VARIOUS POINTS OF VIEW ON SOURCES

VARIOUS POINTS OF VIEW ON SOURCES
IN ANALYTIC INFORMATION THEORY

VARIOUS POINTS OF VIEW ON SOURCES
IN ANALYTIC INFORMATION THEORY

APPLICATIONS to PROBABILISTIC ANALYSES
of DICTIONARY STRUCTURES

Brigitte Vallée

GREYC (CNRS and University of Caen)

VARIOUS POINTS OF VIEW ON SOURCES IN ANALYTIC INFORMATION THEORY

APPLICATIONS to PROBABILISTIC ANALYSES of DICTIONARY STRUCTURES

Brigitte Vallée

GREYC (CNRS and University of Caen)

Results obtained in joint works with

Eda Cesaratto, Julien Clément, Philippe Flajolet,
Kanal Hun, Mathieu Roux

VARIOUS POINTS OF VIEW ON SOURCES
IN ANALYTIC INFORMATION THEORY

APPLICATIONS to PROBABILISTIC ANALYSES
of DICTIONARY STRUCTURES

Brigitte Vallée

GREYC (CNRS and University of Caen)

Results obtained in joint works with

Eda Cesaratto, Julien Clément, Philippe Flajolet,
Kanal Hun, Mathieu Roux

Journées SDA2, Avril 2015

Two main objects : sources and data structures

- Describe a modelling of natural sources
- Deduce consequences for the analysis of related data structures

Two main objects : sources and data structures

- Describe a modelling of natural sources
- Deduce consequences for the analysis of related data structures

Plan of the talk.

Two main objects : sources and data structures

– Describe a modelling of natural sources

– Deduce consequences for the analysis of related data structures

Plan of the talk.

– A general model of sources

– The two digital structures : trie and dst.

– Probabilistic analysis of the structures, and its two steps

– Probabilistic analysis : the combinatorial step

– Probabilistic analysis : the analytic step – Need of more regular sources.

– Analysis of data structures : the result.

Sources (I)

In information theory, a source:=

a probabilistic mechanism which produces symbols from alphabet Σ,

one at each time unit.

When (discrete) time evolves, a source produces (infinite) words

In information theory, a source:=

a probabilistic mechanism which produces symbols from alphabet Σ,

one at each time unit.

When (discrete) time evolves, a source produces (infinite) words

X_n the symbol emitted at time $t = n$.

A probabilistic source is defined by the sequence (X_n) of random variables

In information theory, a source:=

a probabilistic mechanism which produces symbols from alphabet Σ,

one at each time unit.

When (discrete) time evolves, a source produces (infinite) words

$$X_n \text{ the symbol emitted at time } t = n.$$

A probabilistic source is defined by the sequence (X_n) of random variables

The sequence may be bi-infinite (in $\Sigma^{\mathbb{Z}}$) ... or only right–infinite in $\Sigma^{\mathbb{N}}$.

In information theory, a source:=

a probabilistic mechanism which produces symbols from alphabet Σ,

one at each time unit.

When (discrete) time evolves, a source produces (infinite) words

X_n the symbol emitted at time $t = n$.

A probabilistic source is defined by the sequence (X_n) of random variables

The sequence may be bi-infinite (in $\Sigma^{\mathbb{Z}}$) ... or only right–infinite in $\Sigma^{\mathbb{N}}$.

– bi–infinite (in $\Sigma^{\mathbb{Z}}$) or $\mathbb{Z}$–history : the indices $n \in \mathbb{Z}$

easier for probabilistic studies

In information theory, a source:=

a probabilistic mechanism which produces symbols from alphabet Σ,

one at each time unit.

When (discrete) time evolves, a source produces (infinite) words

X_n the symbol emitted at time $t = n$.

A probabilistic source is defined by the sequence (X_n) of random variables

The sequence may be bi-infinite (in $\Sigma^{\mathbb{Z}}$) ... or only right–infinite in $\Sigma^{\mathbb{N}}$.

- bi–infinite (in $\Sigma^{\mathbb{Z}}$) or $\mathbb{Z}$–history : the indices $n \in \mathbb{Z}$

  easier for probabilistic studies

- right–infinite (in $\Sigma^{\mathbb{N}}$) or $\mathbb{N}$–history : the indices $n \in \mathbb{N}$

  natural for algorithmic applications in text algorithms

In information theory, a source:$=$

a probabilistic mechanism which produces symbols from alphabet Σ,

one at each time unit.

When (discrete) time evolves, a source produces (infinite) words

X_n the symbol emitted at time $t = n$.

A probabilistic source is defined by the sequence (X_n) of random variables

The sequence may be bi-infinite (in $\Sigma^{\mathbb{Z}}$) ... or only right–infinite in $\Sigma^{\mathbb{N}}$.

– bi–infinite (in $\Sigma^{\mathbb{Z}}$) or $\mathbb{Z}$–history : the indices $n \in \mathbb{Z}$

easier for probabilistic studies

– right–infinite (in $\Sigma^{\mathbb{N}}$) or $\mathbb{N}$–history : the indices $n \in \mathbb{N}$

natural for algorithmic applications in text algorithms

Compromise: Only the positive part of the history is "shown"....

The negative part of the history

– is produced

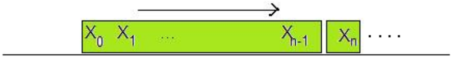
– may have an influence on the positive part

– but remains "hidden"

A source on the alphabet Σ with a positive history :

An origin for time : $t = 0$

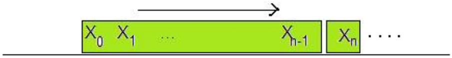
a sequence of random variables $(X_0, X_1, \ldots, X_n, X_{n+1} \ldots)$



A source on the alphabet Σ with a positive history :

An origin for time : $t = 0$

a sequence of random variables $(X_0, X_1, \ldots, X_n, X_{n+1} \ldots)$

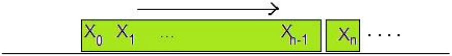


Simple sources: sources with weak correlations between successive symbols

A source on the alphabet Σ with a positive history :

An origin for time : $t = 0$

a sequence of random variables $(X_0, X_1, \ldots, X_n, X_{n+1} \ldots)$



Simple sources: sources with weak correlations between successive symbols

Memoryless source :

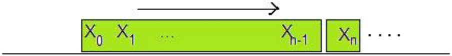
The variables X_i are independent,

with the same distribution defined by $p_i := \Pr[X_n = i]$ $(i \in \Sigma)$

A source on the alphabet Σ with a positive history :

An origin for time : $t = 0$

a sequence of random variables $(X_0, X_1, \ldots, X_n, X_{n+1} \ldots)$



Simple sources: sources with weak correlations between successive symbols

Memoryless source :

The variables X_i are independent,

with the same distribution defined by $p_i := \Pr[X_n = i]$ $(i \in \Sigma)$

Markov chain:

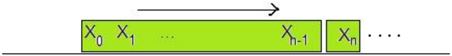
The only dependence is between consecutive X_n's, does not depend on n

defined by the transition matrix $p_{i|j} := \Pr[X_{n+1} = i \,|X_n = j]$

A source on the alphabet Σ with a positive history :

An origin for time : $t = 0$

a sequence of random variables $(X_0, X_1, \ldots, X_n, X_{n+1} \ldots)$



Simple sources: sources with weak correlations between successive symbols

Memoryless source :

The variables X_i are independent,

with the same distribution defined by $p_i := \Pr[X_n = i]$ $(i \in \Sigma)$

Markov chain:

The only dependence is between consecutive X_n's, does not depend on n

defined by the transition matrix $p_{i|j} := \Pr[X_{n+1} = i \,|X_n = j]$

A general source may have many, strong correlations between its symbols.

For $w \in \Sigma^\star$, $p_w :=$ probability that a word begins with the prefix w.

The set $\{p_w, \quad w \in \Sigma^\star\}$ defines the source $\mathcal{S}$.

