# Chapter 8

# Analysis of Pattern Occurances

Roland Aydin

This paper will summarize the proof for the formula to compute the expected number of occurrences of a given pattern $H$ in a text of size $n$. The intuitive solution of $E[O_n(H)] = P(H)(n - m + 1)$ will be verified utilising generating functions. Frequency analysis will rely on the decomposition of the text $T$ onto languages, the so-called initial, minimal, and tail languages. Going from there to their generating functions both for a Markovian and a Bernoulli environment, the formula will be shown to work due to properties of the respective generating functions.

## 8.1 Preliminaries

### Markov sequence

A sequence $X_1, X_2, ...$ of random variates is called a *Markov sequence* of order 1 iff, for any $n$,

$$F(X_n|X_{n-1}, X_{n-2}, ...X_1) = F(X_n|X_{n-1})$$

i.e., if the conditional distribution $F$ of $X_n$, assuming $X_{n-1}, X_{n-2}, ...X_1$ equals the conditional distribution $F$ of $X_n$ assuming only $X_{n-1}$.

### Markov chain

If a Markov sequence of random variates $X_n$ take the *discrete values* $a_1, ..., a_N$ then

$$P(x_n = a_{in}|x_{n-1} = a_{in-1}, ..., x_1 = a_{i1}) = P(x_n = a_{in}|x_{n-1} = a_{in-1})$$

and the sequence $x_n$ is called a *Markov chain* of order 1.

## Correlation of patterns

A *correlation* of two patterns $X$ (size m) and $Y$ is a string, denoted by $XY$, over the set $\Omega = \{0, 1\}$.

$$|XY| = |X|$$

Each position $i$ can be computed as

$$i = 1 \Leftrightarrow \text{place } Y \text{ at } X_i \wedge \text{ all overlapping pairs are identical} \text{else} \qquad i = 0$$

## Example of pattern correlation

Let $\Omega = \{M, P\}$, $X = MPMPPM$ and $Y = MPPMP$. Then $XY$ can be deduced in the following manner:

```
X: HTHTTH
Y: HTTHT        0
     HTTHT      0
       HTTHT    1
         HTTHT  0
           HTTHT 0
             HTTHT 1
```

whilst $YX$ can be shown to equal 00010

## Representation of the correlation

Other representations of either string:

1. as a number in some base $t$. Thus, e.g. $XY_2 = 9$
2. as a polynomial. Thus, e.g. $XY_t = t^3 + 1$

## Autocorrelation

Furthermore, *autocorrelation* of $X$ can be defined as $XX$. It represents the periods of $X$, i.e. those shifts of $X$ that cause that pattern to overlap itself. Using $Y = MPPMP$ from our previous example, $YY$ evaluates to 10010 Using $A = MMM$, $AA$ evaluates to 111

## Autocorrelation set

Given a string $H$, the autocorrelation *set* $A_{HH}$ or just $A$ is defined as

$$A_{HH} = \{H_{k+1}^{m} : H_1^k = H_{m-k+1}^{m}\}$$

## Example of an autocorrelation set

Let $H = SOS$ The autocorrelation reveals to be

$$HH = 101$$

whereas the autocorrelation set in that case is

$$A = \{\epsilon, 01\}$$

### Let's play a game

The Penny game - invented by Penney.
Each player chooses a pattern.
They then flip a coin until the pattern comes up consecutively. The player who chooses only one symbol ($k$ times), has a chance to win of at least 0.5 This is because of the "optimal" autocorrelation.

## 8.2 Sources

### Bernoulli

A *Bernoulli Source*, or *memoryless source*, generates text randomly.
Every subsequent symbol (of a finite alphabet) is created independently of its predecessors, and the probability of each symbol is not necesserily the same.
If it is, the Source is called a *symmetric*, or *unbiased* Bernoulli Source.
If text over an alphabet $S$ is generated by a Bernoulli Source, then each symbol $s \in S$ *always* occurs with probability $P(s)$.

