Adaptive Context Trees and Text Clustering

Jean-Philippe Vert

***Abstract*—In the finite-alphabet context we propose four alternatives to fixed-order Markov models to estimate a conditional distribution. They consist in working with a large class of variable-length Markov models represented by context trees, and building an estimator of the conditional distribution with a risk of the same order as the risk of the best estimator for every model simultaneously, in a conditional Kullback–Leibler sense. Such estimators can be used to model complex objects like texts written in natural language and define a notion of similarity between them. This idea is illustrated by experimental results of unsupervised text clustering.**

***Index Terms*—Adaptive mixture of models, context-tree weighting method, mean Kullback risk, text modeling.**

I. Introduction

CONSIDER the problem of measuring the similarity between two long strings in the finite-alphabet context, e.g., two English texts or two DNA sequences. A possible approach to cope with the impossibility of comparing them directly consists in replacing the initial strings by *representations* easier to handle and compare. For this purpose, finite-order Markov models are widely used to catch statistical information from the initial strings and represent them. A trivial example is the so-called *vector-space model* introduced by Salton *et al.* [1] for indexing texts by the statistical distribution of words they contain, which can be seen as a zeroth-order Markov model. Larger order models appear for language models, e.g., in speech or optical character recognition systems (see a survey in [2]),

The order of any Markov model is usually limited because the number of parameters to estimate increases exponentially with it, while the initial strings have finite length. On the other hand, these strings are supposed to have long-range correlations, which might be better caught by models of high order.

Our contribution in this paper is to present and study several alternatives to fixed-order Markov models, and show through an experiment of unsupervised text clustering how to use our results to measure similarities between English texts. More precisely, we consider a larger class $\mathcal{M}$ of Markov models in which the conditional distribution of the next symbol depends on a variable number of preceding symbols. Hence a particular model $m \in \mathcal{M}$ is a parametric family of conditional distributions $\{P_{\theta_m}, \theta_m \in \Theta_m \subset \mathbb{R}^{d(m)}\}$. Such models are interesting because they can catch long-range dependencies on some particular strings without having necessarily an exponentially growing number of parameters. However, it is unknown *a priori* which model to use when confronted with a given text or DNA sequence: we show in the sequel how to use "aggregation rules" among models, i.e., methods of combining several models as opposed to selecting a particular one, to build an estimator $\hat{P}$ whose risk approaches the risk of the best conditional density in the family of models considered (Theorems 4 and 6), in the sense that

$$R_P(\hat{P}) \leq \inf_{m\in\mathcal{M},\, \theta\in\Theta_m} \left\{ R_P(P_{\theta_m}) + \frac{c_N(m)}{N} \right\} \qquad (1)$$

where R_P denotes the distance of a conditional density with the true unknown density P in a Kullback–Leibler sense (see (2)), and $c_N(m)/N$ should be as close as possible as the minimax risk for the model m. The bound (1) is *universal* because it is obtained without restrictive hypotheses on P, in particular P is not required to belong to any model m. Yet if it does it can be approximated at the minimax rate in the model considered (with a loss in the constant), as if this information were known *a priori*: in such a case, we say the estimator is *adaptive*.

There are many connections between our results and universal coding as defined by Davisson [3], which consists in building a probability on the set of strings of length N that approximates simultaneously every probability of a predefined set as N increases, in the Kullback–Leibler distance sense. The literature about universal codes is very rich, and many authors have proposed solutions to problem (1) in that case with $1/N$ being replaced by $\log N$ (including Rissanen and Langdon [4], Davisson [5], Ryabko [6], Willems *et al.* [7], Feder and Merhav [8], and Barron *et al.* [9]). The link with our concern in this paper is that the redundancy criterion of universal coding is the sum of the expected distances we consider for string sizes growing from 1 to N. In spite of this, results are difficult to adapt because a control of the Cesaro mean of a sequence does not always lead to a control of the sequence itself: We overcome this issue of *universal prediction* by using statistical aggregation methods.

This paper is organized as follows. After setting up the statistical framework and presenting the family of Markov models in Section II, we study two estimators for the parameters of a single model in Section III, and prove universal bounds on their risk. In Section IV, we build a probability on the family of Markov models defined earlier, and propose two aggregation methods with universal bounds in Sections V and VI. Each of these two methods can be used to aggregate each of the estimators studied for a given model, therefore resulting in four possible global estimators. In Section VII, we show how using a data-dependent prior on the models improves the estimators, and in Section VIII we propose an efficient implementation in the spirit of the context tree weighting algorithm [7]. Finally, Section IX is devoted

Manuscript received January 21, 2000; revised February 2, 2001. The material in this paper contains research used for the Ph.D. dissertation supervised by O. Catoni.

The author was with the Department of Mathematics and Applications, Ecole Normale Supérieure, 75230 Paris Cedex 05, France. He is now with the Institute for Chemical Research, Bioinformatics Center, Uji, Kyoto 611-0011, Japan (e-mail: Jean-Philippe. Vert@mines.org).

Communicated by G. Lugosi, Associate Editor for Nonparametric Estimation, Classification, and Neural Networks.

Publisher Item Identifier S 0018-9448(01)04415-7.

0018–9448/01$10.00 © 2001 IEEE

to presenting some experimental results. The estimators studied in the paper are used to represent texts written in natural language, and an unsupervised text clustering experiment based on this representation is carried out.

II. Definitions and Framework

Fixed throughout this paper, let $a \in \mathbb{N}^*$ be an integer. Consider an *alphabet*, $\mathcal{A} = \{1, \ldots, a\}$ with size $|\mathcal{A}| = a$ and whose elements are called *letters*. A *string* s is a finite concatenation of letters which can be written as $s = q_{1-l}q_{2-l}\cdots q_0$ with $q_{-i} \in \mathcal{A}$ for $i = 0, 1, \ldots, l-1$. l is called the *length* of the string s and written $l(s)$. The empty string λ has length $l(\lambda) = 0$. The set of all strings is

$$\mathcal{A}^* = \bigcup_{i=0}^{\infty} \mathcal{A}^i.$$

The concatenation of two strings s and s' is written ss'. We say that a string $s = q_{1-l}q_{2-l}\cdots q_0$ is a *suffix* of the string $s' = q'_{1-l}q'_{2-l}\cdots q'_0$ if $l \le l'$ and $q_{-i} = q'_{-i}$ for $i = 0, \ldots, l-1$. The empty string λ is a suffix of all strings.

For any random variable X on a finite space $\mathcal{X}$ with probability distribution P we use the notation $P(x) = \Pr\{X = x\}$. The expectation of a measurable function $f: \mathcal{X} \to \mathbb{R}$ with respect to P is denoted by $\boldsymbol{E}_{P(dX)}f(X)$ or $\boldsymbol{E}_P f(X)$ if there is no ambiguity.

A. Statistical Framework

Let D be an integer, fixed throughout this paper. We consider the measurable product space $(\mathcal{X}\times\mathcal{Y}, \mathcal{B}_1\otimes\mathcal{B}_2)$, where $\mathcal{Y}=\mathcal{A}$, $\mathcal{X}=\mathcal{A}^D$, and $\mathcal{B}_1$ and $\mathcal{B}_2$ are the discrete sigma algebras on $\mathcal{X}$ and $\mathcal{Y}$. We address in this paper the issue of estimating the conditional distribution of a letter $Y \in \mathcal{Y}$ given a string $X \in \mathcal{X}$ based on a series of observations. In order to model the random nature of X and Y we suppose that a family of unknown probability distributions is given

$$\forall N \in \mathbb{N}, \qquad P_N \in \mathcal{M}_+^1\left((\mathcal{X}\times\mathcal{Y})^N, (\mathcal{B}_1\otimes\mathcal{B}_2)^{\otimes N}\right)$$

and we let

$$\{(X_i, Y_i) = Z_i;\ i = 1, \ldots, N\}$$

be the canonical process.