A main analytical object related to any source:
the Dirichlet generating functions of the source

$$\Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s, \qquad \Lambda^{[k]}(s) = \sum_{w \in \Sigma^k} p_w^s, \qquad \left[\Lambda = \sum_{k \geq 0} \Lambda^{[k]} \right]$$

Remark: $\Lambda^{[k]}(1) = 1$ for any k, $\qquad \Lambda(1) = \infty$.

A main analytical object related to any source:

the Dirichlet generating functions of the source

$$\Lambda(s) := \sum_{w\in\Sigma^\star} p_w^s, \qquad \Lambda^{[k]}(s) = \sum_{w\in\Sigma^k} p_w^s, \qquad \left[\Lambda = \sum_{k\geq 0} \Lambda^{[k]}\right]$$

Remark: $\Lambda^{[k]}(1) = 1$ for any k, $\qquad \Lambda(1) = \infty$.

– they encapsulate the main probabilistic properties of the source

– they translate them into analytic properties

A main analytical object related to any source:
the Dirichlet generating functions of the source

$$\Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s, \qquad \Lambda^{[k]}(s) = \sum_{w \in \Sigma^k} p_w^s, \qquad \left[\Lambda = \sum_{k \geq 0} \Lambda^{[k]} \right]$$

Remark: $\Lambda^{[k]}(1) = 1$ for any k, $\qquad \Lambda(1) = \infty$.

– they encapsulate the main probabilistic properties of the source

– they translate them into analytic properties

For instance, the entropy $h_{\mathcal{S}}$, (if it exists) is

$$h_{\mathcal{S}} := \lim_{k \to \infty} \frac{-1}{k} \sum_{w \in \Sigma^k} p_w \log p_w = \lim_{k \to \infty} \left[-\frac{1}{k} \frac{d}{ds} \Lambda^{[k]}(s) \Big|_{s=1} \right]$$

A main analytical object related to any source:
the Dirichlet generating functions of the source

$$\Lambda(s) := \sum_{w\in\Sigma^\star} p_w^s, \qquad \Lambda^{[k]}(s) = \sum_{w\in\Sigma^k} p_w^s, \qquad \left[\Lambda = \sum_{k\geq 0} \Lambda^{[k]}\right]$$

Remark: $\Lambda^{[k]}(1) = 1$ for any k, $\quad \Lambda(1) = \infty$.

– they encapsulate the main probabilistic properties of the source

– they translate them into analytic properties

For instance, the entropy $h_{\mathcal{S}}$, (if it exists) is

$$h_{\mathcal{S}} := \lim_{k\to\infty} \frac{-1}{k} \sum_{w\in\Sigma^k} p_w \log p_w = \lim_{k\to\infty} \left[-\frac{1}{k}\, \frac{d}{ds} \Lambda^{[k]}(s)\Big|_{s=1}\right]$$

– they intervene in probabilistic analyses of algorithms and data structures.

A main analytical object related to any source:

the Dirichlet series of probabilities, $\quad \Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s$

A main analytical object related to any source:

the Dirichlet series of probabilities, $\Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s$

Memoryless sources, with probabilities (p_i)

$$\Lambda(s) = \frac{1}{1 - \lambda(s)} \qquad \text{with} \quad \lambda(s) = \sum_{i=1}^{r} p_i^s$$

A main analytical object related to any source:

the Dirichlet series of probabilities, $\Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s$

Memoryless sources, with probabilities (p_i)

$$\Lambda(s) = \frac{1}{1 - \lambda(s)} \qquad \text{with} \quad \lambda(s) = \sum_{i=1}^{r} p_i^s$$

Markov chains, defined by – the vector $\mathbf{R}$ of initial probabilities (r_i)

– and the transition matrix $\mathbf{P} := (p_{j|i})$

$$\Lambda(s) = 1 + {}^t\mathbf{R}_s (I - \mathbf{P}_s)^{-1} \mathbf{1} \qquad \text{with} \quad \mathbf{P}_s = (p_{j|i}^s), \quad \mathbf{R}_s = (r_i^s).$$

A main analytical object related to any source:

the Dirichlet series of probabilities, $\Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s$

Memoryless sources, with probabilities (p_i)

$$\Lambda(s) = \frac{1}{1-\lambda(s)} \qquad \text{with} \quad \lambda(s) = \sum_{i=1}^{r} p_i^s$$

Markov chains, defined by – the vector $\mathbf{R}$ of initial probabilities (r_i)

– and the transition matrix $\mathbf{P} := (p_{j|i})$

$$\Lambda(s) = 1 + {}^t\mathbf{R}_s(I - \mathbf{P}_s)^{-1}\mathbf{1} \qquad \text{with} \quad \mathbf{P}_s = (p_{j|i}^s), \quad \mathbf{R}_s = (r_i^s).$$

These nice expressions are due to multiplicative properties of probabilities.

And for a general source?

Does $\Lambda(s)$ admit a nice alternative expression?

A general source and its shifted sources

A general source $\mathcal{S}$ is completely defined by its fundamental probabilities

$p_w :=$ the probability that a word of $\mathcal{S}$ begins with the prefix $w \in \Sigma^\star$

A general source and its shifted sources

A general source $\mathcal{S}$ is completely defined by its fundamental probabilities

$p_w :=$ the probability that a word of $\mathcal{S}$ begins with the prefix $w \in \Sigma^\star$

The source $\mathcal{S}$ defines a sequence of sources $\mathcal{S}_{(u)}$ (for $u \in \Sigma^\star$)

For $u \in \Sigma^\star$ with $p_u \neq 0$, the source $\mathcal{S}_{(u)} = \mathcal{S}|_u$ is a shifted source

- which gathers all the words of $\mathcal{S}$ which begin with $u \in \Sigma^\star$,
- from which the prefix u is removed.

A general source and its shifted sources

A general source $\mathcal{S}$ is completely defined by its fundamental probabilities

$p_w :=$ the probability that a word of $\mathcal{S}$ begins with the prefix $w \in \Sigma^\star$

The source $\mathcal{S}$ defines a sequence of sources $\mathcal{S}_{(u)}$ (for $u \in \Sigma^\star$)

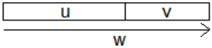
For $u \in \Sigma^\star$ with $p_u \neq 0$, the source $\mathcal{S}_{(u)} = \mathcal{S}|_u$ is a shifted source

- which gathers all the words of $\mathcal{S}$ which begin with $u \in \Sigma^\star$,
- from which the prefix u is removed.

The source $\mathcal{S}_{(u)}$ is completely defined

- by the fundamental (conditional) probabilities p_w/p_u,
- when w is any finite prefix for which $u \leq w$.

In this case, w can be written as $w = u \cdot v$



A general source and its shifted sources

A general source $\mathcal{S}$ is completely defined by its fundamental probabilities

$p_w :=$ the probability that a word of $\mathcal{S}$ begins with the prefix $w \in \Sigma^\star$

The source $\mathcal{S}$ defines a sequence of sources $\mathcal{S}_{(u)}$ (for $u \in \Sigma^\star$)

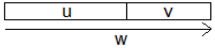
For $u \in \Sigma^\star$ with $p_u \neq 0$, the source $\mathcal{S}_{(u)} = \mathcal{S}|_u$ is a shifted source

- which gathers all the words of $\mathcal{S}$ which begin with $u \in \Sigma^\star$,
- from which the prefix u is removed.

The source $\mathcal{S}_{(u)}$ is completely defined

- by the fundamental (conditional) probabilities p_w/p_u,
- when w is any finite prefix for which $u \leq w$.

In this case, w can be written as $w = u \cdot v$

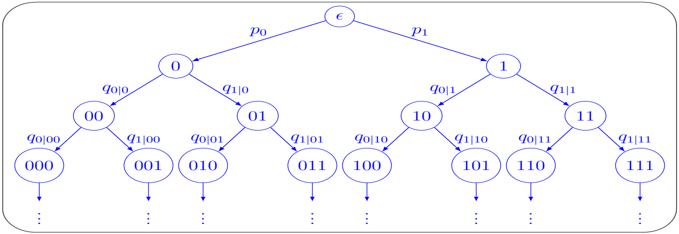


The conditional probabilities $p_{w|u} = p_{(u.v)}/p_u$ are denoted as $q_{v|u}$.