### Markovian Source

A *Markovian Source* generates symbols based not on the *a priori* probability of each symbol.
Instead, it only heeds a (finite) set of predecessors to ascertain the probability of each next symbol.
In order to do so, it requires a *memory* of previously emitted symbols.
Text generated by a Markovian Source is a realization of a Markov sequence of order $K$.
$K$ denotes the number of previous symbols that the probability of the next symbol depends on.
In our application, this sequence will be stationary and $K = 1$, i.e. a first-order Markov sequence.
When computing the next symbol, we only need to observe the last symbol.
In our case ($K = 1$), the transition matrix is defined by

$$P = \{p_{i,j}\}_{i,j \in S}$$

where

$$p_{i,j} = \text{Probability } (t_{k+1} = j | t_k = i)$$

The matrix entry $(i, j)$ denotes the conditional probability of the next symbol being $j$ if the current symbol is $i$.

## 8.3 Generating functions of languages

### What is a language, after all

A language $L$ is a collection of words.
This collection must satisfy certain properties to belong to a specific language.
Thus, we can associate with a language $L$ its generating function $L(z)$.

## Generating functions

Given a sequence $\{a_n\}_{n\geq 0}$, we know its generating function is defined as

$$A(z) = \sum_{n\geq 0} a_n z^n$$

For sinister purposes, we represent it differently as

$$A(z) = \sum_{\alpha\in S} z^{w(\alpha)}$$

where $S$ is a set of objects (words ...) and $w(\alpha)$ is a weight function.
Henceforth we will interpret it as the size of $\alpha$, i.e. $w(\alpha) = |\alpha|$
The equivalence becomes evident when we set $a_n$ to be the number of objects $\alpha$ satisfying $w(\alpha) = n$. Now we have a more combinatorial view

## Generating function of a language

Now, for any language $L$, we define its generating function $L(z)$ as

$$L(z) = \sum_{w\in L} P(w) z^{|w|}$$

where $P(w)$ is the probability of word $w$'s occurence and $|w|$ is the length of $w$.
So the coefficient of $z^{|w|}$ is the sum of the probabilites all words of that length.
In addition, we assume that $P(\epsilon) = 1$. So every language includes the empty word (as we know).

## Conditional generating function

In addition, the $H$-conditional generating function of $L$ is given as

$$\begin{aligned} L_H(z) &= \sum_{w\in L} P(w|w_{-m} = h_1 \dots w_{-1} = h_m) z^{|w|} \\ &= \sum_{w\in L} P(w|w_{-m}^{-1} = H) z^{|w|} \end{aligned}$$

where $w_{-i}$ is the symbol preceding the first character of $w$ at distance $i$.
We use this definition for Markovian sources, where the probability depends on the previous symbols.

## Example: autocorrelation generating function

In our previous example, the autocorrelation set was

$$A = \{\epsilon, 01\}$$

The generating function of the set is

$$A(z) = 1 + \frac{z^2}{4}$$

given a Bernoulli source, and

$$A_{SOS}(z) = 1 + p_{SO} p_{OS} z^2$$

given a Markovian source of order one.

### Formulating our objective

We will now formulate the special generating functions whose closed form we will later strive to compute:

1. $T^{(r)}(z) = \sum_{n \geq 0} Pr(O_n(H) = r) z^n$
2. $T(z,u) = \sum_{r=1}^{\infty} T^{(r)}(z) u^r = \sum_{r=1}^{\infty} \sum_{n=0}^{\infty} Pr(O_n(H) = r) z^n u^r$

## 8.4 Declaring languages

### Introduction

Let $H$ be a given pattern.

- The *initial language* $R$ is the set of words containing only **one** occurrence of $H$, located at the **right** end.
- The *tail language* $U$ is defined as the set of words $u$ such that $Hu$ has exactly **one** occurrence of $H$, which occurs at the **left** end.
- The *minimal language* $M$ is the set of words $w$ such that $Hw$ has exactly **two** occurrences of $H$, located at its **left** and **right** ends.

### Component languages

We differentiate several special languages, given a pattern $H$. "$\cdot$" stands for concatenation of words.

