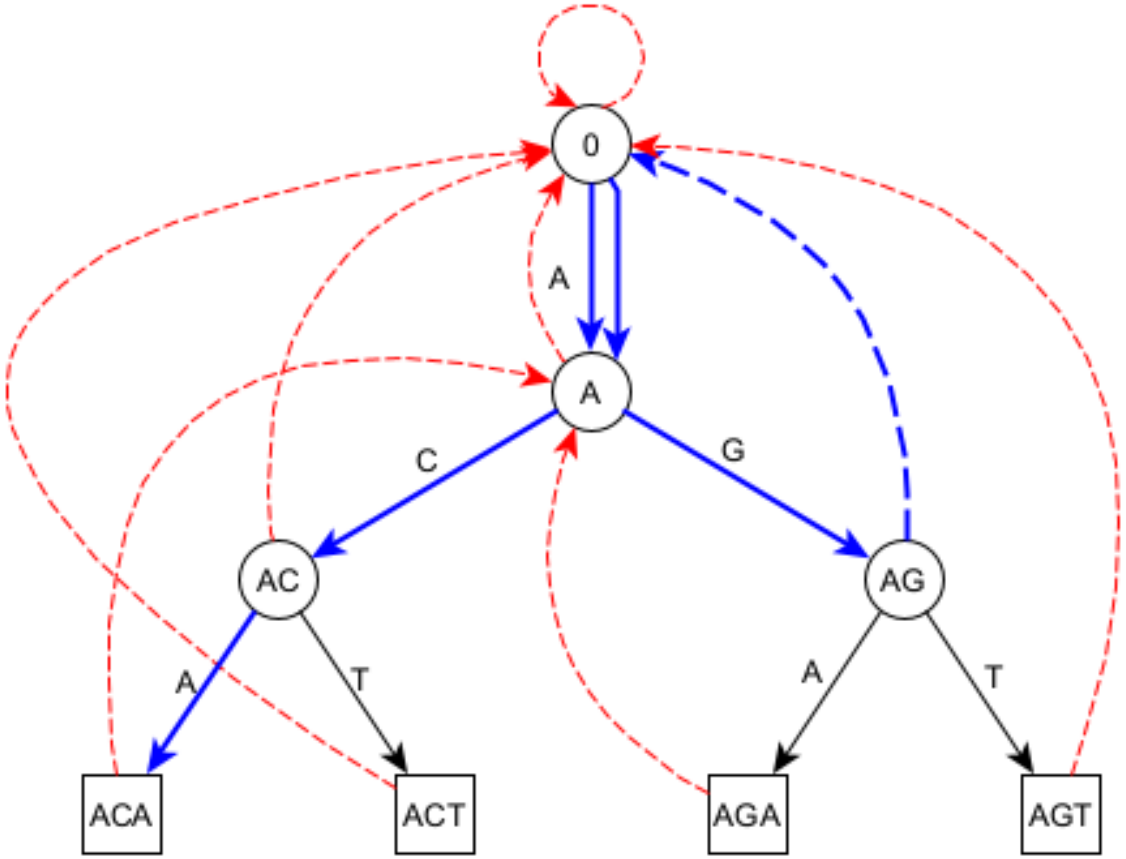


H – набор, состоящий из трехбуквенных слов, на первом месте в которых стоит буква A, на втором – C или G, на третьем – A или T.

Красными дугами отмечены суффиксные ссылки, черными стрелками – прямые переходы. Обратные переходы находятся по суффиксным ссылкам.

Автомат Ахо-корасик для *H* = A{C,G}{A,T}.