These are the fundamental probabilities of the source $S_{(u)}$.

The weighted graph of the source

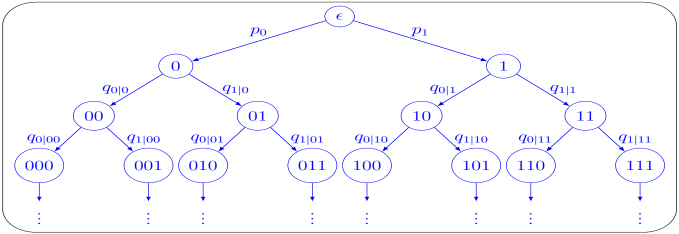
- vertices = sources $\mathcal{S}_{(u)}$
- edges weighted by the probabilities $q_{v|u}$



The generalized transition matrix of a source $\mathcal{S}$

The weighted graph of the source

- vertices = sources $\mathcal{S}_{(u)}$
- edges weighted by the probabilities $q_{v|u}$



$\mathbf{P}$ = the transition matrix of the graph.

= an infinite matrix, whose rows and columns are indexed by $\Sigma^\star$

The non zero elements at the row w are located at the columns $w \cdot i$.

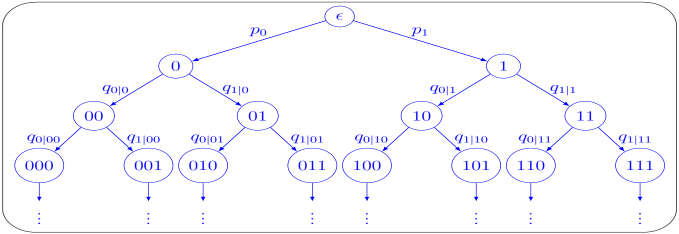
The generalized transition matrix $\mathbf{P}$ of the source $\mathcal{S}$

extends the transition matrix of a Markov chain.

The generalized transition matrix of a source $\mathcal{S}$

The weighted graph of the source

- vertices = sources $\mathcal{S}_{(u)}$
- edges weighted by the probabilities $q_{v|u}$



$\mathbf{P}$ = the transition matrix of the graph.

= an infinite matrix, whose rows and columns are indexed by $\Sigma^\star$

The non zero elements at the row w are located at the columns $w \cdot i$.

The generalized transition matrix $\mathbf{P}$ of the source $\mathcal{S}$

extends the transition matrix of a Markov chain.

For $s \in \mathbb{C}$, the matrix $\mathbf{P}_s$ is obtained from $\mathbf{P}$ by raising its elements to the power s

The pruned graph and the pruned matrix (I)

Sometimes, the graph (and thus the matrix) can be pruned:

With an equivalence relation on the "shifted" sources

$$\mathcal{S}_{(u)} \equiv \mathcal{S}_{(v)} \qquad \Longleftrightarrow \qquad \forall w \in \Sigma^{\star}, \quad q_{w|u} = q_{w|v},$$

one only keeps the sources $S_{(u)}$ which have a different distribution

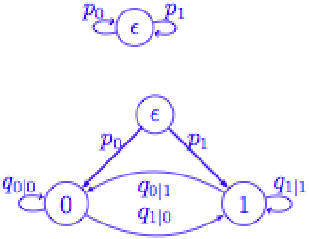
The pruned graph and the pruned matrix (I)

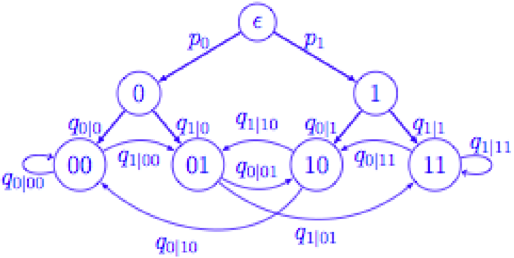
Sometimes, the graph (and thus the matrix) can be pruned:
With an equivalence relation on the “shifted” sources

$$\mathcal{S}_{(u)} \equiv \mathcal{S}_{(v)} \qquad \Longleftrightarrow \qquad \forall w \in \Sigma^\star, \quad q_{w|u} = q_{w|v},$$

one only keeps the sources $S_{(u)}$ which have a different distribution

For simple sources, this provides a finite graph (a finite matrix).





The pruned graph and the pruned matrix (II)

There are pruned graphs which remain infinite.

An instance of a VLMC (Variable Length Markov Chain)

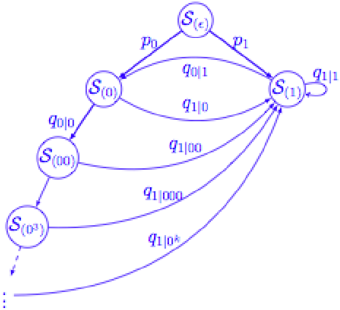
The distribution of X_n depends
on the length of the run 0^k which precedes it

The pruned graph and the pruned matrix (II)

There are pruned graphs which remain infinite.

An instance of a VLMC (Variable Length Markov Chain)

The distribution of X_n depends
on the length of the run 0^k which precedes it



Pruned graph :

– vertices $\mathcal{S}_{(\epsilon)}, \mathcal{S}_{(1)}$ and $\mathcal{S}_{(0^k)}$ for $k > 0$
– all the edges labeled with 1
return to the source $\mathcal{S}_{(1)}$.

Return to the Dirichlet generating function of the source, $\Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s$

Return to the Dirichlet generating function of the source, $\Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s$

Memoryless sources, with probabilities (p_i)

$$\Lambda(s) = \frac{1}{1 - \lambda(s)} \qquad \text{with} \quad \lambda(s) = \sum_{i=1}^{r} p_i^s$$

Return to the Dirichlet generating function of the source, $\Lambda(s) := \sum_{w\in\Sigma^\star} p_w^s$

Memoryless sources, with probabilities (p_i)

$$\Lambda(s) = \frac{1}{1-\lambda(s)} \qquad \text{with} \quad \lambda(s) = \sum_{i=1}^{r} p_i^s$$

Markov chains, defined by – the vector $\mathbf{R}$ of initial probabilities (r_i)

– and the transition matrix $\mathbf{P} := (p_{j|i})$

$$\Lambda(s) = 1 + {}^t\mathbf{R}_s(I-\mathbf{P}_s)^{-1}[\mathbf{1}] \qquad \text{with} \quad \mathbf{P}_s = (p_{j|i}^s), \quad \mathbf{R}_s = (r_i^s).$$

Return to the Dirichlet generating function of the source, $\Lambda(s) := \sum_{w \in \Sigma^\star} p_w^s$

Memoryless sources, with probabilities (p_i)

$$\Lambda(s) = \frac{1}{1 - \lambda(s)} \qquad \text{with} \quad \lambda(s) = \sum_{i=1}^{r} p_i^s$$

Markov chains, defined by – the vector $\mathbf{R}$ of initial probabilities (r_i)
– and the transition matrix $\mathbf{P} := (p_{j|i})$

$$\Lambda(s) = 1 + {}^t\mathbf{R}_s (I - \mathbf{P}_s)^{-1}[\mathbf{1}] \qquad \text{with} \quad \mathbf{P}_s = (p_{j|i}^s), \quad \mathbf{R}_s = (r_i^s).$$

A general source, with its (pruned) transition matrix $\mathbf{P}_s$,

$$\Lambda(s) = {}^t\mathbf{E} \cdot (I - \mathbf{P}_s)^{-1}[\mathbf{1}] \quad \text{with} \quad {}^t\mathbf{E} := (1, 0, 0 \ldots)$$

(II) Two data structures: trie and dst

dst : digital search tree — trie: shorthand for tree retrieval

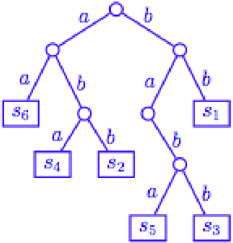
(II) Two data structures: trie and dst

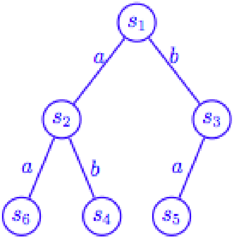
dst : digital search tree — trie: shorthand for tree retrieval

Dynamical data structures which contain words.