1. $R = \{r : r \in T_1 \wedge H \text{ occurs at the right end of} r\}$
2. $U = \{u : H \cdot u \in T_1\}$
3. $M = \{w : H \cdot w \in T_2 \wedge H \text{ occurs at the right end of } H \cdot w\}$

## 8.5 Language relationships

### Qualities of $T_r$

At first, we will try to describe the languages $T$ and $T_r$ in terms of $R$, $M$ and $U$:
$\forall r \geq 1$ :

$$T_r = R \cdot M^{r-1} \cdot U$$

### Composition proof ($T_r$)

Proof:
First occurance of $H$ in a $T_r$ word determines the prefix $p$
which is in $R$.
From that prefix on, we look onward until the next occurance of $H$.
The found word $w$ is $\in M$.
After $r-1$ iterations, we add a $H$-devoid suffix, which is in $U$, because its prefix has $H$ at the end.

□

## Qualities of $T$

The "extended" version of $T_r$, its words including an arbitrary number of $H$ occurrences, can be composed similarily:

$$T = R \cdot M^* \cdot U$$

where $M^* := \bigcup_{r=0}^{\infty} M^r$

## Composition proof ($T$)

Proof:
A word belongs to $T$, if for some $1 \leq r < \infty$ it belongs to $T_r$.
As $\bigcup_{r=1}^{\infty} M^{r-1} = \bigcup_{r=0}^{\infty} M^r = M^*$, the assertion is proven.

□

# Four language relationships

Analyzing the relationships between M, U and R further, we introduce

1. $W$, the set of all words
2. $S$, the alphabet set
3. the operators "+" and "-", which denote disjoint union and language subtraction

## Four language relationships I

$$\bigcup_{k \geq 1} M^k = W \cdot H + (A - \{e\})$$

Proof:
$\leftarrow$:
Let $k$ be the number how often $H$ occurs in $W \cdot H$.
$k \geq 1$.
The *last* occurrence of $H$ in every included word is on the right.
That means, that $W \cdot H \subseteq \bigcup_{k \geq 1} M^k$.
$\rightarrow$:
Let $w \in \bigcup_{k \geq 1} M^k$.
Iff $|w| \geq |H|$, then surely the inclusion is correct.
Iff $|w| < |H|$ (how can that be?), then $w \notin W \cdot H$.
But then, necessarily, $w \in A - \{\epsilon\}$, because the second $H$ in $Hw$ overlaps with the first $H$ by definition (it is element of $M^k$), so $w$ must be in the autocorrelation set $A$.

□

## Four language relationships II

$$U \cdot S = M + U - \{e\}$$

Proof:
All words of $S$ consist of a single character $s$.
Given a word $u \in U$ and concatenating them, we differentiate two cases.
If $Hus$ contains a second occurrence of $H$, it is clearly at the right end. Then $us \in M$.
If $Hus$ does contain only a single $H$, then $us$ must be non-empty word of $U$.

□

### Four language relationships III

$$H \cdot M = S \cdot R - (R - H)$$

Proof: $\rightarrow$: Let $sw$ be a word in $H \cdot M$, $s \in S$ (we can write every such word in this way WLOG).
$sw$ contains exactly two times $H$, evidently at its left, and also at its right end. Thus, $sw$ is also $\in S \cdot R$
$\leftarrow$: If a word $swH$ from $S \cdot R$ is not in $R$, then because it contains a second $H$ starting at the left end of $sw$, because $wH \in R$. Of course, in that case it is $\in H \cdot M$.

$\square$

### Four language relationships IV

$$T_0 \cdot H = R \cdot A$$

Proof:
Let $wH$ be $\in T_0 \cdot H$. Then there can be either be one or more occurences of $H$ in $wH$, one of which is at the right end.
If there is no second one, then $wH$ is $\in R$ by definition of $R$
If, however, there is a second one, then it overlaps somehow with the first one.
So we view the word until the end of the *first* $H$, which is in $R$. Due to the overlapping, the remaining part is $\in A$.

