Chapter 5

Asymptotic Properties of Suffix Trees.

Ivan Kazmenko

Unlike the previous chapters, this one is not going to introduce a new sophisticated suffix tree construction algorithm, dig into its properties and prove that it works fast and fine. Instead, we'll consider one of the most dumb algorithms of suffix tree construction and find out that under certain conditions on the text, it almost surely works rather well, meaning that we can find almost sure upper and lower bound for the complexity of new suffix insertion while the size of the text tends to infinity.

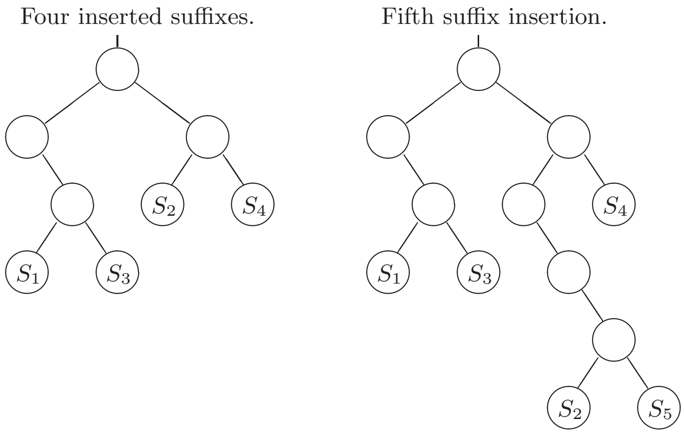
This chapter is based on the article *Asymptotic Properties of Data Compression And Suffix Trees* by Wojciech Szpankowski [Szp93] and the book *Average case analysis of algorithms on sequences* [Szp00] of the same author.

5.1 Suffix Tree Construction

Let's start with some common symbol definitions which we will use in the entire chapter.
Let Σ is a finite alphabet of size $|\Sigma| = V$, $\{X_k\}_{k=1}^{\infty}$ be a stationary ergodic sequence of symbols generated from Σ, and $X_m^n = (X_m, ..., X_n)$ for $m < n$ be a partial sequence of the whole sequence $\{X_k\}_{k=1}^{\infty}$.
We shall now consider a very simple algorithm of suffix tree construction. A node of our tree can be either internal, i. e. branching node, or external node storing one of the suffixes $S_i = \{X_k\}_{k=i}^{\infty}$. Each edge is labeled by some symbol from Σ. When adding a suffix, we start from the root of our tree and try to 'align' the suffix to the tree, that is, move by the edge corresponding to the current symbol of our suffix and change the current symbol to the next one in the suffix. That procedure continues until we find no such edge at the vertex we are currently in. We then add that edge and create a new vertex at its end storing the suffix we were adding.
More formally, consider a digital tree built in the following way:
Step 0. At the beginning, the tree consists of its root only.
Step 1. Consider a tree $\mathcal{T}_n$ built for the partial sequence $X_1^n = (X_1, ..., X_n)$.
Step 2. Set current vertex to root.
Step 3. Starting with $j = n + 1$, we either

(A) move by the edge marked by X_j from the current vertex if it exists thus changing the current vertex and increase j by 1, or
(B) construct a new edge marked with symbol X_j from the current vertex to a new vertex marked with our suffix X_{n+1}^{∞} and proceed to *Step 1* with n increased by 1 otherwise.
Note that $j - n$ is the number of case *(A)* occurences during a single *Step 3*.
The picture shows an example of a single loop of our algorithm.

Let $X_1^{10} = (0, 1, 0, 1, 1, 0, 1, 1, 1, 0)$.



$S_1 = 0101101110$
$S_2 = 101101110$
$S_3 = 01101110$
$S_4 = 1101110$
$S_5 = 101110$

We do not formalize our 'splitting policy', that is, the way how we split an external node that becomes internal during some other suffix insertion. The natural way to do the 'splitting' is shown on the picture. We can consider all previous suffix marks to be infinite branches of our tree to make the algorithm formally correct.
We are interested in the complexity of a single loop of our algorithm. Formally, our main questions regarding the algorithm described will be the following:
What is the typical height of $\mathcal{T}_n$?
What is the typical difference $j - n$ when Step 3 is finished?
What is the typical minimal possible difference $j - n$ at the end of Step 3 for the tree $\mathcal{T}_n$?
In the next section, we will present some assumptions on the sequence $\{X_k\}_{k=1}^{\infty}$ that, being not too restrictive, will get us some bounds on the value in question.