– Useful for sorting, and searching words.

– Important to analyze their probabilistic shape
  when built on a sequence of words emitted by a general source





Two types of fundamental digital structures.

trie: introduced by Fredkin, 1959;   dst: by Coffman and Eve, 1970

Two types of fundamental digital structures.

trie: introduced by Fredkin, 1959; dst: by Coffman and Eve, 1970

Both are trees used as dictionaries,

with three main operations (Search, Insert and Delete)

Play a central role in the Lempel-Ziv data compression scheme

Two types of fundamental digital structures.

trie: introduced by Fredkin, 1959; dst: by Coffman and Eve, 1970

Both are trees used as dictionaries,

with three main operations (Search, Insert and Delete)

Play a central role in the Lempel-Ziv data compression scheme

These trees direct words to subtrees according to their first symbol

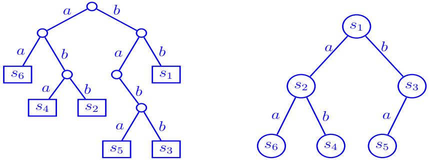
In a trie, – internal nodes do not contain data,

– the order of insertion does not intervene.

In a dst, a word is placed on the first free node.

In a trie, the word is placed when it is alone in its subtree.

$s_1{=}bbab\cdots$; $s_2{=}abbaa\cdots$ $s_3{=}babba\cdots$, $s_4{=}ababb\cdots$; $s_5{=}babab\cdots$; $s_6{=}aaaab\cdots$



Their recursive definitions

Let $\Sigma = \{a, b\}$ and a sequence $\mathcal{Y}$ of words over Σ

$\mathcal{Y}_{(\alpha)}$:= subsequence of $\mathcal{Y}$ beginning with α, the first symbol α removed.

Their recursive definitions

Let $\Sigma = \{a, b\}$ and a sequence $\mathcal{Y}$ of words over Σ

$\mathcal{Y}_{(\alpha)}$:= subsequence of $\mathcal{Y}$ beginning with α, the first symbol α removed.

$\mathrm{trie}(\mathcal{Y})$

- If $|\mathcal{Y}| = 0$, $\mathrm{trie}(\mathcal{Y}) = \varnothing$.
- If $|\mathcal{Y}| = 1$, $\mathrm{trie}(\mathcal{Y}) = \boxed{Y}$
- If $|\mathcal{Y}| \geq 2$,

$$\mathrm{trie}(\mathcal{Y}) = \langle \bullet, \mathrm{trie}(\mathcal{Y}_{(a)}), \mathrm{trie}(\mathcal{Y}_{(b)}) \rangle$$

Their recursive definitions

Let $\Sigma = \{a, b\}$ and a sequence $\mathcal{Y}$ of words over Σ

$\mathcal{Y}_{(\alpha)}$:= subsequence of $\mathcal{Y}$ beginning with α, the first symbol α removed.

$\mathrm{trie}(\mathcal{Y})$

- If $|\mathcal{Y}| = 0$, $\mathrm{trie}(\mathcal{Y}) = \varnothing$.
- If $|\mathcal{Y}| = 1$, $\mathrm{trie}(\mathcal{Y}) = \boxed{Y}$
- If $|\mathcal{Y}| \geq 2$,

$$\mathrm{trie}(\mathcal{Y}) = \langle \bullet, \mathrm{trie}(\mathcal{Y}_{(a)}), \mathrm{trie}(\mathcal{Y}_{(b)}) \rangle$$

$\mathrm{dst}(\mathcal{Y})$

- If $|\mathcal{Y}| = 0$, $\mathrm{dst}(\mathcal{Y}) = \varnothing$
- If $|\mathcal{Y}| \geq 1$, $\underline{\mathcal{Y}} := \mathcal{Y} \setminus \{\mathrm{First}(\mathcal{Y})\}$

$$\mathrm{dst}(\mathcal{Y}) = \langle \mathrm{First}(\mathcal{Y}), \mathrm{dst}(\underline{\mathcal{Y}}_{(a)}), \mathrm{dst}(\underline{\mathcal{Y}}_{(b)}) \rangle$$

Their recursive definitions

Let $\Sigma = \{a, b\}$ and a sequence $\mathcal{Y}$ of words over Σ

$\mathcal{Y}_{(\alpha)}$:= subsequence of $\mathcal{Y}$ beginning with α, the first symbol α removed.

$\mathrm{trie}(\mathcal{Y})$

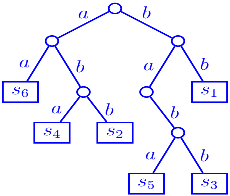
- If $|\mathcal{Y}| = 0$, $\mathrm{trie}(\mathcal{Y}) = \varnothing$.
- If $|\mathcal{Y}| = 1$, $\mathrm{trie}(\mathcal{Y}) = \boxed{Y}$
- If $|\mathcal{Y}| \geq 2$,

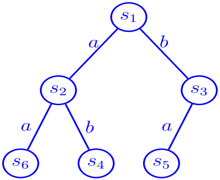
$$\mathrm{trie}(\mathcal{Y}) = \langle \bullet, \mathrm{trie}(\mathcal{Y}_{(a)}), \mathrm{trie}(\mathcal{Y}_{(b)}) \rangle$$

$\mathrm{dst}(\mathcal{Y})$

- If $|\mathcal{Y}| = 0$, $\mathrm{dst}(\mathcal{Y}) = \varnothing$
- If $|\mathcal{Y}| \geq 1$, $\underline{\mathcal{Y}} := \mathcal{Y} \setminus \{\mathrm{First}(\mathcal{Y})\}$

$$\mathrm{dst}(\mathcal{Y}) = \langle \mathrm{First}(\mathcal{Y}), \mathrm{dst}(\underline{\mathcal{Y}}_{(a)}), \mathrm{dst}(\underline{\mathcal{Y}}_{(b)}) \rangle$$





– We will use these recursive definitions to write systems of equations.

Role of the dst in the Lempel–Ziv Algorithm.

The Lempel-Ziv algorithm is a dictionary-based scheme

– it partitions a sequence into phrases of variable size

– a new phrase is the shortest substring not seen in the past as a phrase
obtained by adding a new symbol to a "Déjà Vu" phrase

Role of the dst in the Lempel–Ziv Algorithm.

The Lempel-Ziv algorithm is a dictionary-based scheme
– it partitions a sequence into phrases of variable size
– a new phrase is the shortest substring not seen in the past as a phrase
obtained by adding a new symbol to a "Déjà Vu" phrase

The text 11000101011011101
is partitioned into phrases

Role of the dst in the Lempel–Ziv Algorithm.

The Lempel-Ziv algorithm is a dictionary-based scheme
– it partitions a sequence into phrases of variable size
– a new phrase is the shortest substring not seen in the past as a phrase
obtained by adding a new symbol to a "Déjà Vu" phrase

The text 11000101011011101
is partitioned into phrases

$$(\epsilon) - (1) - (10) - (0) - (01) -$$

$$(010) - (11) - (011) - (101)$$

Role of the dst in the Lempel–Ziv Algorithm.

The Lempel-Ziv algorithm is a dictionary-based scheme

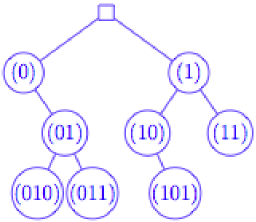
– it partitions a sequence into phrases of variable size

– a new phrase is the shortest substring not seen in the past as a phrase
obtained by adding a new symbol to a "Déjà Vu" phrase

The text 11000101011011101
is partitioned into phrases

$$(\epsilon) - (1) - (10) - (0) - (01) -$$

$$(010) - (11) - (011) - (101)$$



The phrases are inserted in a DST

Parameters for digital trees.

Two types of nodes in the digital structures

– nodes containing data   – or nodes containing no data

A full node is a node containing data

(an internal node for the dst, an external node for the trie)

Parameters for digital trees.