$\square$

### One more

Combining relationships II and III yields

$$H \cdot U \cdot S - H \cdot U = (S - \epsilon)R$$

No proof is necessary, as we have validated both ingredients.
Using II, the left side is $H(U \cdot S - U) = H \cdot M$
The right side is

$$S \cdot R - R = S \cdot R - (R \cap S \cdot R) \qquad = S \cdot R - (R - H)$$

Together, that is just relationship III.

## 8.6 Languages & Generating Functions

### in the bernoulli environment

We will now transcend from languages to their generating functions. Given any language $L_1$, we know its generating function to be

$$A_1(z) = \sum_{w \in L_1} P(w) z^{|w|}$$

So what is the the result of multiplying two languages (i.e. concatenating them) in respect to their gen. func.? What is $L_3 = L_1 \cdot L_2$?

$$\begin{aligned}
A_3(z) &= \sum_{w \in L_3} P(w) z^{|w|} \\
&= \sum_{w \in L_1 \wedge w \in L_2} P(w_1) P(w_2) z^{|w_1|+|w_2|} \\
&= \sum_{w \in L_1} P(w_1) z^{|w_1|} \sum_{w \in L_2} P(w_2) z^{|w_2|} \\
&= A_1(z) A_2(z)
\end{aligned}$$

*!* The assumption $P(wv) = P(w)P(v)$ only holds true with a memoryless source.

### Special Cases

A few particular cases:

- $S$ (alphabet set) $\Rightarrow S(z) = \sum_{s \in S} P(s) z^{|s|} = z$
- $L = S \cdot L_1 \Rightarrow L(z) = z L_1(z)$
- $\{\epsilon\} \Rightarrow E(z) = \sum_{w \in \{\epsilon\}} P(w) z^{|w|} = 1 \cdot 1 = 1$
- $H \Rightarrow H(z) = \sum_{w=H} P(H) z^{|H|} = P(H) z^m$
- $W$ (the set of *all* words) $\Rightarrow W(z) = \sum P(w) z^{|k|} = \sum_{k \geq 0} z^k = \frac{1}{1-z}$

## 8.7 Looking for Generating Functions

### Translating I

We will now attempt to translate our known language relationships into generating functions: In case I only, the formula we derive is correct just for a memoryless source.

$$\begin{aligned}
\bigcup_{k \geq 1} M^k &= W \cdot H + (A - \{e\}) \\
\sum_{k=1}^{\infty} M_H(z)^k &= W(z) \cdot P(H) z^m + A_H(z) - 1 \\
\sum_{k=0}^{\infty} M_H(z)^k - 1 &= \frac{1}{1-z} \cdot P(H) z^m + A_H(z) - 1 \\
\frac{1}{1 - M_H(z)} &= \frac{1}{1-z} \cdot P(H) z^m + A_H(z)
\end{aligned}$$

### Translating II

$$\begin{aligned}
U \cdot S &= M + U - \{e\} \\
U \cdot S - U &= M - \{e\} \\
U_H(z) z - U_H(z) &= M_H(z) - 1 \\
U_H(z)(z-1) &= M_H(z) - 1 \\
U_H(z) &= \frac{M_H(z) - 1}{(z-1)}
\end{aligned}$$

### Translating III

$$
\begin{aligned}
H \cdot M &= S \cdot R - (R - H)H \cdot M - H \qquad = S \cdot R - R \\
P(H)z^m M_H(z) - P(H)z^m &= S(z) \cdot R(z) - R(z) \\
P(H)z^m (M_H(z) - 1) &= R(z)(z - 1) \\
R(z) &= P(H)z^m \frac{M_H(z) - 1}{z - 1} \\
R(z) &= P(H)z^m U_H(z)
\end{aligned}
$$

## 8.8 Main findings I

### $T^{(r)}(z)$

We remember, that for $r \geq 1$

$$T_r = R \cdot M^{r-1} \cdot U$$

We have now gleaned every component, and can translate it (for $r \geq 1$) into

$$T^{(r)}(z) = R(z)M^{r-1}(z)U_H(z)$$

### $T(z, u)$

We do also remember, that

$$T = R \cdot M^* \cdot U$$

As $T$ is the language with *any* number of $H$s, its generating function is indeed ...

$$T(z, u) = R(z)\frac{u}{1 - uM_H(z)}U_H(z)$$

## 8.9 On to other shores

### What is left to do?

We still have no formula of gathering $O_n(H)$, i.e. the frequency of $H$-occurrences ($|H| = m$) in random text of length $n$ over an alphabet $S$ with $|S| = V$.
Let us make an educated guess, though. What we do not know, is how important *overlapping* is. Assuming to disregard that topic, the answer *could* be

$$E[O_n(H)] = P(H)(n - m + 1)$$

It is.
But why?