One can, for instance, think of P_N as $P^{\otimes N}$, with P being a probability on $(\mathcal{X}\times\mathcal{Y}, \mathcal{B}_1\otimes\mathcal{B}_2)$, if the observations are supposed to be independent and identically distributed (i.i.d.). However, we will only use the weaker assumption that P_N is *exchangeable*, i.e., that for any permutation σ of $\{1, \ldots, N\}$ and any $A \in (\mathcal{B}_1\otimes\mathcal{B}_2)^N$

$$P_N(Z_1^N \in A) = P_N\left((\sigma Z)_1^N \in A\right),$$

where σZ is the exchanged process

$$(\sigma Z)_i = Z_{\sigma(i)}, \qquad i = 1, \ldots, N.$$

An estimator $\hat{P}_N$ for the conditional probability of Y_N knowing X_N maps any observation (z_1^{N-1}, x_N) to a probability distribution $\hat{P}_N(.\mid z_1^{N-1}, x_N)$ on $\mathcal{Y}$. The performance of an estimator is measured in terms of the Kullback–Leibler divergence $D(.\|.)$ as follows:

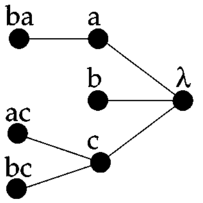

Fig. 1. Representation of the tree model $\{\lambda, a, ba, b, c, ac, bc\}$.

$$\begin{aligned} r_{P_N,\hat{P}_N}&\left(z_1^{N-1}, x_N\right) \\ &= D\left(P_N\left(.\mid z_1^{N-1}, x_N\right) \big\| \hat{P}_N\left(.\mid z_1^{N-1}, x_N\right)\right) \\ &= \sum_{y_N\in\mathcal{Y}} P_N\left(y_N\mid z_1^{N-1}, x_N\right)\log\frac{P_N\left(y_N\mid z_1^{N-1}, x_N\right)}{\hat{P}_N\left(y_N\mid z_1^{N-1}, x_N\right)}. \end{aligned}$$

The observation itself having a random nature, the performance of the estimator is judged according to its expected divergence, which we call the *risk* of the estimator $\hat{P}_N$

$$\begin{aligned} R_{P_N}\left(\hat{P}_N\right) &= \boldsymbol{E}_{P_N}\left(r_{P_N,\hat{P}_N}\left(Z_1^{N-1}, X_N\right)\right) \\ &= \sum_{z_1^N\in(\mathcal{X}\times\mathcal{Y})^N} P_N(z_1^N)\log\frac{P_N\left(y_N\mid z_1^{N-1}, x_N\right)}{\hat{P}_N\left(y_N\mid z_1^{N-1}, x_N\right)}. \end{aligned} \tag{2}$$

This risk is the *conditional Kullback–Leibler divergence* (also called *conditional relative entropy*, see, e.g., [10, p. 22]) and plays a central role in universal coding and prediction (see a survey in [11]).

B. Tree Models

In order to estimate the conditional distribution of Y_N let us consider a family of conditional probability models. As in the statistical literature, a *model* m is a family of conditional distributions which are indexed by a parameter $\theta_m \in \Theta_m \subset \mathbb{R}^{d(m)}$, where $d(m)$ is called the *dimension* of the model m.

The models we consider are represented by *trees*. A tree $\mathcal{S}$ is by definition a nonempty set of strings $\mathcal{S} \subset \mathcal{A}^*$ such that *every suffix of every string of* $\mathcal{S}$ *be also in* $\mathcal{S}$. In particular, this implies that the empty string λ belongs to $\mathcal{S}$. Any tree can be represented graphically as a graph whose vertices are the strings it is made of and whose edges link together every string $s \in \mathcal{S}$ with its suffix of size $l(s)-1$. As an example, Fig. 1 shows a tree $\mathcal{S} = \{\lambda, a, ba, b, c, ac, bc\}$ when $\mathcal{A} = \{a, b, c\}$. The parent of a string $s \in \mathcal{S}$ is its suffix of size $l(s)-1$, and its children are the set of strings $s' \in \mathcal{S}$ of length $l(s') = l(s)+1$ such that s is a suffix of s'. Not that a tree might be *incomplete*, i.e., the number of children of any string $s \in \mathcal{S}$ might be different from 0 or a.

We denote by $\mathcal{C}_D$ the *tree class of memory* D, i.e., the set of trees $\mathcal{S}$ such that for any $s \in \mathcal{S}$, $l(s) \le D$.

For any tree $\mathcal{S} \in \mathcal{C}_D$ the *suffix functional* $s_{\mathcal{S}}: \mathcal{X} \to \mathcal{S}$ is the mapping which transforms any string $x \in \mathcal{X}$ into its longest suffix that is an element of $\mathcal{S}$. If there is no ambiguity on the tree considered, we will also write s instead of $s_{\mathcal{S}}$.

Example 1: The suffix functional s associated with the tree represented in Fig. 1 is such that $s(\cdots bac) = ac$ and $s(\cdots bcc) = c$

Any tree $\mathcal{S} \in \mathcal{C}_D$ can be considered as a conditional distribution model thanks to the following construction.

Definition 1: Let $\mathcal{S} \in \mathcal{C}_D$ be a tree and Σ be the $(a-1)$-dimensional simplex

$$\Sigma = \left\{ \theta \in [0,\, 1]^{\mathcal{Y}} \colon \sum_{y \in \mathcal{Y}} \theta(y) = 1 \right\}.$$

For any $\theta = (\theta_s)_{s \in \mathcal{S}} \in \Sigma^{\mathcal{S}}$ let $P_{\mathcal{S},\,\theta}$ denote the conditional probability density on $\mathcal{X} \times \mathcal{Y}$ defined by

$$\forall\, (x,\, y) \in \mathcal{X} \times \mathcal{Y}, \qquad P_{\mathcal{S},\theta}(y \,|\, x) \overset{def}{=} \theta_{s_{\mathcal{S}}(x)}(y).$$

The *tree model* $\mathcal{S}$ is by definition the set of conditional densities $\{P_{\mathcal{S},\,\theta} \colon \theta \in \Sigma^{\mathcal{S}}\}$.

As a result, a tree model $\mathcal{S}$ is a model with dimension $d(\mathcal{S}) = |\mathcal{S}| \times (a-1)$.

III. Estimator for a Given Tree Model

Let us first consider the case when a tree model $\mathcal{S} \in \mathcal{C}_D$ is given and one wants to use the observations Z_1^{N-1} is order to estimate a parameter $\hat{\theta}(Z_1^{N-1}) \in \Sigma^{\mathcal{S}}$ such that $R_{P_N}(P_{\mathcal{S},\,\hat{\theta}(Z_1^{N-1})})$ is "small." We propose two estimators for this problem: the first one is the well-known *Laplace estimator* for which we generalize known universal bounds (Theorem 1), while the second one is a new estimator for which we prove a better bound when the support of the conditional distribution is smaller than the whole alphabet (Theorem 2). $\mathcal{S}$ being fixed, we will use the notation $s(.)$ instead of $s_{\mathcal{S}}(.)$ for the suffix functional associated with $\mathcal{S}$.