Two types of nodes in the digital structures

– nodes containing data – or nodes containing no data

A full node is a node containing data

(an internal node for the dst, an external node for the trie)

The level of a node: the length of the path from the root to it.

The size is the number of full nodes.

The two main shape parameters:

– Profile $b_{n,k} :=$ the number of full nodes at level k in a tree of size n.

– Depth $D_n :=$ the level of a randomly selected full node.

Parameters for digital trees.

Two types of nodes in the digital structures

– nodes containing data   – or nodes containing no data

A full node is a node containing data

(an internal node for the dst, an external node for the trie)

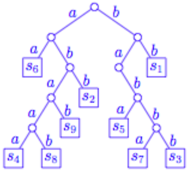
The level of a node: the length of the path from the root to it.

The size is the number of full nodes.

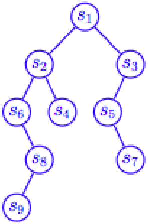
The two main shape parameters:

– Profile $b_{n,k}$:= the number of full nodes at level k in a tree of size n.

– Depth D_n := the level of a randomly selected full node.



$b_{9,0} = 0,$
$b_{9,1} = 0$
$b_{9,2} = 2,$
$b_{9,3} = 1,$
$b_{9,4} = 2$
$b_{9,5} = 4$



$b_{9,0} = 1,$
$b_{9,1} = 2,$
$b_{9,2} = 3,$
$b_{9,3} = 2$
$b_{9,4} = 1$

$$D_n = (1/9)\,[2\cdot 2 + 3\cdot 1 + 4\cdot 2 + 5\cdot 4] = 3.88 \qquad D_n = (1/9)\,[1\cdot 2 + 2\cdot 3 + 3\cdot 2 + 4\cdot 1] = 2$$

(III) Probabilistic analysis of the data structures.

Probabilistic study

Input $=$ a sequence $\mathcal{X}$ of words (independently) produced by the source.

Set of inputs $=$ the set $\mathcal{M}^\star$ of such sequences $\mathcal{X}$

Aim $=$ the probabilistic shape of $\mathrm{Tree}\,(\mathcal{X})$ for $\mathcal{X} \in \mathcal{M}^\star$

Two different probabilistic models : Poisson and Bernoulli

– In the Bernoulli model, the cardinality N of $\mathcal{X}$ is fixed.

– In the Poisson model, the cardinality N follows a Poisson law of parameter z

$$\Pr[N = k] = e^{-z} \frac{z^k}{k!}.$$

The Poisson model is easier to deal with (independence properties).

Thus: begin in the Poisson model and then return to the Bernoulli model...

Probabilistic study

Input $=$ a sequence $\mathcal{X}$ of words (independently) produced by the source.

Set of inputs $=$ the set $\mathcal{M}^\star$ of such sequences $\mathcal{X}$

Aim $=$ the probabilistic shape of $\mathrm{Tree}\ (\mathcal{X})$ for $\mathcal{X} \in \mathcal{M}^\star$

Two different probabilistic models : Poisson and Bernoulli

– In the Bernoulli model, the cardinality N of $\mathcal{X}$ is fixed.

– In the Poisson model, the cardinality N follows a Poisson law of parameter z

$$\Pr[N = k] = e^{-z}\frac{z^k}{k!}.$$

The Poisson model is easier to deal with (independence properties).

Thus: begin in the Poisson model and then return to the Bernoulli model...

For a random variable R defined on the set $\mathcal{M}^\star$ of inputs,

there is a relation between the two expectations

$P(z)$ in the Poisson model and B_n in the Bernoulli model,

$$P(z) = e^{-z}\sum_{n\geq 0} B_n \frac{z^n}{n!}$$

Two steps in the analysis of the profile polynomial $b_N(u) := \sum_{k \geq 0} b_{N,k}\, u^k$,

Deal with the expectations of $b_N(u)$: $B_n(u)$ [Bernoulli] and $P(z,u)$ [Poisson].

(A) The first (combinatorial) step provides an exact expression for $B_n(u)$

Expectation $P(z,u)$ in the Poisson model	$\Longrightarrow$	Mellin transform $s \mapsto Z(s,u)$ of $z \mapsto P(z,u)$	$\Longrightarrow$	Binomial expression of the expectation $B_n(u)$ in the Bernoulli model

$$B_n(u) = \sum_{\ell=2}^{n} (-1)^\ell \binom{n}{\ell}\, \Delta(\ell, u), \qquad \text{with} \quad \Delta(s,u) := \frac{1}{\Gamma(-s)} Z(-s,u)$$

Two steps in the analysis of the profile polynomial $b_N(u) := \sum_{k\geq 0} b_{N,k}\, u^k$,

Deal with the expectations of $b_N(u)$: $B_n(u)$ [Bernoulli] and $P(z,u)$ [Poisson].

(A) The first (combinatorial) step provides an exact expression for $B_n(u)$

Expectation $P(z,u)$ in the Poisson model	$\Longrightarrow$	Mellin transform $s \mapsto Z(s,u)$ of $z \mapsto P(z,u)$	$\Longrightarrow$	Binomial expression of the expectation $B_n(u)$ in the Bernoulli model

$$B_n(u) = \sum_{\ell=2}^{n} (-1)^\ell \binom{n}{\ell} \Delta(\ell,u), \qquad \text{with} \quad \Delta(s,u) := \frac{1}{\Gamma(-s)} Z(-s,u)$$

(B) The second (analytic) step provides an asymptotic estimate for $B_n(u)$.

– It transforms the binomial expression into an integral expression.

– It transfers the knowledge about singularities of $s \mapsto \Delta(s,u)$ into asymptotic estimates of $B_n(u)$

– It depends on the "tameness" of $s \mapsto \Delta(s,u)$.

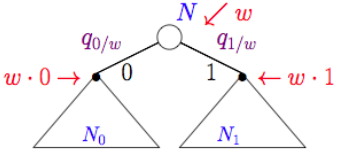
(III) Probabilistic analysis : the combinatorial step.

Profile in the Poisson model

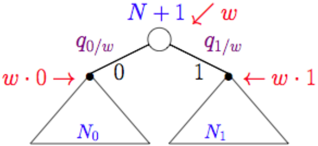
Associate with a source $\mathcal{S}$ all its shifted sources $\mathcal{S}_{(w)}$.

Profile $b^{(w)}_{N,k}$:= the number of full nodes at level k of a digital tree of size N built on the source $\mathcal{S}_{(w)}$

For a trie of size N



For a dst of size $N+1$



$$b^{(w)}_{N,k} = b^{(w\cdot 0)}_{N_0,k-1} + b^{(w\cdot 1)}_{N_1,k-1}$$

$$b^{(w)}_{N}(u) = u b^{(w\cdot 0)}_{N_0}(u) + u b^{(w\cdot 1)}_{N_1}(u)$$

$$b^{(w)}_{N+1,k} = b^{(w\cdot 0)}_{N_0,k-1} + b^{(w\cdot 1)}_{N_1,k-1}$$

$$b^{(w)}_{N+1}(u) = u b^{(w\cdot 0)}_{N_0}(u) + u b^{(w\cdot 1)}_{N_1}(u)$$

$$N = N_0 + N_1$$

The number N_j of nodes in the j-th subtree (that begin with the symbol j) follows a Poisson law of parameter $q_{j|w}\, z$

System of equations on Poisson expectations.

$$\begin{cases} P^{(w)}(z,u) & = z(1-e^{-z}) + u \sum_{i\in\Sigma} P^{(w\cdot\cdot i)}(q_{i|w}\, z, u) & \text{[for trie]} \\ P^{(w)}(z,u) & + \frac{d}{dz} P^{(w)}(z,u) = z + u \sum_{i\in\Sigma} P^{(w\cdot i)}(q_{i|w} z, u) & \text{[for dst]} \end{cases}$$

System of equations on Poisson expectations.

$$\begin{cases} P^{(w)}(z,u) & = z(1-e^{-z}) + u \sum_{i\in\Sigma} P^{(w\cdot i)}(q_{i|w}\, z, u) \qquad \text{[for trie]} \\ P^{(w)}(z,u) & + \frac{d}{dz} P^{(w)}(z,u) = z + u \sum_{i\in\Sigma} P^{(w\cdot i)}(q_{i|w} z, u) \qquad \text{[for dst]} \end{cases}$$

For each type of tree,

a system of functional equations that involves in both cases

– the mapping $z \mapsto qz$ – the shift on words $w \mapsto w \cdot i$

– the derivation d/dz occurs for dst, not for tries.