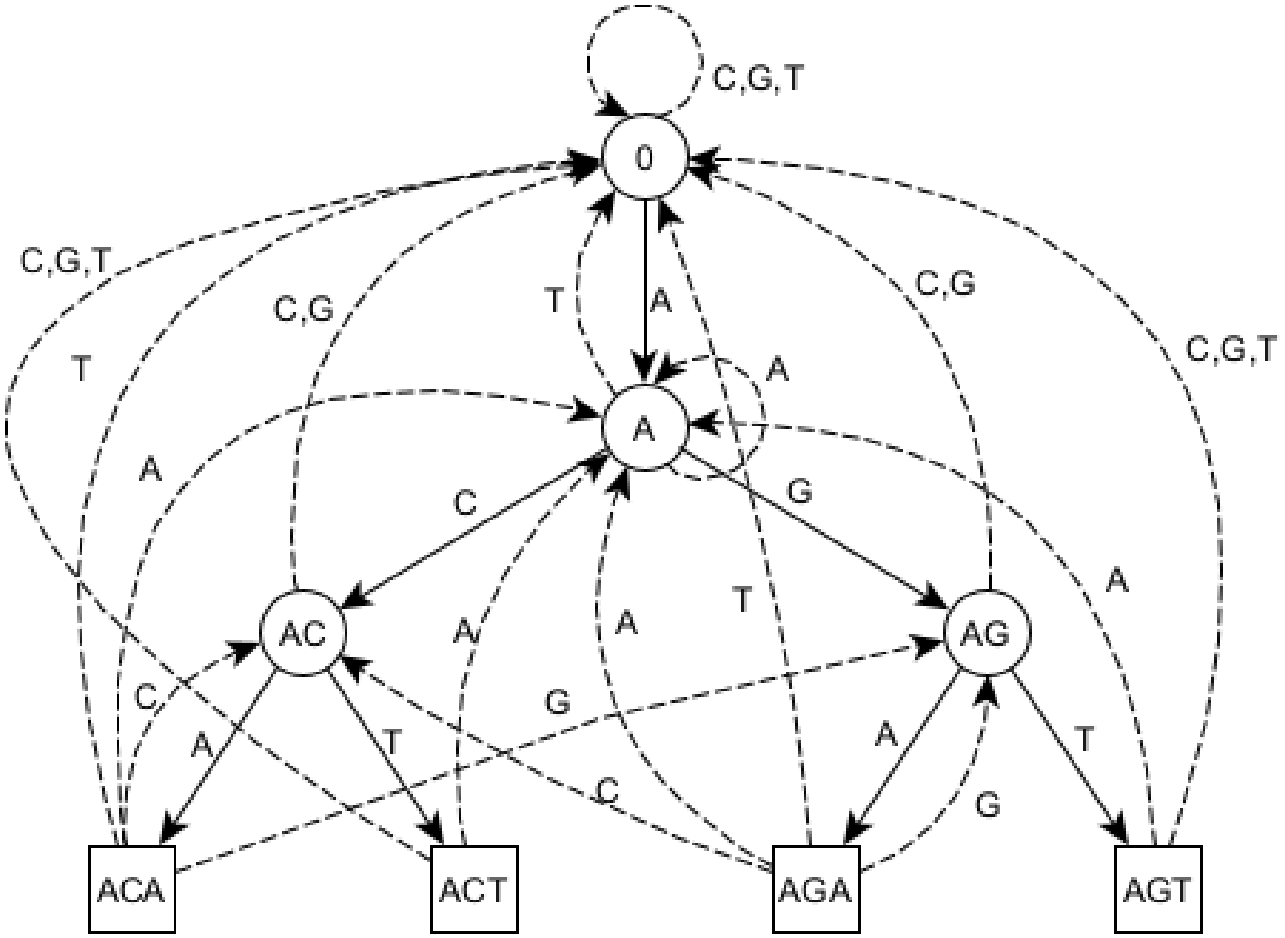
Поиск вхождения *H* в последовательности AGACA



Траектория переходов автомата при поиске отмечена синим цветом. Если автомат переходит в допускающее состояние, то имеет место вхождение H.