Remark 1: The problem of parameter estimation for an i.i.d. source on a finite space is well known in information theory. It seems that first the method was considered in [12]; then the problem of optimal estimation was considered in [13] and an asymptotically optimal method was suggested. Recently, new results about exact prediction were found in [14]. The results that follow are nonasymptotic (as opposed to [13]) and remain true if the samples are not i.i.d. but only drawn from an exchangeable distribution. Even though the estimators we study are not asymptotically minimax (as opposed to [13]) the nonasymptotic upper bounds we obtain are of the order of the minimax risk.

A. Laplace Estimator

For any $n \in \mathbb{N}$ let us introduce the random variables

$$\begin{cases} \forall\, (y,\, s) \in \mathcal{Y} \times \mathcal{S}, \\ \quad \mu_n(s,\, y) = \sum_{i=1}^{n} \mathbf{1}(s(X_i) = s \text{ and } Y_i = y) \\ \forall\, s \in \mathcal{S}, \\ \quad \nu_n(s) = \sum_{i=1}^{n} \mathbf{1}(s(X_i) = s). \end{cases} \tag{3}$$

Hence $\nu_n(s)$ counts the number of samples Z_i in $Z_1, \ldots, Z_n$ such that X_i is mapped to s by the suffix functional $s(.)$, and $\mu_n(s,\, y)$ counts the number of samples in that subset such that $Y_i = y$.

A node $s \in \mathcal{S}$ is said to be *visited* by Z_1^N if $\nu_N(s) > 0$, and we denote by $v_N(\mathcal{S})$ the random set of visited nodes, i.e.,

$$v_N(\mathcal{S}) \overset{\text{def}}{=} \{s \in \mathcal{S} \colon \nu_N(s) > 0\}.$$

The Laplace estimator $\hat{\theta}$ is defined by

$$\forall\, (s,\, y) \in \mathcal{S} \times \mathcal{Y}, \qquad \hat{\theta}_s(y) = \frac{\mu_{N-1}(s,\, y) + 1}{\nu_{N-1}(s) + a}$$

and results in an estimator which we call the *Laplace estimator for the tree* $\mathcal{S}$ defined by the formula

$$\forall\, z_1^N \in (\mathcal{X} \times \mathcal{Y})^N, \qquad Q_{\mathcal{S}}^N\left(y_N \,|\, x_N;\, z_1^{N-1}\right) = \frac{\mu_{N-1}(s(x_N),\, y_N) + 1}{\nu_{N-1}(s(x_N)) + a}. \tag{4}$$

The following theorem gives an upper bound for the risk of this estimator:

Theorem 1: For any exchangeable distribution P_N on $(\mathcal{X} \times \mathcal{Y})^N$ and for any tree $\mathcal{S} \in \mathcal{C}_D$ the risk of the Laplace estimator for the tree $\mathcal{S}$ satisfies

$$\begin{aligned} R_{P_N}(Q_{\mathcal{S}}^N) &\le \inf_{\theta \in \Sigma^{\mathcal{S}}} R_{P_N}(P_{\mathcal{S},\theta}) + \frac{a-1}{N} \boldsymbol{E}_{P_N} |v_N(\mathcal{S})| \\ &\le \inf_{\theta \in \Sigma^{\mathcal{S}}} R_{P_N}(P_{\mathcal{S},\theta}) + \frac{a-1}{N} |\mathcal{S}|. \end{aligned}$$

Remark 2: The first inequality of Theorem 1 shows that the risk bound depends on the design distribution, i.e., on the distribution of X_1^N, and, therefore, that the Laplace estimator can *adapt* to it.

When $\mathcal{S}$ is reduced to a single node, this result is proven in [12] when P_N is a product distribution and in [15] when P_N is exchangeable. Here we generalize the method of proof of the latter for a general tree model $\mathcal{S}$ (see also [16] for a similar result in the case of decision trees).

Proof of Theorem 1: First observe that for any $s \in \mathcal{S}$

$$\begin{aligned} \nu_N(s(X_N)) &= \sum_{i=1}^{N} \mathbf{1}(s(X_i) = s(X_N)) \\ &= \sum_{i=1}^{N-1} \mathbf{1}(s(X_i) = s(X_N)) + 1 \\ &= \nu_{N-1}(s(X_N)) + 1. \end{aligned}$$

A similar computation shows that for any $(s,\, y) \in \mathcal{S} \times \mathcal{Y}$

$$\mu_N(s(X_N),\, Y_N) = \mu_{N-1}(s(X_N),\, Y_N)) + 1.$$

As a result, the Laplace estimator (4) can be rewritten in terms of μ_N and ν_N as follows:

$$\forall\, z_1^N \in (\mathcal{X} \times \mathcal{Y})^N, \qquad Q_{\mathcal{S}}^N\left(y_N \,|\, x_N;\, z_1^{N-1}\right) = \frac{\mu_N(s(x_N),\, y_N)}{\nu_N(s(x_N)) + a - 1}.$$

Observe also that the maximum-likelihood estimator for $\prod_{i=1}^{N} P_{\mathcal{S},\theta}(Y_i \,|\, X_i)$ is $\hat{\theta}_s(y) = \mu_N(s, y)/\nu_N(s)$ with corresponding log-likelihood

$$\sup_{\theta \in \Sigma^{\mathcal{S}}} \log \prod_{i=1}^{N} P_{\mathcal{S},\theta}(Y_i \,|\, X_i) = \sum_{\substack{s \in \mathcal{S} \\ \nu_N(s) > 0}} \sum_{y \in \mathcal{Y}} \mu_N(s, y) \log \frac{\mu_N(s, y)}{\nu_N(s)}.$$

Using the fact that P_N is exchangeable to get the first equality and the fact that μ_N and ν_N are invariant under permutations of $\{1, \ldots, N\}$ to get the second, we can now write

$$\begin{aligned}
&\boldsymbol{E}_{P_N} \log \frac{1}{Q_{\mathcal{S}}^N\left(Y_N \,|\, X_N;\, Z_1^{N-1}\right)} \\
&\quad = -\frac{1}{N} \boldsymbol{E}_{P_N} \sum_{i=1}^{N} \log Q_{\mathcal{S}}^N\left(Y_i \,|\, X_i;\, Z_k,\, k \neq i,\, 1 \leq k \leq N\right) \\
&\quad = -\frac{1}{N} \boldsymbol{E}_{P_N} \sum_{s \in \mathcal{S}} \sum_{y \in \mathcal{Y}} \mu_N(s, y) \log \frac{\mu_N(s, y)}{\nu_N(s) + a - 1} \\
&\quad = \boldsymbol{E}_{P_N} \inf_{\theta \in \Sigma^{\mathcal{S}}} -\frac{1}{N} \log \prod_{i=1}^{N} P_{\mathcal{S},\theta}(Y_i \,|\, X_i) \\
&\qquad + \frac{1}{N} \sum_{\substack{s \in \mathcal{S} \\ \nu_N(s) > 0}} \boldsymbol{E}_{P_N} \nu_n(s) \log\left(1 + \frac{a-1}{\nu_n(s)}\right) \\
&\quad \leq \inf_{\theta \in \Sigma^{\mathcal{S}}} \boldsymbol{E}_{P_N} \log \frac{1}{P_{\mathcal{S},\theta}(Y_N \,|\, X_N)} + \frac{a-1}{N} \boldsymbol{E}_{P_N} |v_N(\mathcal{S})|.
\end{aligned}$$

Theorem 1 follows by adding

$$\boldsymbol{E}_{P_N} \log P_N(Y_N \,|\, X_N;\, Z_1^{N-1})$$

to both sides of the inequality and observing that $v_N(\mathcal{S}) \subset \mathcal{S}$ implies $|v_N(\mathcal{S})| \leq |\mathcal{S}|$. □

B. Adaptive Laplace Estimator

In this section, we suppose that P_N is a product measure $P^{\otimes N}$ with $P \in \mathcal{M}_+^1(\mathcal{X} \times \mathcal{Y})$, i.e., $Z_1, \ldots, Z_N$ are supposed to be i.i.d. with common distribution P.