5.2 Depth of Insertion in a Suffix Tree

As we study our sequence $\{X_k\}_{k=1}^{\infty}$ in a probabilistic framework, its most important characteristic is nth order probability distribution $P(X_1^n) = Pr\{X_k = x_k, 1 \leqslant k \leqslant n, x_k \in \Sigma\}$. The entropy of our sequence is the limit $h = \lim\limits_{n\to\infty} \frac{E\{-\log P(X_1^n)\}}{n}$. It is known that $h \leqslant \log V$. All logarithms are natural ones in this chapter.
Another characteristic of much interest is the parameter L_n which is the smallest integer $L > 0$ such that $X_m^{m+L-1} \neq X_{n+1}^{n+L}$ for all $1 \leqslant m \leqslant n$. Informally, it has the following meaning: when we insert the suffix S_{n+1}, we will require exactly L_n steps *(A)* to do it.
Returning to our example, let $X_1^{10} = (0, 1, 0, 1, 1, 0, 1, 1, 1, 0)$. Here $L_1 = 1$, $L_2 = 3$, $L_3 = 2$, and $L_4 = 5$ since $X_5^8 = X_2^5 = (1, 0, 1, 1)$ and therefore $L_4 > 4$.

So, what will be our assumptions on the sequence $\{X_k\}$? Below we introduce the mixing condition - a weakened form of independence.

Remember that $\{X_k\}$ is called an independent sequence if for every set of indexes $I = \{i_1, \dots, i_r\}$ the probablity of $\{X_k\}_{k\in I}$ being in $\bigotimes\{A_k\}_{k=1}^r$ is equal to the product of the corresponding probabilities: $Pr\{X_{i_1} \in A_1, \dots, X_{i_r} \in A_r\} = Pr\{X_{i_1} \in A_1\} \dots Pr\{X_{i_r} \in A_r\}$. Somewhat weaker is pairwise independent condition which takes only the sets I of size 2 into consideration, stating that $Pr\{X_{i_1} \in A_1, X_{i_2} \in A_2\} = Pr\{X_{i_1} \in A_1\}Pr\{X_{i_2} \in A_2\}$. The independence itself can be also written in pairwise form with events being not subsets of a single copy of Σ, but elements of a more complex σ-field.

Let F_m^n be a σ-field (also known as σ-algebra) generated by $\{X_k\}_{k=m}^n$ for $m \leqslant n$. Independence means that for every pair of events $A \in F_0^m$ and $B \in F_{m+1}^\infty$ it is true that $Pr\{AB\} = Pr\{A\}Pr\{B\}$. The mixing condition creates a gap of size d between our σ-fields so that $A \in F_0^m$ and $B \in F_{m+d}^\infty$ and transforms our equality into two inequalities bounding the left term with the right one multiplied by some constants from both sides. The strong α-mixing condition substitutes that constants by functions tending to 1 from both sides while the gap size d tends to infinity. The formal definitions follow.

We say that $\{X_k\}$ satisfies the *mixing condition* if and only if there exist constants $0 < c_1 \leqslant c_2$ and an integer d such that for all $A \in F_0^m$, $B \in F_{m+d}^\infty$ and $0 \leqslant m \leqslant m+d \leqslant n$ the following condition is true: $c_1 Pr\{A\}Pr\{B\} \leqslant Pr\{AB\} \leqslant c_2 Pr\{A\}Pr\{B\}$.

Now let α be a function of d such that $\alpha(d) \xrightarrow[d\to\infty]{} 0$. $\{X_k\}$ satisfies the *strong α-mixing condition* if and only if for all $A \in F_0^m$, $B \in F_{m+d}^\infty$ and $0 \leqslant m \leqslant m+d \leqslant n$ the following condition is true: $(1-\alpha(d))Pr\{A\}Pr\{B\} \leqslant Pr\{AB\} \leqslant (1+\alpha(d))Pr\{A\}Pr\{B\}$.

We define two new parameters of $\{X_k\}$. They are parameters h_1 and h_2:

$h_1 = \lim\limits_{n\to\infty} \frac{\max\{\log P^{-1}(X_1^n),\ P(X_1^n)>0\}}{n} = \lim\limits_{n\to\infty} \frac{\log(1/\min\{P(X_1^n),\ P(X_1^n)>0\})}{n}$,

$h_2 = \lim\limits_{n\to\infty} \frac{\log(E\{P(X_1^n)\})^{-1}}{2n} = \lim\limits_{n\to\infty} \frac{\log(\sum_{X_1^n} P^2(X_1^n))^{-1}}{2n}$.

The relationship with entropy h is as follows: $0 \leqslant h_2 \leqslant h \leqslant h_1$. The values h_1 and h_2 are also known as Rényi entropy of order $-\infty$ and 2, respectively.