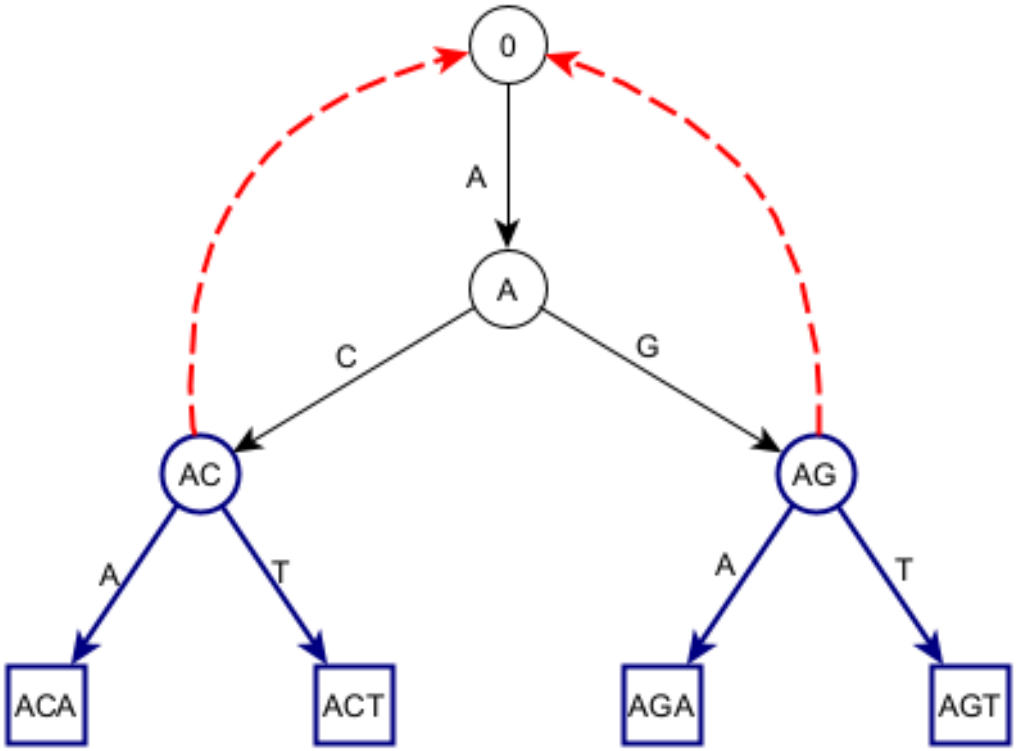
Автомат Ахо-корасик для *H* = A{C,G}{A,T}.

Полный граф автомата.



Пунктирными дугами отмечены обратные переходы, прямыми стрелками – прямые переходы.

Эквивалентные состояния



Эквивалентные поддеревья с вершинами AC и AG.

~ -эквивалентность

Состояния x и y из Q являются ~-эквивалентными ($x \sim y$), если и только если $x = y$ или для любого слова t из Σ^* одновременно выполняется:

1. $xt \in F \Leftrightarrow yt \in F$;
2. $sl(x) \sim sl(y)$.

Если выполняется первое условие, то будем говорить, что состояния префиксно-эквивалентны.

Состояния AC и AG вышеуказанного автомата являются ~ -эквивалентными.

~ -эквивалентность и эквивалентность Нероуда

Состояния x и y конечного автомата **эквивалентны** в классическом смысле (эквивалентны по Нероуду), если для любого t из Σ^*

$$xt \in L \Leftrightarrow yt \in L,$$

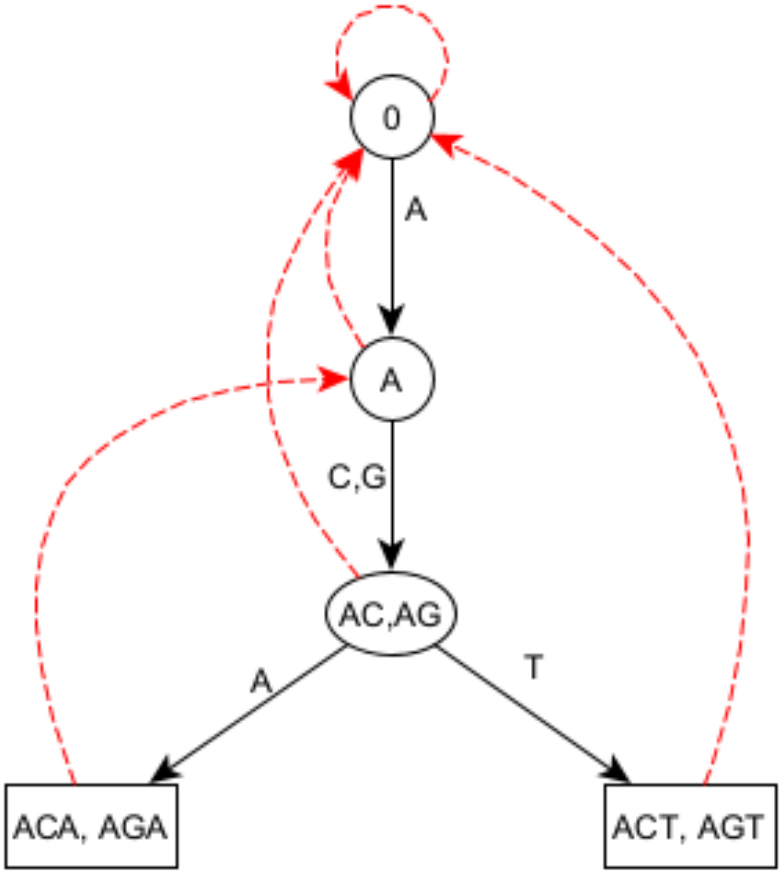
где $L = \Sigma^* H$.