Suppose that for every $s \in \mathcal{S}$ the support of the conditional distribution $P(Y \,|\, s(X) = s)$ is known to be a subset $\mathcal{A}_s \subset \mathcal{A}$ of size $a(s) = |\mathcal{A}_s|$, i.e.,

$$\forall (x, y) \in \mathcal{X} \times \mathcal{Y}, \qquad \begin{cases} P(y \,|\, x) > 0, & \text{if } y \in \mathcal{A}_{s(x)} \\ P(y \,|\, x) = 0, & \text{otherwise.} \end{cases}$$

In that case, one could replace the Laplace estimator for the tree $\mathcal{S}$ by the following estimator which takes into account the information about the supports:

$$\overline{Q}_{\mathcal{S}}^N\left(y_N \,|\, x_N;\, z_1^{N-1}\right) = \begin{cases} \dfrac{\mu_{N-1}(s(x_N), y_N) + 1}{\nu_{N-1}(s(x_N)) + a(s)}, & \text{if } y_N \in \mathcal{A}_{s(x_n)} \\ 0, & \text{otherwise.} \end{cases}$$

Using a computation similar to the one in the proof of Theorem 1, it is straightforward to show that this estimator satisfies

$$R_{P_N}\left(\overline{Q}_{\mathcal{S}}^N\right) \leq \inf_{\theta \in \Sigma^{|\mathcal{S}|}} R_{P_N}(P_{\mathcal{S},\theta}) + \frac{\sum_{s \in \mathcal{S}} (a(s) - 1)}{N} \tag{5}$$

which is a smaller upper bound than the one given in Theorem 1 if $a(s) < a$ for some $s \in \mathcal{S}$. However, this estimator requires prior knowledge of the supports $\{\mathcal{A}_s\}_{s \in \mathcal{S}}$. In case these supports are not known, it is still possible to observe the size of the empirical supports given by

$$\forall (n, s) \in \mathbb{N} \times \mathcal{S}, \qquad a_n(s) = \sum_{y \in \mathcal{A}} \mathbf{1}(\mu_n(s, y) > 0).$$

Using these observations we define the *adaptive Laplace estimator for the tree* $\mathcal{S}$ by the formula, $\forall z_1^N \in (\mathcal{X} \times \mathcal{Y})^N$,

$$\tilde{Q}_{\mathcal{S}}^N\left(y_N \,|\, x_N;\, z_1^{N-1}\right) = \begin{cases} \dfrac{\mu_{N-1}(s(x_N), y_N) + \frac{a_{N-1}(s(x_N))}{a}}{\nu_{N-1}(s(x_N), y_N) + a_{N-1}(s(x_N))}, & \\ & \text{if } \nu_{N-1}(s(x_N)) > 0 \\ \dfrac{1}{a}, & \text{otherwise.} \end{cases}$$

The effect of this modification to the Laplace estimator is to "boost" the estimated probabilities of letters which have already been observed. It is easy to check that

$$\forall \left(z_1^{N-1}, x_N\right) \in (\mathcal{X} \times \mathcal{Y})^{N-1} \times \mathcal{X}, \qquad \sum_{y \in \mathcal{Y}} \tilde{Q}_{\mathcal{S}}^N\left(y \,|\, x_N;\, z_1^{N-1}\right) = 1$$

which ensures that $\tilde{Q}_{\mathcal{S}}^N$ is an admissible conditional probability density. The risk of this estimator can be upper-bounded as follows.

Theorem 2: For any probability distribution P on $\mathcal{X} \times \mathcal{Y}$ and $P_N = P^{\otimes N}$, for any incomplete tree model $\mathcal{S} \in \mathcal{C}_D$

$$R_{P_N}\left(\tilde{Q}_{\mathcal{S}}^N\right) \leq \inf_{\theta \in \Sigma^{\mathcal{S}}} R_{P_N}(P_{\mathcal{S},\theta}) + \frac{\sum_{s \in \mathcal{S}} \gamma_N(s)}{N}$$

with

$$\forall s \in \mathcal{S}, \qquad \gamma_N(s) = a(s)\left(1 - \frac{a(s)}{a}\right) + a(s) - 1 + \zeta_{P,N}(s)$$

where $\lim_{N \mapsto \infty} \zeta_{P,N}(s) = 0$ for any $s \in \mathcal{S}$ (a precise expression of $\zeta_{P,N}(s)$ is given in the proof in (8)).

Remark 3: Up to the vanishing terms $\zeta_{P,N}(s)$, the upper bound provided in Theorem 2 is smaller than the upper bound provided by Theorem 1 for the Laplace estimator by a factor

$$\frac{1}{N} \sum_{s \in \mathcal{S}} \left(a - 1 - a(s) + 1 - a(s)\left(1 - \frac{a(s)}{a}\right)\right) = \frac{1}{N} \sum_{s \in \mathcal{S}} \frac{(a - a(s))^2}{a}$$

which is always positive. Therefore, the asymptotic rate of convergence to zero is smaller for the adaptive Laplace estimator than for the Laplace estimator if $a(s) < a$ for some s.

However, by (5), the corresponding rate of convergence for the risk of the Laplace estimator $\overline{Q}^N_{\mathcal{S}}$ in the case $\{\mathcal{A}_s\}_{s\in\mathcal{S}}$ is known is $\sum_{s\in\mathcal{S}}(a(s)-1)/N$, which is smaller than the upper bound of Theorem 2 by a factor

$$\frac{1}{N}\sum_{s\in\mathcal{S}} a(s)\left(1-\frac{a(s)}{a}\right) \le \frac{1}{N}\sum_{s\in\mathcal{S}} a(s).$$

This factor can be considered as the "cost" of not knowing $\{\mathcal{A}_s\}_{s\in\mathcal{S}}$.