The formulas are complex, so we could use a simple example, Bernoulli model, to see what these values are like.

Assume that symbols X_i are generated indepenently, and ith symbol is generated according to the probability p_i. Thus, $h = \sum\limits_{i=1}^{V} p_i \log(p_i^{-1})$, $h_1 = \log(1/p_{min})$ and $h_2 = 2\log(1/P)$ where $p_{min} = \min\limits_{1\leqslant i\leqslant V}\{p_i\}$ is the probability of least probable symbol occurence and $P = \sum\limits_{i=1}^{V} p_i^2$ can be interpreted as a probability of a match between any two symbols.

Now, we are ready to present our main result, Theorem 5.1. It proposes the conditions under which we can find almost sure lower and higher bounds for L_n, the value we are interested in. An important finding is that we not only know how it behaves (its behavior is logarithmic with respect to n), but also find the range of the constant by that logarithm.

Theorem 5.1. *Consider stationary ergodic sequence* $\{X_k\}_{k=-\infty}^\infty$.

1. Assume strong α-mixing condition.

2. Let $h_1 < \infty$ and $h_2 > 0$.

$(\ast)$ $\exists\rho : 0 < \rho < 1, \exists\beta$ *such that* $\alpha(d) = O(d^\beta\rho^d)$ *for* $d \to \infty$.

Then

(1) $\liminf\limits_{n\to\infty} \frac{L_n}{\log n} = \frac{1}{h_1}$ *(a.s.),*

(2) $\limsup\limits_{n\to\infty} \frac{L_n}{\log n} = \frac{1}{h_2}$ *(a.s.).*

How restrictive is the condition $(*)$? Many practically occuring cases fit it, for example, in Bernoulli model, $\alpha(d) = 0$ because of independence of X_k, and if the sequence $\{X_k\}$ is Markovian, $\alpha(d)$ decays exponentially fast. In general, statement (1) of Theorem 5.1 does not hold without the $(*)$ condition.

5.3 Height and Shortest Feasible Path in a Suffix Tree

In this section, we will introduce yet another bundle of auxiliary definitions to formulate our Theorem thm:kaz-2, and then prove Theorem 5.1 using Theorem 5.2. The proof of Theorem 5.2 itself will not be given due to its complexity, however, a short overview of its proof techniques will be done in Section 5.4.
Let us define some more depth characteristics. Let $\mathcal{T}_n$ be a suffix tree constructed from the first n suffixes of $\{X_k\}$. *mth depth* $L_n(m)$ is the depth of the ith suffix in $\mathcal{T}_n$; note that $L_n = L_{n+1}(n+1)$. *Average depth* D_n is the depth of a randomly selected suffix, that is, $D_n = \frac{1}{n}\sum\limits_{m=1}^{n} L_n(m)$.
Height and shortest feasible path are defined as follows. *Height* H_n is the length of the longest path in $\mathcal{T}_n$; $H_n = \max\limits_{1\leqslant m\leqslant n}\{L_n(m)\}$. *Available node* is a node which does not belong to $\mathcal{T}_n$ but its predecessor does, that is, a node that could be inserted in $\mathcal{T}_{n+1}$ at the next insertion with no other nodes added. *Shortest feasible path* s_n is the length of the shortest path from the root to an available node.
For each two suffixes, we can find their longest common prefix by walking down the tree along them till they part. *Self-alignment* $C_{i,j}$ is the length of the longest common prefix of S_i and S_j.
One can easily prove the following relations of self-alignment to other suffix tree parameters:
$L_n(m) = \max\limits_{1\leqslant k\leqslant n, k\neq m}\{C_{k,m}\} + 1$;
$H_n = \max\limits_{1\leqslant i<j\leqslant n}\{C_{i,j}\} + 1$;
$L_n = \max\limits_{1\leqslant m\leqslant n}\{C_{m,n+1}\} + 1$.
Returning to our example, let $X_1^{10} = (0,1,0,1,1,0,1,1,1,0)$. Consider suffix tree $\mathcal{T}_4$ built from first 4 suffixes. $L_4(1) = 3$, $L_4(2) = 2$, $L_4(3) = 3$, $L_4(4) = 2$. $H_4 = 3$, $s_4 = 2$. But $L_4 = L_5(5) = 5$.
Note that the S_5 node of $\mathcal{T}_5$ is not an available node in $\mathcal{T}_4$ since it requires auxiliary internal nodes to be inserted. In $\mathcal{T}_5$, $H_5 = 5$, and $s_5 = 2 = s_4$.
Digging into the properties of $C_{i,j}$ gives the proof of Theorem 5.2 formulated below. It is a variant of Theorem 5.1 with L_n substituted by s_n and H_n. As we already observed, the statement (2) of the theorem does not need $(*)$ condition to hold.