Минимальным называется автомат, который не содержит различных эквивалентных состояний

Показано, что ~-эквивалентные состояния являются эквивалентными по Нероуду, обратное неверно.

В ряде случаев ~-эквивалентность тождественна эквивалентности Нероуда. Например, для наборов, состоящих из слов одинаковой длины.

Минимальный автомат Ахо-Корасик



В данном случае ~-минимальный автомат совпадает с минимальным.
Эквивалентные состояния имеют эквивалентные суффиксные ссылки, на графе они заменены одной.

Наборы слов одинаковой длины

Примерами таких наборов являются мотивы биологических последовательностей, заданные:

- Матрицей позиционных весов (PWM)
- Словом-шаблоном (Consensus, signature, degenerate pattern), буквы которого – множества букв алфавита, которые могут стоять в словах набора в соответствующих позициях.

Если набор задан словом-шаблоном, то состояния автомата равной длины являются префиксно-эквивалентными.

Наборы, заданные словом-шаблоном

- семейство белков GAS_VESICLE_C (база данных PROSITE), заданное шаблоном FLXXTXXXRXXXAXXQXXXLXXF (X – произвольная аминокислота)

автомат Ахо-Корасик: ≈10^30 состояний

минимальный автомат: 1480 состояний

- шаблон TGTTTCCN(18)TGTTTCT (N – произвольный нуклеотид) из базы данных YEASTRACT

автомат Ахо-Корасик: ≈ 10^12 состояний

минимальный автомат: 754 состояния

Алгоритм построения ~-минимального автомата Ахо-Корасик

Шаг 1. Находим префиксно-эквивалентные состояния путем обхода автомата по прямым переходам от допускающих состояний к начальному состоянию.

Шаг 2. Разбиваем классы префиксной эквивалентности на группы, где состояния находятся в одной группе, если они имеют эквивалентные суффиксные ссылки.

Если набор задан словом-шаблоном, то сразу выполняется Шаг 2.

Время работы алгоритма (общий случай): $O(|\Sigma| \cdot S)$

Время работы алгоритма (набор задан словом-шаблоном): $O(|\Sigma| \cdot S_m)$

Время работы алгоритма Хопкрофта : $O(|\Sigma| \cdot \log(S) \cdot S)$

S – число состояний автомата Ахо-Корасик. S_m – число состояний минимального автомата.