Proof of Theorem 2: First observe that if

$$\mu_N(s(X_N),\, Y_N) = 1$$

then, for all $i < N$, $s(X_i) \neq s(X_N)$, or $Y_i \neq Y_N$. As a result

$$a_N(s(X_N)) = a_{N-1}(s(X_N)) + 1.$$

On the other hand, if $\mu_N(s(X_N),\, Y_N) > 1$ then

$$a_N(s(X_N)) = a_{N-1}(s(X_N)).$$

Therefore, we can compute

$$\begin{aligned}
&\boldsymbol{E}_{P_N} \log \frac{1}{\tilde{Q}^N_{\mathcal{S}}\left(Y_N \,|\, X_N;\, Z_1^{N-1}\right)}\\
&\quad= -\frac{1}{N}\boldsymbol{E}_{P_N}\sum_{i=1}^{N} \log \tilde{Q}^N_{\mathcal{S}}(Y_i \,|\, X_i;\, Z_k,\, k\neq i,\, 1\le k\le N)\\
&\quad= -\frac{1}{N}\boldsymbol{E}_{P_N}\sum_{\substack{s\in\mathcal{S}\\ \nu_N(s)>1}}\\
&\qquad\times\left(\sum_{y\in\mathcal{Y}}\mu_N(s,\, y)\log\frac{\mu_N(s,\, y)-1+\frac{a_{N-1}(s)}{a}}{\nu_N(s)-1+a_{N-1}(s)}\right)\\
&\qquad-\frac{1}{N}\boldsymbol{E}_{P_N}\sum_{\substack{s\in\mathcal{S}\\ \nu_N(s)=1}}\log\frac{1}{a}\\
&\quad= -\frac{1}{N}\boldsymbol{E}_{P_N}\sum_{\substack{s\in\mathcal{S}\\ \nu_N(s)>1}}\\
&\qquad\times\left(\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\ge 2}}\mu_N(s,\, y)\log\frac{\mu_N(s,\, y)-1+\frac{a_N(s)}{a}}{\nu_N(s)+a_N(s)-1}\right.\\
&\qquad\qquad\left.+\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)=1}}\log\frac{\frac{a_N(s)-1}{a}}{\nu_N(s)+a_N(s)-2}\right)\\
&\qquad-\frac{1}{N}\boldsymbol{E}_{P_N}\sum_{\substack{s\in\mathcal{S}\\ \nu_N(s)=1}}\log\frac{1}{a}\\
&\quad\le \inf_{\theta\in\Sigma^{\mathcal{S}}}\boldsymbol{E}_{P_N}\log\frac{1}{P_{\mathcal{S},\theta}(Y_N \,|\, X_N)}\\
&\qquad+\frac{1}{N}\sum_{s\in\mathcal{S}}(A_s+B_s+C_s+D_s+E_s)
\end{aligned}$$

with

$$\begin{cases}
A_s = \boldsymbol{E}_{P_N}\displaystyle\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\ge 2}}\mu_N(s,y)\log\frac{\mu_N(s,\, y)}{\mu_N(s,\, y)-1+\frac{a_N(s)}{a}}\\
B_s = \boldsymbol{E}_{P_N}\displaystyle\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)\ge 2}}\mu_N(s,\, y)\log\frac{\nu_N(s)+a_N(s)-1}{\nu_N(s)}\\
C_s = \boldsymbol{E}_{P_N}\left(\mathbf{1}(\nu_N(s)>1)\displaystyle\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)=1}}\log\frac{a}{a_N(s)-1}\right)\\
D_s = \boldsymbol{E}_{P_N}\left(\mathbf{1}(\nu_N(s)>1)\displaystyle\sum_{\substack{y\in\mathcal{Y}\\ \mu_N(s,y)=1}}\right.\\
\qquad\qquad\left.\log\dfrac{\nu_N(s)+a_N(s)-2}{\nu_N(s)}\right)\\
E_s = \boldsymbol{E}_{P_N}(\mathbf{1}(\nu_N(s)=1)\log a).
\end{cases}$$

For any $s\in\mathcal{S}$ and $y\in\mathcal{Y}$, let

$$P(s) = \Pr\{s(X)=s\}$$

and

$$\theta_s(y) = \Pr\{Y=y \,|\, s(X)=x\}.$$

Then, $\nu_N(s)$ and $\mu_N(s,\, y)$ are binomial variables $B(N,\, P(s))$ and $B(N,\, P(s)\theta_s(y))$. Let $\epsilon > 0$ be defined by

$$\epsilon = \min_{\substack{s\in\mathcal{S}\\ P(s)>0}}\ \min_{y\in\mathcal{A}_s} P(s)\theta_s(y).$$

Then, for any $s\in\mathcal{S}$ such that $P(s)>0$, the expectation of the empirical support size satisfies

$$\begin{aligned}
a(s)-\boldsymbol{E}_{P_N}a_N(s) &= a(s)-\sum_{k=0}^{a(s)}k.\Pr\{a_N(s)=k\}\\
&\le a(s)\Pr\{a_N(s)<a(s)\}\\
&\le a(s)\left(\sum_{y\in\mathcal{A}_s}\Pr\{\mu_N(s,\, y)=0\}\right)\\
&\le a(s)\sum_{y\in\mathcal{A}_s}(1-\epsilon)^N\\
&\le a(s)^2\, e^{-N\epsilon}.
\end{aligned}\tag{6}$$

On the other hand, if $\mu_N(s,\, y)\ge 2$ then

$$\frac{1}{\mu_N(s,\, y)-1}\le\frac{3}{\mu_N(s,\, y)+1}$$

and, therefore, for any $s \in \mathcal{S}$ such that $P(s) > 0$ we have

$$\begin{aligned}
\boldsymbol{E}_{P_N} \sum_{\substack{y \in \mathcal{Y} \\ \mu_N(s,y) \geq 2}} \frac{1}{\mu_N(s, y) - 1}
&\leq 3\boldsymbol{E}_{P_N} \sum_{\substack{y \in \mathcal{Y} \\ \mu_N(s,y) \geq 2}} \frac{1}{\mu_N(s, y) + 1} \\
&\leq 3 \sum_{y \in \mathcal{A}_s} \boldsymbol{E}_{P_N} \frac{1}{1 + B(N, P(s)\theta_s(y))} \\
&\leq 3 \sum_{y \in \mathcal{A}_s} \frac{1}{(N+1)P(s)\theta_s(s)} \\
&\leq \frac{3a(s)}{N\epsilon} \qquad (7)
\end{aligned}$$

where we used the fact (see, e.g., [17, p. 587]) that for a binomial $B(n, p)$

$$\boldsymbol{E}1/(1 + B(n, p)) \leq 1/((n+1)p).$$

We can now upper-bound the five terms for any $s \in \mathcal{S}$ such that $P(s) > 0$. For A_s we write

$$\begin{aligned}
A_s \leq &\boldsymbol{E}_{P_N} \sum_{\substack{y \in \mathcal{Y} \\ \mu_N(s,y) \geq 2}} \mu_N(s,y) \frac{1 - \frac{a_N(s)}{a}}{\mu_N(s, y) - 1 + \frac{a_N(s)}{a}} \\
\leq &\boldsymbol{E}_{P_N} \sum_{\substack{y \in \mathcal{Y} \\ \mu_N(s,y) \geq 2}} \left[1 - \frac{a_N(s)}{a} + \frac{\left(1 - \frac{a_N(s)}{a}\right)^2}{\mu_N(s, y) - 1 + \frac{a_N(s)}{a}} \right] \\
\leq &a(s) \left(1 - \frac{a(s)}{a}\right) + \frac{a(s)}{a} \left(a(s) - \boldsymbol{E}_{P_N} a_N(s)\right) \\
&+ \boldsymbol{E}_{P_N} \sum_{\substack{y \in \mathcal{Y} \\ \mu_N(s,y) \geq 2}} \frac{1}{\mu_N(s, y) - 1} \\
\leq &a(s) \left(1 - \frac{a(s)}{a}\right) + \frac{a(s)^3}{a} e^{-N\epsilon} + \frac{3a(s)}{N\epsilon}
\end{aligned}$$

where (6) and (7) are used to get the last inequality. The terms B_s and D_s can be taken together

$$\begin{aligned}
B_s + D_s \leq &\boldsymbol{E}_{P_N} \left(\mathbf{1}(\nu_N(s) > 1) \sum_{y \in \mathcal{A}_s} \mu_N(s, y) \right. \\
&\left. \times \log \frac{\nu_N(s) + a(s) - 1}{\nu_N(s)} \right) \\
\leq &\boldsymbol{E}_{P_N} \left(\mathbf{1}(\nu_N(s) > 1) \nu_N(s) \log \left(1 + \frac{a(s) - 1}{\nu_N(s)}\right)\right) \\
\leq &a(s) - 1.
\end{aligned}$$