$\Longrightarrow$ Analysis is more involved for dst.

The Mellin transform of the Poisson expectation.

Begin with the equations satisfied by the Poisson expectations,

$$\begin{cases} P^{(w)}(z,u) & = z(1-e^{-z}) + u \sum_{i\in\Sigma} P^{(w\cdot\cdot i)}(q_{i|w}\, z, u) \qquad \text{[for trie]} \\ P^{(w)}(z,u) & + \frac{d}{dz} P^{(w)}(z,u) = z + u \sum_{i\in\Sigma} P^{(w\cdot i)}(q_{i|w} z, u) \qquad \text{[for dst]} \end{cases}$$

Consider

– their Mellin transforms $Z^{(w)}(s,u) := \int_0^{+\infty} P^{(w)}(x,u) x^{s-1} dx$

– then $\Delta^{(w)}(s,u) := \frac{1}{\Gamma(-s)} Z^{(w)}(-s,u)$,

– then the vector $\mathbf{\Delta}(s,u)$ whose components are $\Delta^{(w)}(s,u)$.

We finally obtain a linear system for $\mathbf{\Delta}(s,u)$

which involves the transition matrix $\mathbf{P}_s$ of the source

$$\begin{cases} \mathbf{\Delta}_T(s,u) & -s\mathbf{1} & = u\,\mathbf{P}_s \mathbf{\Delta}_T(s,u) \qquad \text{[for trie]} \\ \mathbf{\Delta}_D(s,u) & -\mathbf{\Delta}_D(s+1,u) & = u\mathbf{P}_s\, \mathbf{\Delta}_D(s,u) \qquad \text{[for dst]} \end{cases}$$

with $\mathbf{1} = {}^t(1,1,1,\dots)$

The vectors $\mathbf{\Delta}(s,u)$ satisfy,

$$\begin{aligned}
\mathbf{\Delta}_T(s,u) &= s(I - u\mathbf{P}_s)^{-1}\mathbf{1} \\
\mathbf{\Delta}_D(s,u) &= (I - u\mathbf{P}_s)^{-1}\mathbf{\Delta}_D(s+1,u)
\end{aligned}$$

The vectors $\mathbf{\Delta}(s,u)$ satisfy,

$$\begin{aligned} \mathbf{\Delta}_T(s,u) &= s(I-u\mathbf{P}_s)^{-1}\mathbf{1} \\ \mathbf{\Delta}_D(s,u) &= (I-u\mathbf{P}_s)^{-1}\mathbf{\Delta}_D(s+1,u) \end{aligned}$$

For dst, iterate: it appears an infinite product

$$\mathbf{Q}(s,u) := (I-u\mathbf{P}_s)^{-1}\cdot\ldots(I-u\mathbf{P}_{s+k})^{-1}\ldots.$$

The vectors $\mathbf{\Delta}(s,u)$ satisfy,

$$\begin{aligned}
\mathbf{\Delta}_T(s,u) &= s(I - u\mathbf{P}_s)^{-1}\mathbf{1} \\
\mathbf{\Delta}_D(s,u) &= (I - u\mathbf{P}_s)^{-1}\mathbf{\Delta}_D(s+1,u)
\end{aligned}$$

For dst, iterate: it appears an infinite product

$$\mathbf{Q}(s,u) := (I - u\mathbf{P}_s)^{-1} \cdot \ldots (I - u\mathbf{P}_{s+k})^{-1} \ldots .$$

Return to the initial source $\mathcal{S}$ $[\mathbf{E} := {}^t(1,0,0,\ldots)]$

$$\begin{aligned}
\Delta_T(s,u) &= s\ {}^t\mathbf{E}\,(I - u\mathbf{P}_s)^{-1}\,\mathbf{1}, && \text{[for trie]} \\
\Delta_D(s,u) &= {}^t\mathbf{E}\,(I - u\mathbf{P}_s)^{-1}\,\mathbf{Q}(s+1,u)\cdot\mathbf{Q}(2,u)^{-1}\,\mathbf{1} && \text{[for dst]}
\end{aligned}$$

The vectors $\mathbf{\Delta}(s,u)$ satisfy,

$$\begin{aligned}\mathbf{\Delta}_T(s,u) &= s(I-u\mathbf{P}_s)^{-1}\mathbf{1}\\ \mathbf{\Delta}_D(s,u) &= (I-u\mathbf{P}_s)^{-1}\mathbf{\Delta}_D(s+1,u)\end{aligned}$$

For dst, iterate: it appears an infinite product

$$\mathbf{Q}(s,u) := (I-u\mathbf{P}_s)^{-1}\cdot\ldots(I-u\mathbf{P}_{s+k})^{-1}\ldots.$$

Return to the initial source $\mathcal{S}$ $[\mathbf{E} := {}^t(1,0,0,\ldots)]$

$$\begin{aligned}\Delta_T(s,u) &= s\ {}^t\mathbf{E}\,(I-u\mathbf{P}_s)^{-1}\,\mathbf{1}, && \text{[for trie]}\\ \Delta_D(s,u) &= \ {}^t\mathbf{E}\,(I-u\mathbf{P}_s)^{-1}\,\mathbf{Q}(s+1,u)\cdot\mathbf{Q}(2,u)^{-1}\,\mathbf{1} && \text{[for dst]}\end{aligned}$$

An exact expression for $\Delta(s,u)$ $\Longrightarrow$ a binomial expression for $B_n(u)$

The end of the combinatorial step.

(IV) Probabilistic analysis : the analytic step.

Return to the operator $\mathbf{P}_s$ and its quasi-inverse $(I - u\mathbf{P}_s)^{-1}$.

Return to the operator $\mathbf{P}_s$ and its quasi-inverse $(I - u\mathbf{P}_s)^{-1}$.

Remind: $\mathbf{P}_s$ is a matrix whose rows and columns are induced by $\Sigma^\star$.

Its non zero coefficients at row w are located at columns $w.i$,

and are equal to $q_{i|w}^s = (p_{w \cdot i}/p_w)^s$

Return to the operator $\mathbf{P}_s$ and its quasi-inverse $(I - u\mathbf{P}_s)^{-1}$.

Remind: $\mathbf{P}_s$ is a matrix whose rows and columns are induced by $\Sigma^\star$.
Its non zero coefficients at row w are located at columns $w.i$,

and are equal to $q^s_{i|w} = (p_{w \cdot i}/p_w)^s$

The operator $\mathbf{P}_s$ operates on $L^\infty(\Sigma^\star)$ in a natural way:
$L^\infty(\Sigma^\star)$:= the Banach space of the bounded functions $X : \Sigma^\star \to \mathbb{C}$,

endowed with the sup norm.

$$Y = \mathbf{P}_s[X] \qquad \Longleftrightarrow \qquad Y(w) = \mathbf{P}_s[X](w) := \sum_{i \in \Sigma} q^s_{i|w} \, X(w \cdot i)$$

$\mathbf{P} := \mathbf{P}_1$ is stochastic, $\quad\Longrightarrow\quad$ a dominant eigenvalue equal to 1.
Need : precise information for the quasi-inverse $(I - u\mathbf{P}_s)^{-1}$

for u close to 1 and $\Re s$ close to 1.

Related to spectral properties of $\mathbf{P}_s$ on a convenient functional space....

Which functional space ?

There are two cases (for the source)

- (i) The pruned graph becomes finite
- (ii) it remains infinite.

There are two cases (for the tree) $=$ the T-case and the D-case.

For (ii) – we have to find a space where the infinite matrix $\mathbf{P}_s$ well behaves.