Применение автомата Ахо-Корасик при вычислении P-значения

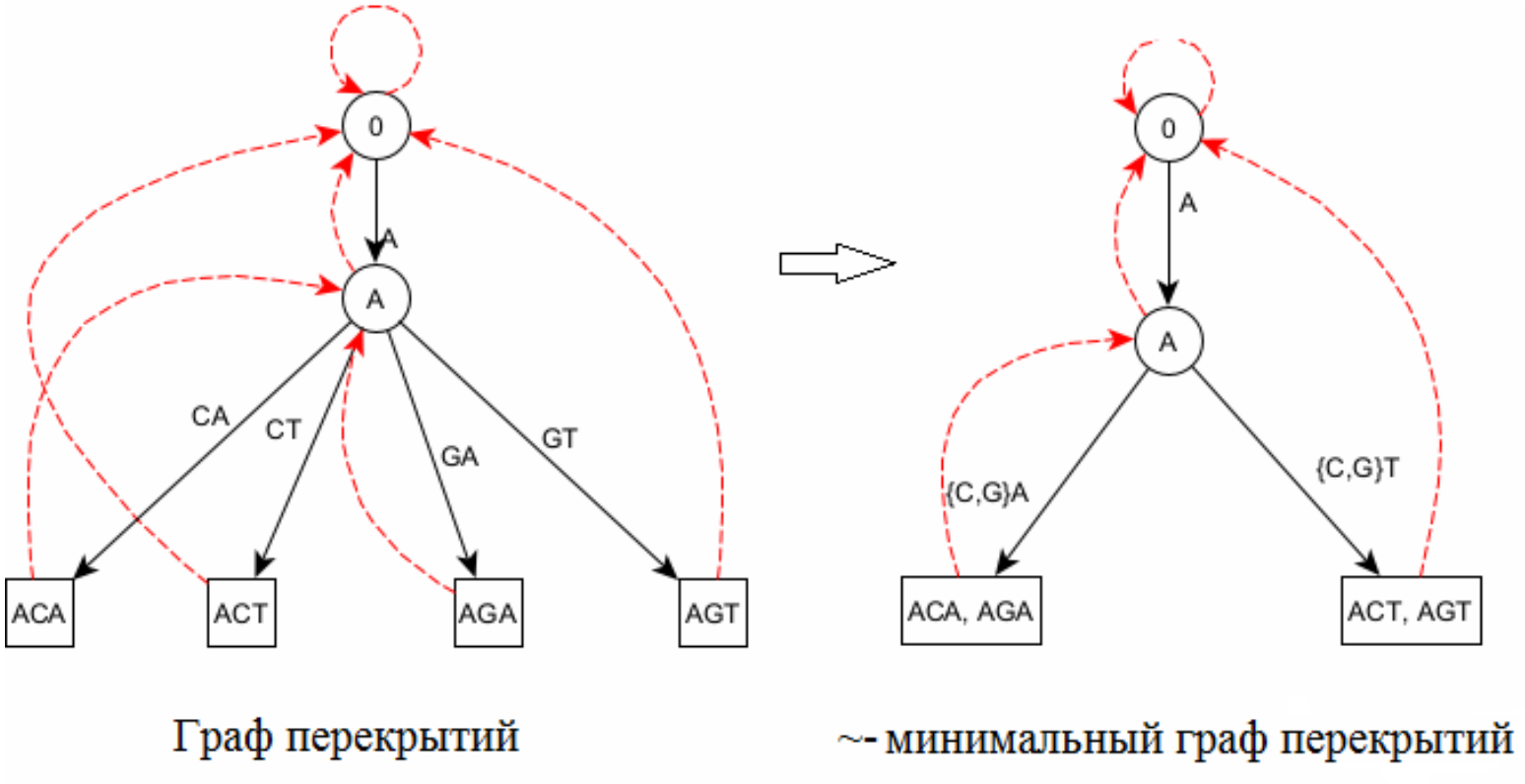
Автомат Ахо-Корасик и производные от него структуры используются в следующих алгоритмах вычисления *P*-значения:

- **AhoPro** [2]
- **SufPref** [3]
- **AutoClump** [4]
- и другие [5]