Theorem 5.2. *Consider stationary ergodic sequence $\{X_k\}_{k=1}^{\infty}$.*
1. Assume strong α-mixing condition.
2. Let $h_1 < \infty$ and $h_2 > 0$.
Then
(1) $\lim\limits_{n\to\infty}\frac{s_n}{\log n} = \frac{1}{h_1}$ *(a.s.) when $(*)$ holds,*
(2) $\lim\limits_{n\to\infty}\frac{H_n}{\log n} = \frac{1}{h_2}$ *(a.s.) when $\alpha(d)$ satisfies the following:* $\sum\limits_{d=0}^{\infty}\alpha^2(d) < \infty$.

Proof of Theorem 5.1 by Theorem 5.2: For each of the two statements, we will bound the left side of equality by the right side from both sides.
(1): $\limsup\limits_{n\to\infty}\frac{L_n}{\log n} \leqslant \lim\limits_{n\to\infty}\frac{H_n}{\log n}$ (a.s.) simply holds by definition as $L_n \leqslant H_n$; let's prove that $\limsup\limits_{n\to\infty}\frac{L_n}{\log n} \geqslant \lim\limits_{n\to\infty}\frac{H_n}{\log n}$ (a.s.). Note that H_n is a non-decreasing sequence;

$L_n = H_n$ when $H_{n+1} > H_n$, and that occurs infinitely often since $H_n \to \infty$ and $\{X_k\}$ is an ergodic sequence, so $Pr\{L_n = H_n \ i.o.\} = 1$ and there exists a subsequence $n_k \to \infty$ such that $L_{n_k} = H_{n_k}$. It is clear now that the upper limit of L_n in not less than the limit of H_n with an arbitrary common denominator, which is equal to $\log n$ in our case.
(2) can be proved in a similar way: s_n is a non-decreasing sequence also.

□

5.4 Proof Techniques

In this section, we will throw a short glance on the tools used to prove Theorem 5.2 itself. The whole proof is complex and technically hard.
One of the methods used in the proof is a technique called *String-Ruler Approach*. According to it, the correlation between different substrings is measured using another string ω called a string-ruler. To illustrate it, we shall find the longest common prefix of two independent strings $\{X_k(1)\}_{k=1}^{\infty}$ and $\{X_k(2)\}_{k=1}^{\infty}$. Let its length be $C_{1,2}$. The following equivalence is obvious:
$C_{1,2} \geqslant k \Longleftrightarrow \exists\omega$ of length k: $X_1^k(1) = \omega = X_1^k(2)$.
We then construct a set $\mathcal{W}_k = \{\omega \in \Sigma^k : |\omega| = k\}$ and estimate the probabilities $P(\omega_k) = P(X_{m+1}^{m+k} = \omega_k)$ for a fixed position m in our sequence $\{X_k\}$.
Another important method is a probabilistic one, called *Second Moment Method*. The version by Chung and Erdös of this method states that for a sequence of events A_i we have
$Pr\{\bigcup_{i=1}^{n} A_i\} \geqslant \frac{(\sum_{i=1}^{n} Pr\{A_i\})^2}{\sum_{i=1}^{n} Pr\{A_i\} + \sum_{i \neq j} Pr\{A_i \cap A_j\}}$.
We then apply it to the sets $A_{i,j} = \{C_{i,j} \geqslant k\}$.
The reasoning of the latter method is elementary. Let us remember Markov's inequality
$Pr\{X \geqslant t\} \leqslant \frac{E\{X\}}{t}$
and Chebyshev's inequality
$Pr\{|X - E\{X\}| \geqslant t\} \leqslant \frac{Var\{X\}}{t^2}$.
After some trivial calculations we get First Moment Method:
for integer-valued nonnegative random variable X
$Pr\{X > 0\} \leqslant E\{X\}$
and Second Moment Method (Chebyshev's version):
$Pr\{X = 0\} \leqslant \frac{Var\{X\}}{(E\{X\})^2}$,
respectively. The version by Chung and Erdös is derived from the latter one.

5.5 Summary

In our main result, Theorem 5.1, we have shown that, given a stationary ergodic sequence generated over a finite alphabet, under strong α-mixing condition on the sequence, the depth of the nth suffix insertion into a partial suffix tree of that sequence using simple and natural algorithm specified above can be described by the expression $c \log n$ where c almost surely lies between $1/h_1$ and $1/h_2$ and the parameters h_1 and h_2 can be found explicitly.