– there is an extra difficulty in the D-case: the infinite product

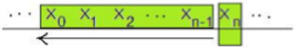
and we thus need a source with a past

Sources with a past

When the symbol X_n is emitted,

– it "looks at" (from its relative point of view) its neighbors,

– which form its reverse past $X_{n-1}, \cdots, X_1, X_0$ in this order



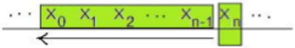
– If w is the previously emitted prefix, it considers its mirror $\phi(w)$.

Sources with a past

When the symbol X_n is emitted,

– it "looks at" (from its relative point of view) its neighbors,

– which form its reverse past $X_{n-1}, \cdots, X_1, X_0$ in this order



– If w is the previously emitted prefix, it considers its mirror $\phi(w)$.

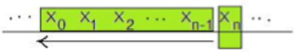
We introduce the g-function defined as $g(i \cdot w) := q_{i|\phi(w)}$

Sources with a past

When the symbol X_n is emitted,

– it "looks at" (from its relative point of view) its neighbors,

– which form its reverse past $X_{n-1}, \cdots, X_1, X_0$ in this order



– If w is the previously emitted prefix, it considers its mirror $\phi(w)$.

We introduce the g-function defined as $g(i \cdot w) := q_{i|\phi(w)}$

Properties of g for "simple" sources:

Memoryless source $\iff g$ constant on each $i \cdot \Sigma^\star$, $i \in \Sigma$

Markov chains of order $1 \iff g$ constant on each $ij \cdot \Sigma^\star$, $i, j \in \Sigma$

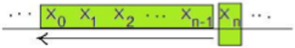
Markov chains of order $k \iff g$ constant on each $w \cdot \Sigma^\star$, $w \in \Sigma^{k+1}$

Sources with a past

When the symbol X_n is emitted,

– it "looks at" (from its relative point of view) its neighbors,

– which form its reverse past $X_{n-1}, \cdots, X_1, X_0$ in this order



– If w is the previously emitted prefix, it considers its mirror $\phi(w)$.

We introduce the g-function defined as $g(i \cdot w) := q_{i|\phi(w)}$

Properties of g for "simple" sources:

Memoryless source $\iff$ g constant on each $i \cdot \Sigma^\star$, $i \in \Sigma$

Markov chains of order $1 \iff g$ constant on each $ij \cdot \Sigma^\star$, $i, j \in \Sigma$

Markov chains of order $k \iff g$ constant on each $w \cdot \Sigma^\star$, $w \in \Sigma^{k+1}$

For "good" sources: one may assume g to be continuous or even Hölder with respect to the usual "distance" δ on $\Sigma^\star$,

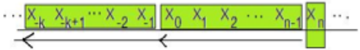
$\delta(x, y) = 2^{-\gamma(x,y)}$ where $\gamma(x, y)$ the coincidence between x and y

Sources with an infinite past.

If the source is regular enough (with a Hölder g-function for instance),

this finite reverse past can be extended to an infinite reverse past

It admits a stationary measure, and we consider the stationary source.

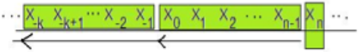


Sources with an infinite past.

If the source is regular enough (with a Hölder g-function for instance),

this finite reverse past can be extended to an infinite reverse past

It admits a stationary measure, and we consider the stationary source.



The (mirror of) transition matrix $\mathbf{P}_s$ is then extended into an operator

– which acts on the space $\mathcal{H}(\Sigma^{\mathbb{N}})$ of Hölder functions $X : \Sigma^{\mathbb{N}} \to \mathbb{C}$.

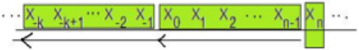
– with now good spectral properties

Sources with an infinite past.

If the source is regular enough (with a Hölder g-function for instance),

this finite reverse past can be extended to an infinite reverse past

It admits a stationary measure, and we consider the stationary source.



The (mirror of) transition matrix $\mathbf{P}_s$ is then extended into an operator

– which acts on the space $\mathcal{H}(\Sigma^{\mathbb{N}})$ of Hölder functions $X : \Sigma^{\mathbb{N}} \to \mathbb{C}$.

– with now good spectral properties

$(I - u\mathbf{P}_s)^{-1}$ is extended to $(I - u\mathbb{H}_s)^{-1}$ which is "tame"

for $\Re s$ and u close to 1.

and the $\Delta(s, u)$ related to the two data structures

$$
\begin{aligned}
\Delta_T(s,u) &= s \ {}^t\mathbf{E}\,(I - u\mathbf{P}_s)^{-1}\,\mathbf{1}, && \text{[for trie]} \\
\Delta_D(s,u) &= {}^t\mathbf{E}\,(I - u\mathbf{P}_s)^{-1}\,\mathbf{Q}(s+1,u)\cdot\mathbf{Q}(2,u)^{-1}\,\mathbf{1} && \text{[for dst]}
\end{aligned}
$$

are also "tame", with a "tameness" of the same type.

(V) Probabilistic analysis : the result.

Main results

Consider a stationary tame source $\mathcal{S}$,

and a digital tree built on n words independently drawn from the source.

We consider a trie (type T) or a dst (type D), with $X \in \{T, D\}$

Main results

Consider a stationary tame source $\mathcal{S}$,

and a digital tree built on n words independently drawn from the source.

We consider a trie (type T) or a dst (type D), with $X \in \{T, D\}$

The mean and the variance of the depth D_n satisfy

$$\mathbb{E}[D_n] = \mu \log n + \mu_X + R(n)$$

$$\mathbb{V}[D_n] = \nu \log n + \nu_X + R(n)$$

Main results

Consider a stationary tame source $\mathcal{S}$,
and a digital tree built on n words independently drawn from the source.
We consider a trie (type T) or a dst (type D), with $X \in \{T, D\}$

The mean and the variance of the depth D_n satisfy

$$\mathbb{E}[D_n] = \mu \log n + \mu_X + R(n)$$
$$\mathbb{V}[D_n] = \nu \log n + \nu_X + R(n)$$

– The dominant constants μ, ν only depend on the source, not on the tree type
– The subdominant constants μ_X, ν_X depend on the source and the tree

The inequality $\mu_T > \mu_D$ holds.

– The remainder terms $R(n)$ depend on the tameness of the source.

Main results

Consider a stationary tame source $\mathcal{S}$,
and a digital tree built on n words independently drawn from the source.
We consider a trie (type T) or a dst (type D), with $X \in \{T, D\}$

The mean and the variance of the depth D_n satisfy

$$\mathbb{E}[D_n] = \mu \log n + \mu_X + R(n)$$
$$\mathbb{V}[D_n] = \nu \log n + \nu_X + R(n)$$

– The dominant constants μ, ν only depend on the source, not on the tree type
– The subdominant constants μ_X, ν_X depend on the source and the tree

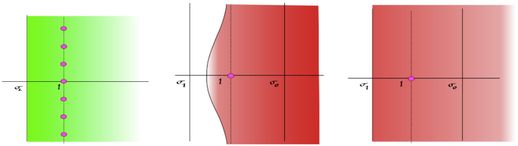
The inequality $\mu_T > \mu_D$ holds.

– The remainder terms $R(n)$ depend on the tameness of the source.

When the source S is not an unbiased memoryless source, one has $\nu \neq 0$ and the depth D_n asymptotically follows a Gaussian law

$$\frac{D_n - \mathbb{E}[D_n]}{\sqrt{\mathbb{V}[D_n]}} \xrightarrow{d} \mathcal{N}(0,1) \qquad \text{[speed of convergence } O(\log n)^{-1/2}\text{]}.$$

Precise results



$$\mathbb{E}[D_n] = \mu \log n + \mu_X + R(n), \quad \mathbb{V}[D_n] = \nu \log n + \nu_X + R(n)$$

Dominant terms	Types of tameness	Remainder terms
$\mu = -\dfrac{1}{\lambda'(1)}$	S-tame	$O(n^{-\delta})$
$\nu = \dfrac{\lambda'(1)^2 - \lambda''(1)}{\lambda'(1)^3}$	H-tame	$O\left(\exp[-(\log n)^{\rho}]\right)$
	P-tame	$\psi(n) + O(n^{-\delta})$

– $\lambda(s)$ is the dominant eigenvalue of the source $(I - \mathbb{H}_s)^{-1} \leadsto 1/(1-\lambda(s))$

– δ and ρ: related to the geometry of the tameness

– $\psi(n)$: a periodic function of $\log n$

Conclusion

Description of the interaction between the source and the data structures,

– via the $\Delta(s, u)$ functions called the mixed Dirichlet series.