Finally, one can observe that if $\mu_N(s, y) = 1$ and $\nu_N(s) \geq 2$ then $a_N(s) \geq 2$. This provides an upper bound for the integrand in C_s and, therefore,

$$\begin{aligned}
C_s + E_s \leq &\boldsymbol{E}_{P_N} \left(\mathbf{1}(\nu_N(s) > 1) \sum_{\substack{y \in \mathcal{Y} \\ \mu_N(s,y) = 1}} \right. \\
&\left. \times \log a + \mathbf{1}(\nu_N(s) = 1) \log a \right) \\
\leq &\log(a) \\
&\times \left(\sum_{y \in \mathcal{A}_s} \Pr\{\mu_N(s, y) = 1\} + \Pr\{\nu_N(s) = 1\} \right) \\
\leq &N \log(a) \\
&\times \left[(1 - P(s))^{N-1} + \sum_{y \in \mathcal{A}_s} (1 - P(s)\theta_s(y))^{N-1} \right] \\
\leq &N \log(a)(a(s) + 1) e^{-(N-1)\epsilon}.
\end{aligned}$$

We can now sum up the upper bounds obtained for A_s, B_s, C_s, D_s, and E_s to get

$$\begin{aligned}
R_{P_N}\left(\tilde{Q}_{\mathcal{S}}^N\right) \leq &\inf_{\theta \in \Sigma^{\mathcal{S}}} R_{P_N}(P_{\mathcal{S},\theta}) + \frac{1}{N} \sum_{\substack{s \in \mathcal{S} \\ P(s) > 0}} \\
&\times \left[a(s) - 1 + a(s) \left(1 - \frac{a(s)}{a}\right) + \zeta_{P,N}(s) \right]
\end{aligned}$$

with

$$\begin{aligned}
\zeta_{P,N}(s) = &\frac{a(s)^3}{a} e^{-N\epsilon} + \frac{3a(s)}{N\epsilon} \\
&+ N \log(a)(a(s) + 1) e^{-(N-1)\epsilon}. \qquad (8)
\end{aligned}$$

This finishes the proof of Theorem 2. □

IV. Probability on the Model Space

The goal in the rest of this paper is to build estimators which satisfy risk bounds like (1). For this purpose, we propose to use aggregation methods introduced by Catoni (the progressive mixture estimator in [15] and the Gibbs estimator in [18]), both of which require a prior probability distribution to be given on the model set. The idea of setting a probability on a model space is well known in source coding and prediction: besides underlying any Bayesian approach it was suggested in [6] and [19] to obtain nonasymptotic risk bounds and later this idea was used in many papers (see, for example, [7] and [20]).

If π is a probability distribution on a model space $\mathcal{M}$ then $\log 1/\pi(m)$ is called *model risk*. The choice of π is arbitrary, but has an influence on the performance of the aggregated estimator. Optimizing this choice is impossible without further assumptions on the true probability distribution P and the approximation properties of the family of models considered.

In addition to performance the possibility of a fast implementation should be regarded as a guideline for the choice of a prior distribution π. For instance, the prior model probability distribution considered in the context tree weighting algorithm [7] leads to a remarkably efficient implementation, which should be regarded as a fundamental advantage of the algorithm.

Generalizing the idea of the context tree weighting method, let us define a probability distribution π_D on $\mathcal{C}_D$, the tree class of memory D, as follows:

$$\forall \mathcal{S} \in \mathcal{C}_D, \qquad \pi_D(\mathcal{S}) = c_D^{|\mathcal{S}|}$$

where $c_D \in \mathbb{R}$ satisfies

$$\sum_{\mathcal{S}\in\mathcal{C}_D} c_D^{|\mathcal{S}|} = 1. \tag{9}$$

The model risk is then linear with respect to the size of the model, because

$$\forall \mathcal{S} \in \mathcal{C}_D, \qquad \log\frac{1}{\pi_D(\mathcal{S})} = |\mathcal{S}|\log\frac{1}{c_D}. \tag{10}$$

The prior π_D will be used in the following sections to build convex combinations of different models. We will obtain particular upper bounds for the risks with this arbitrary choice (Theorems 4 and 6), but the reader should be aware that any different choice of prior is possible and would lead to different upper bounds. We propose to chose a prior which results in a model risk proportional to $|\mathcal{S}|$ because the "parameter risk," i.e., the risk of an estimator for the model $\mathcal{S}$ like the Laplace estimator, is also linear in $|\mathcal{S}|$ (Theorem 1).

The following lemma provides a useful upper bound on the model risk independent of D.

Lemma 1: The family of probabilities $\{\pi_D\}_{D\in\mathbb{N}}$ satisfies

$$\forall D \in \mathbb{N}, \forall \mathcal{S} \in \mathcal{C}_D, \qquad \log\frac{1}{\pi_D(\mathcal{S})} \le |\mathcal{S}|(\log(a)+1).$$

Proof of Lemma 1: By (9), it is clear that $(c_D)_{D\in\mathbb{N}}$ is a decreasing function of D, because $\mathcal{C}_D \subset \mathcal{C}_{D+1}$ for any $D \in \mathbb{N}$. Therefore, this nonnegative series has a limit $c = \lim_{D\to\infty} c_D$, such that $\forall D \in \mathbb{N}$, $c \le c_D$.

For any $(D, x) \in \mathbb{N}\times\mathbb{R}$ let

$$u_D(x) = \sum_{\mathcal{S}\in\mathcal{C}_D} x^{|\mathcal{S}|}.$$

The function $u_D(x)$ is increasing with x and D, and by definition $u_D(c_D) = 1$ for any $D \in \mathbb{N}$. Therefore, $u_D(c) \le 1$ for any $D \in \mathbb{N}$, and

$$\lim_{D\to\infty} u_D(c) \le 1. \tag{11}$$

By decomposing any tree $\mathcal{S} \in \mathcal{C}_D$ as the root node and a (eventually empty) subtrees $(\mathcal{S}_1, \dots, \mathcal{S}_a) \in (\mathcal{C}_{D-1}\cup\{\emptyset\})^a$ one gets the following inductive relation:

$$\begin{aligned} u_D(x) &= \sum_{\mathcal{S}\in\mathcal{C}_D} x^{|\mathcal{S}|} \\ &= \sum_{(\mathcal{S}_1,\dots,\mathcal{S}_a)\in(\mathcal{C}_{D-1}\cup\{\emptyset\})^a} x^{1+|\mathcal{S}_1|+\cdots+|\mathcal{S}_a|} \\ &= x(u_{D-1}(x)+1)^a. \end{aligned}$$

If we introduce the function $f_x(y) = x(1+y)^a$ then this can be rewritten

$$u_D(x) = f_x(u_{D-1}(x)).$$

It is well known that for $u_D(x)$ to stay bounded when D tends to infinity it is necessary that the equation $f_x(y) = y$ have a solution y. By (11), this implies that $f_c(y) - y$ must be equal to zero for some y.