### Using derivatives

Looking at our bivariate generating function of $T$,

$$T(z, u) = \sum_{r=1}^{\infty} \sum_{n=0}^{\infty} Pr(O_n(H) = r)z^n u^r$$

we notice that we would like the two sums to be reversed. Deriving it after $u$ ...

$$T_u(z,u) = \sum_{r=1}^{\infty}\sum_{n=0}^{\infty} Pr(O_n(H)=r)z^n r \text{ (=\#Occ) } u^{r-1}$$

... and setting $u$ to 1 leads to ...

$$T_u(z,1) = \sum_{n=0}^{\infty}(\sum_{r=1}^{\infty} Pr(O_n(H)r)z^n$$

## Proof Preparations

To shorten things, we introduce

$$D_H(z) = (1-z)A_H(z) + z^m P(H)$$

and rewrite $M_H(z)$ as

$$M_H(z) = 1 + \frac{z-1}{D_H(z)}$$

as well as

$$U_H(z) = \frac{1}{D_H(z)}$$

and

$$R(z) = z^m P(H)\frac{1}{D_H(z)}$$

## Deriving the closed form formula (1)

$$\begin{aligned}
T_u(z,u) &= R(z)U_H(z)\frac{u}{(1-uM_H)}\frac{d}{du} \\
&= R(z)U_H(z)\frac{(1-uM)+uM}{(1-uM_H)^2} \\
&= R(z)U_H(z)\frac{1}{(1-uM_H)^2}
\end{aligned}$$

## Deriving the closed form formula (2)

$u$ is now set to 1 due to the previous calculus:

$$\begin{aligned}
T_u(z,1) &= R(z)U_H(z)\frac{1}{(1-M_H)^2} \\
&= R(z)U_H(z)(1-1+\frac{z-1}{D_H(z)})^{-2} \\
&= R(z)U_H(z)\frac{D_H(z)^2}{(z-1)^2} \\
&= R(z)\frac{1}{D_H(z)}\frac{D_H(z)^2}{(z-1)^2} \\
&= z^m P(H)\frac{1}{D_H(z)}\frac{D_H(z)}{(z-1)^2} \\
&= \frac{z^m P(H)}{(z-1)^2}
\end{aligned}$$

## Main findings II

As the text has length $n$, we are extracting the $n$th coefficient of $T_u(z,1)$, and *voilà*

$$\begin{aligned} E[O_n] &= [z^n]T_u(z,1) \\ &= P(H)[z^n]z^m(1-z)^{-2} \\ &= P(H)[z^{n-m}](1-z)^{-2} \\ &= (n-m+1)P(H) \end{aligned}$$

## About certainty

the variance of $E(O_n(H)$ is, for a $r > 1$:

$$Var[O_n(H)] = nc_1 + c_2 + O(r^{-n})$$

where

$$c_1 = P(H)(2A_H(1) - 1 - (2m-1)P(H) + 2P(H)E_1))$$

$$\begin{aligned} c_2 \quad &= P(H)((m-1)(3m-1)P(H) - (m-1) \\ &(2A_H(1) - 1) - 2A'_H(1)) - 2(2m-1) \\ &(P(H)^2E_1 + 2E_2P(H)^2 \end{aligned}$$

$E_1$, $E_2$ are

$$E_1 = \frac{1}{\pi_{h1}}[(P-\Pi)Z]_{h_m,h_1} E_2 \qquad = \frac{1}{\pi_{h1}}[(P^2-\Pi)Z^2]_{h_m,h_1}$$

Without going into detail (cf. literature references), we see that the Variance depens mainly on the length of the text plus a constant.