– precise comparison between the two structures (trie, dst).

Conclusion

Description of the interaction between the source and the data structures,

- via the $\Delta(s, u)$ functions called the mixed Dirichlet series.
- precise comparison between the two structures (trie, dst).

Other instances of this interaction:

Analyses of sorting or searching algorithms when they deal with words, with the cost "number of symbols that are used for comparing words".

Conclusion

Description of the interaction between the source and the data structures,

– via the $\Delta(s,u)$ functions called the mixed Dirichlet series.

– precise comparison between the two structures (trie, dst).

Other instances of this interaction:

Analyses of sorting or searching algorithms when they deal with words, with the cost "number of symbols that are used for comparing words".

Open question:

Is it possible to return to the analysis of the Lempel-Ziv algorithm?

What happens on the left of the vertical line $\Re s = 1$?

It is important for the analysis to deal with a region $\mathcal{R}$ where $(I - \underline{\widehat{\mathbf{P}}}_s)^{-1}$ is tame : analytic (except for $s = 1$) and of polynomial growth ($\Im s \to \infty$)

What happens on the left of the vertical line $\Re s = 1$?

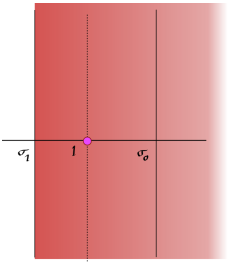
It is important for the analysis to deal with a region $\mathcal{R}$ where $(I - \underline{\widehat{\mathbf{P}}}_s)^{-1}$ is tame : analytic (except for $s = 1$) and of polynomial growth ($\Im s \to \infty$)

Different possible regions $\mathcal{R}$ where $(I - \underline{\widehat{\mathbf{P}}}_s)^{-1}$ is tame.

What happens on the left of the vertical line $\Re s = 1$?

It is important for the analysis to deal with a region $\mathcal{R}$ where $(I - \underline{\widehat{\mathbf{P}}}_s)^{-1}$ is tame : analytic (except for $s = 1$) and of polynomial growth ($\Im s \to \infty$)

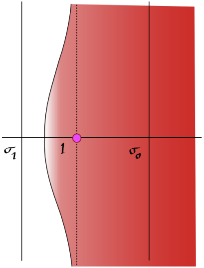
Different possible regions $\mathcal{R}$ where $(I - \underline{\widehat{\mathbf{P}}}_s)^{-1}$ is tame.



Situation 1

Vertical strip

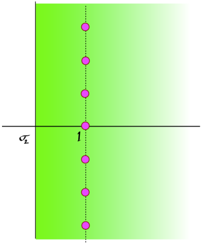
$1 - \sigma \leq a$



Situation 2

Hyperbolic region

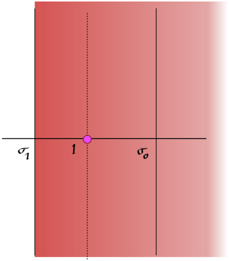
$1 - \sigma \leq t^{-\alpha}$



Situation 3

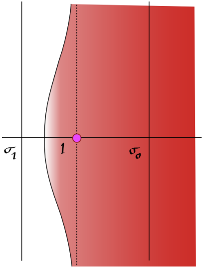
Vertical strip with holes

Possible tameness regions for a simple source



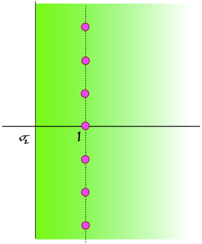
Situation 1

Vertical strip



Situation 2

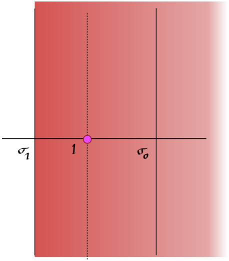
Hyperbolic region



Situation 3

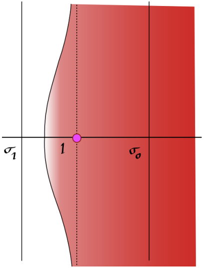
Vertical strip with holes

Possible tameness regions for a simple source



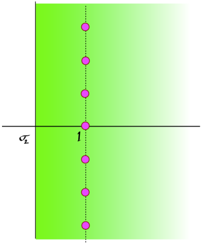
Situation 1

Vertical strip



Situation 2

Hyperbolic region

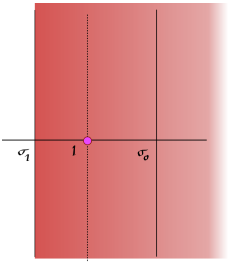


Situation 3

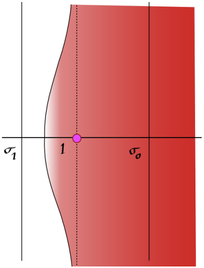
Vertical strip with holes

For which simple sources do these different situations occur?

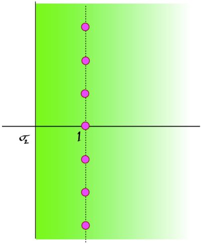
Possible tameness regions for a simple source



Situation 1
Vertical strip



Situation 2
Hyperbolic region



Situation 3
Vertical strip with holes

For which simple sources do these different situations occur?

For memoryless sources relative to probabilities $(p_1, p_2, \ldots, p_r)$

– S1 is impossible

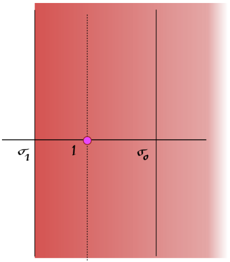
– S3 occurs when all the ratios $\log p_i / \log p_j$ are rational

– S2 occurs if there exists a ratio $\log p_i / \log p_j$

which is "diophantine" [badly approximable by rationals]

For which Lipschitz, stationary, smooth sources do these different situations occur?

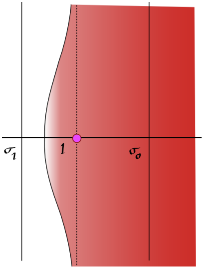
For which Lipschitz, stationary, smooth sources do these different situations occur?



Situation 1

Vertical strip

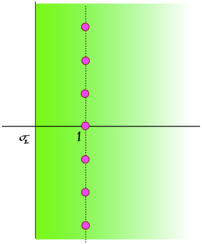
Geometric condition



Situation 2

Hyperbolic region

Arithmetic condition

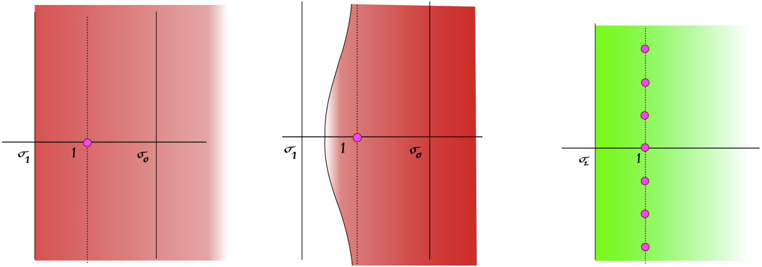


Situation 3

Vertical strip with holes

Periodicity condition

For which Lipschitz, stationary, smooth sources do these different situations occur?



Situation 1	Situation 2	Situation 3
Vertical strip	Hyperbolic region	Vertical strip with holes
Geometric condition	Arithmetic condition	Periodicity condition

– S1: When ? Find some equivalent of the UNI Condition

‘the branches are not too often of the same shape” (??)

– S3: only when the source is conjugated to a simple source.

– S2: when the following condition [DIOP] holds

“there exists two cycles $\mathcal{C}_i$ and $\mathcal{C}_j$

for which the ratio $\log p(\mathcal{C}_i)/\log p(\mathcal{C}_j)$ is “diophantine”