If we now study the function $g_x(y) = f_x(y) - y$ its derivative is

$$g'_x(y) = ax(1+y)^{a-1} - 1$$

therefore, g_x is minimum for y^* such that $g'_x(y^*) = 0$, i.e.,

$$y^* = (ax)^{\frac{1}{1-a}} - 1.$$

As a result, the minimum value of g_x is

$$g_x(y^*) = 1 - \frac{a-1}{a}(ax)^{\frac{1}{1-a}}.$$

The necessary condition that $f_c(y) - y = 0$ for some y is equivalent to $g_c(y^*) \le 0$, i.e.,

$$c \le \frac{(a-1)^{a-1}}{a^a}$$

which implies

$$\begin{aligned} \log\frac{1}{c} &\le a\log(a) - (a-1)\log(a-1) \\ &\le \log(a) + (a-1)\log\frac{a}{a-1} \\ &\le \log(a) + 1. \end{aligned}$$

Lemma 1 now follows from this inequality, the fact that $c \le c_D$, and (10). □

V. Aggregation Using a Progressive Mixture Estimator

In Section III, we presented two estimators for the parameters of every given model $\mathcal{S}$: the Laplace estimator $Q_\mathcal{S}$ and the adaptive Laplace estimator $\tilde{Q}_\mathcal{S}$. In this section, we show how to aggregate the Laplace (resp., adaptive Laplace) estimators for various $\mathcal{S}$, i.e., build a convex combination of the $\{Q_\mathcal{S}\}_{\mathcal{S}\in\mathcal{C}_D}$ (resp., $\{\tilde{Q}_\mathcal{S}\}_{\mathcal{S}\in\mathcal{C}_D}$), by using the so-called *progressive mixture estimator*, introduced by Catoni in [15]. Instead of selecting one model $\hat{\mathcal{S}}$ and the corresponding estimator $Q_{\hat{\mathcal{S}}}$ (resp., $\tilde{Q}_{\hat{\mathcal{S}}}$) as in classical model selection procedures, this estimator is a mixture of all the Laplace (resp., adaptive Laplace) estimators.

Let us first describe the construction of the progressive mixture estimator $Q_\pi^N(Y_N \mid X_N;\, Z_1^{N-1})$ which aggregates the Laplace estimators defined in Section III.

An integer $K \in [1, N-1]$ is first chosen and the observations Z_1^{N-1} are split into an *estimation set* Z_1^K and a *validation set* Z_{K+1}^{N-1}.

For each model $\mathcal{S} \in \mathcal{C}_D$, the estimation set is mapped by the Laplace estimator to a conditional distribution $Q_\mathcal{S}^{K+1}(Y \mid X)$ defined by

$$\forall \mathcal{S} \in \mathcal{C}_D, \qquad Q_\mathcal{S}^{K+1}(Y \mid X) = Q_\mathcal{S}^{K+1}(Y \mid X;\, Z_1^K) \tag{12}$$

where the latter is defined by (4).

For any $n \in [0, N-K-1]$ let now $Q_\pi^{(n)}(Y \mid X)$ be the conditional distribution obtained as a Bayesian mixture of the primary estimators $\{Q_{\mathcal{S}}^{K+1}(Y\,|\,X)\}_{\mathcal{S}\in\mathcal{C}_D}$ with the prior distribution π on $\mathcal{C}_D$ and the observations Z_{K+1}^{K+n}, i.e.,

$$Q_\pi^{(n)}(Y\,|\,X) = \frac{\sum_{\mathcal{S}\in\mathcal{C}_D} \pi(\mathcal{S}) \left(\prod_{i=K+1}^{K+n} Q_{\mathcal{S}}^{K+1}(Y_i\,|\,X_i)\right) Q_{\mathcal{S}}^{K+1}(Y\,|\,X)}{\sum_{\mathcal{S}\in\mathcal{C}_D} \pi(\mathcal{S}) \left(\prod_{i=K+1}^{K+n} Q_{\mathcal{S}}^{K+1}(Y_i\,|\,X_i)\right)}.$$

The progressive mixture estimator Q_π^N is then a Cesaro mean of these Bayesian estimators trained on subsamples of growing sizes, i.e,

$$Q_\pi^N\left(Y_N\,|\,X_N;\,Z_1^{N-1}\right) \overset{def}{=} \frac{1}{N-K}\sum_{n=0}^{N-K-1} Q_\pi^{(n)}(Y_N\,|\,X_N).$$

The idea of building a progressive estimator has been proposed independently by Barron [21] [22] and Catoni [15] who proved the following property.

Theorem 3 (Catoni, [15]):

$$R_{P_N}\left(Q_\pi^N\right) \le \inf_{\mathcal{S}\in\mathcal{C}_D}\left\{R_{P_N}\left(Q_{\mathcal{S}}^{K+1}\right) + \frac{1}{N-K}\log\frac{1}{\pi(\mathcal{S})}\right\}. \tag{13}$$

The construction of the progressive mixture estimator $\tilde{Q}_\pi^N$ which aggregates the adaptive Laplace estimators is exactly the same as the construction of Q_π^N except that each Q should be replaced by $\tilde{Q}$.

We can now evaluate the risks of Q_π^N and $\tilde{Q}_\pi^N$.

Theorem 4: Let Q_π^N (resp., $\tilde{Q}_\pi^N$) denote the progressive mixture estimator based on the family of Laplace estimators $\{Q_{\mathcal{S}}^{K+1}\}_{\mathcal{S}\in\mathcal{C}_D}$ (resp., adaptive Laplace estimators $\{\tilde{Q}_{\mathcal{S}}^{K+1}\}_{\mathcal{S}\in\mathcal{C}_D}$) and on the prior π defined in Section IV, with the size of the training being set to

$$K = \left[\frac{N\sqrt{a-1} - \sqrt{\log(a)+1}}{\sqrt{a-1}+\sqrt{\log(a)+1}}\right]$$

where [.] denotes greatest integer.

For any exchangeable distribution P_N on $(\mathcal{X}\times\mathcal{Y})^N$, the risk of Q_π^N satisfies

$$R_{P_N}\left(Q_\pi^N\right) \le \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}}\left\{R_{P_N}(P_{\mathcal{S},\theta}) + \frac{|\mathcal{S}|C_N}{N+1}\right\}$$

with

$$C_N = \left(\sqrt{1+\log(a)}+\sqrt{a-1}\right)^2\left(1+\frac{1}{N-2}\right).$$

Let γ_{K+1} be defined in as in Theorem 2. The risk of $\tilde{Q}_\pi^N$ satisfies

$$R_{P_N}\left(\tilde{Q}_\pi^N\right) \le \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{|\mathcal{S}|}}\left\{R_{P_N}(P_{\mathcal{S},\theta}) + \frac{\sum_{s\in\mathcal{S}}\delta_N(s)}{N+1}\right\}$$

with

$$\delta_N(s) = \left(\sqrt{\gamma_{K+1}(s)}+\sqrt{\log(a)+1}\right)^2 + \sqrt{\frac{\log(a)+1}{a-1}}\left(\sqrt{a-1}-\sqrt{\gamma_{K+1}(s)}\right)^2 + \frac{\gamma_{K+1}(s)}{(a-1)(N-1)}\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)^2.$$

Remark 4: The definition of K shows that the larger the alphabet, the longer it takes to train the Laplace estimators compared with the time it takes to aggregate them with the progressive mixture estimator (i.e., K/N increases with a, with limit 1 as a tends to infinity). For a large a, the risk bound associated with any model $\mathcal{S}$ is very close to $|\mathcal{S}|(a-1)/N$, which is the risk of the Laplace estimator for this model.

Remark 5: The term $\delta_N(s)$ is the sum of three terms. The first is the term one would expect if $\gamma_{K+1}(s)$ were known *a priori* so that the size of the training set K could be better adjusted. The second is the loss due to the fact that $\gamma_{K+1}(s)$ is not known *a priori* and we decided to take for K the value corresponding to the best split for Q_π^N instead of $\tilde{Q}_\pi^N$. The third term vanishes to zero and is the loss due to the fact that K has to be an integer.