Сложности алгоритмов пропорциональны размерам используемых структур

Алгоритмы можно оптимизировать, используя ~-минимальные автоматы

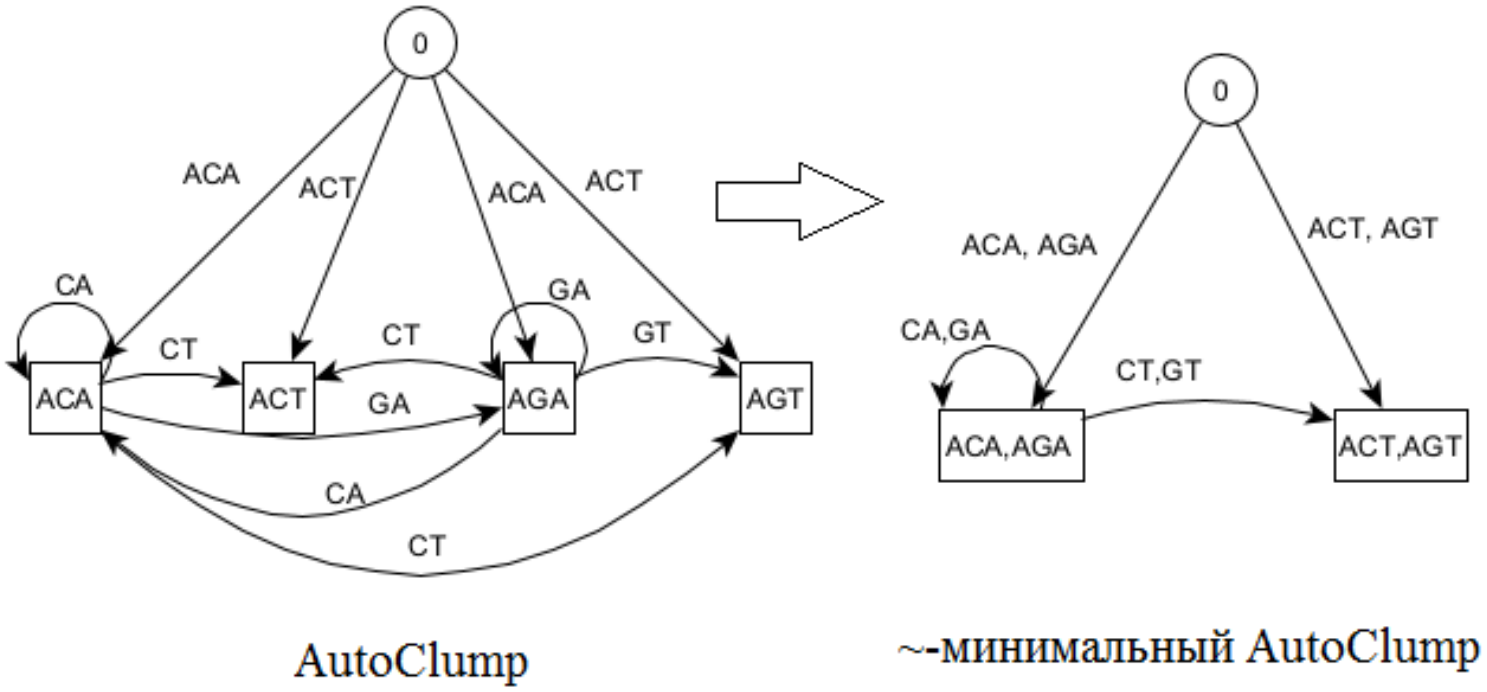
Минимальный Граф перекрытий (SufPref)



Внутренние вершины графа перекрытий – перекрытия (перекрытие – слово, которое является суффиксом одного слова из *H* и префиксом другого слова.) Листья – слова из *H*.

Минимальный граф перекрытий строится по минимальному автомату Ахо-Корасик

Минимальный AutoClump



AutoClump

~-минимальный AutoClump

Листья – слова из *H*. Между двумя словами из *H* есть переходы, если конец первого слова является началом второго.

Минимальный AutoClump строится по минимальному автомату Ахо-Корасик

Список литературы

1. Aho A.V., Corasick M.J. Efficient string matching: An aid to bibliographic search. *Communications of the ACM*. 1975. V. 18. P. 6.
2. Boeva V., Clément J., Régnier M., et all. Exact p-value calculation for heterotypicclusters of regulatory motifs and its application in computational annotation of cisregulatory modules. *Algorithms Mol Biol*. 2007. V. 2. P. 13. doi: 10.1186/1748-7188-2-13.
3. Régnier M., Furletova E., Yakovlev V., Roytberg M. Analysis of pattern overlaps and exact computation of *P*-values of pattern occurrences numbers: Case of Hidden Markov Models. *Algorithms Mol Biol.* 2014. V. 9. № 25. doi: 10.1186/s13015-014-0025-1.
4. M. Regnier, B. Fang, and D. Iakovishina. Clump combinatorics, automata, and word asymptotics. *ANALCO.* 2014. P. 62–73. doi: 10.1137/1.9781611973204.6.
5. Lothaire. M. Statistics on words with applications to biological sequence. *Applied combinatorics on words*. (Encyclopedia of Mathematics and its Applications). Cambridge: Cambridge University Press. 2005. V. 105. P. 268–352. doi: 10.1017/CBO9781107341005.

Соавторы

Mireille Regnier
Ecole Polytechnique, INRIA, France

Jan Holub
Czech Technical University in Prague, Czech Republic

Спасибо за внимание ☺