Proof of Theorem 4: Using Theorem 1, Theorem 3, and Lemma 1 we can write

$$\begin{aligned}R_{P_N}\left(Q_\pi^N\right) &\le \inf_{\mathcal{S}\in\mathcal{C}_D}\left\{R_{P_N}\left(Q_{\mathcal{S}}^{K+1}\right) + |\mathcal{S}|\frac{\log(a)+1}{N-K}\right\}\\ &\le \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}}\left\{R_{P_N}(P_{\mathcal{S},\theta}) + |\mathcal{S}|\left(\frac{a-1}{K+1}+\frac{\log(a)+1}{N-K}\right)\right\}.\end{aligned}$$

The function

$$x\mapsto f(x) = \frac{a-1}{x+1}+\frac{\log(a)+1}{N-x}$$

is minimum on $(0,\,N)$ at the point

$$x^* = \frac{N\sqrt{a-1}-\sqrt{\log(a)+1}}{\sqrt{a-1}+\sqrt{\log(a)+1}}.$$

K must be an integer so a good candidate to ensure a risk as small as possible for Q_π^N is $K=[x^*]$ for which we can compute

$$\begin{aligned}f(K) &\le \frac{a-1}{x^*}+\frac{\log(a)+1}{N-x^*}\\ &\le \frac{(a-1)\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)}{N\sqrt{a-1}-\sqrt{\log(a)+1}} + \frac{(\log(a)+1)\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)}{(N+1)\sqrt{\log(a)+1}}\\ &\le \frac{\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)^2}{N+1} \times\left(1+\frac{1}{N-\sqrt{\frac{\log(a)+1}{a-1}}}\right).\end{aligned}$$

The upper bound concerning the Laplace estimator in Theorem 4 follows by observing that $a \geq 2$ and, therefore,

$$\sqrt{\frac{\log(a)+1}{a-1}} \leq \sqrt{1+\log(2)} \leq 2.$$

For the second part of the theorem concerning the aggregation of adaptive Laplace estimators we follow the same computation except that by Theorem 2 we get

$$R_{P_N}\left(\tilde{Q}_\pi^N\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D,\, \theta\in\Sigma^{\mathcal{S}}} \left\{ R_{P_N}(P_{\mathcal{S},\theta}) + \sum_{s\in\mathcal{S}} g_s(K) \right\}$$

where g_s is defined by

$$g_s(x) = \frac{\gamma_{K+1}(s)}{x+1} + \frac{\log(a)+1}{N-x}.$$

We now just need an upper bound for $g_s(K)$ where K is chosen as in Theorem 4, which is given by

$$\begin{aligned}
g_s(K) &\leq \frac{\gamma_{K+1}(s)}{x^*} + \frac{\log(a)+1}{N-x^*} \\
&\leq \frac{\gamma_{K+1}(s)\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)}{N\sqrt{a-1}-\sqrt{\log(a)+1}} \\
&\quad + \frac{(\log(a)+1)\left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)}{(N+1)\sqrt{\log(a)+1}} \\
&\leq \frac{\sqrt{a-1}+\sqrt{\log(a)+1}}{N+1} \\
&\quad \times \left[\frac{\gamma_{K+1}(s)}{\sqrt{a-1}} \left(1 + \frac{\sqrt{a-1}+\sqrt{\log(a)+1}}{N\sqrt{a-1}-\sqrt{\log(a)+1}}\right) \right. \\
&\qquad \left. + \sqrt{\log(a)+1} \right] \\
&\leq \frac{1}{N+1} \left\{ \left(\sqrt{\gamma_{K+1}(s)} + \sqrt{\log(a)+1}\right)^2 \right. \\
&\qquad + \sqrt{\frac{\log(a)+1}{a-1}} \left(\sqrt{a-1} - \sqrt{\gamma_{K+1}(s)}\right)^2 \\
&\qquad \times \frac{\gamma_{K+1}(s)}{(a-1)(N-2)} \\
&\qquad \left. \times \left(\sqrt{a-1}+\sqrt{\log(a)+1}\right)^2 \right\}. \qquad \square
\end{aligned}$$

VI. Aggregation Using a Gibbs Estimator

In this section, we present a second aggregation method based on the Gibbs estimator, introduced by Catoni in [18]. Let us first describe this estimator $G_{\pi,\beta}^N(Y_N \,|\, Y_N,\, Z_1^{N-1})$ to aggregate Laplace estimators.

As for the progressive mixture estimator presented in Section V, the observations Z_1^{N-1} are split into two sets Z_1^K and Z_{K+1}^{N-1} where K is an integer in $[1,\, N-1]$, and the observation set Z_1^K is used to define the set of primary estimators $\{Q_{\mathcal{S}}^{K+1}(Y \,|\, X)\}_{\mathcal{S}\in\mathcal{C}_D}$ using the Laplace estimators as in (12).

The Gibbs estimator at inverse temperature $\beta \in \mathbb{R}_+$ using the prior π on $\mathcal{C}_D$ is now the following conditional distribution:

$$G_{\pi,\beta}^N\left(Y_N \,|\, X_N;\, Z_1^{N-1}\right) \overset{\text{def}}{=} \frac{\sum\limits_{\mathcal{S}\in\mathcal{C}_D} \pi(\mathcal{S}) \left(\prod\limits_{n=K+1}^{N-1} Q_{\mathcal{S}}^{K+1}(Y_n \,|\, X_n) \right)^{\beta} Q_{\mathcal{S}}^{K+1}(Y_N \,|\, X_N)}{\sum\limits_{\mathcal{S}\in\mathcal{C}_D} \pi(\mathcal{S}) \left(\prod\limits_{n=K+1}^{N-1} Q_{\mathcal{S}}^{K+1}(Y_n \,|\, X_n) \right)^{\beta}}. \tag{14}$$

This definition shows that the Gibbs estimator can be considered as a "thermalized" version of both the Bayesian ($\beta = 1$) and the maximum-likelihood ($\beta = +\infty$) estimators. Catoni studied in [18] this estimator in the high-temperature region $\beta < 1$ which is equivalent to a deliberate underestimation of the sample size: to compute the Gibbs estimator, the empirical distribution of $N-K-1$ observations is plugged into the Bayes estimator for a sample of size $\beta(N-K-1)$. The reason to consider high temperatures is that the estimator gains stability with respect to the empirical process when β decreases (at the limit, it is constant when $\beta = 0$). This property is used by Catoni to prove a general upper bound for its risk in the spirit of (1), which takes the following form in the particular case when the primary estimators are log-bounded.

Theorem 5 (Catoni, [18]): Let $\chi > 0$ such that

$$\forall \mathcal{S} \in \mathcal{C}_D, \forall (z_1^K,\, z) \in (\mathcal{X} \times \mathcal{Y})^{K+1}, \qquad -\chi \leq \log Q_{\mathcal{S}}^{K+1}\left(y \,|\, x,\, z_1^K\right) \leq 0.$$

If β satisfies

$$\beta \leq \frac{1}{\chi-1} \left(\sqrt{1-(\chi-1)\left(2-\frac{\log\chi}{\chi}\right)\frac{\log\chi}{\chi}} - 1 \right)$$

then the Gibbs estimator $G_{\pi,\beta}^N$ defined by (14) satisfies

$$R_{P_N}(G_{\pi,\beta}^N) \leq \inf_{\mathcal{S}\in\mathcal{C}_D} \left\{ R_{P_N}\left(Q_{\mathcal{S}}^{K+1}\right) + \frac{1}{\beta(N-K)} \log \frac{1}{\pi(\mathcal{S})} \right\}. \tag{15}$$

The definition of the Gibbs estimator

$$\tilde{G}_{\pi,\beta}^N\left(Y_N \,|\, X_N;\, Z_1^{N-1}\right)$$

to aggregate adaptive Laplace estimators follows exactly the same construction by replacing every Q by $\tilde{Q}$.

We can now evaluate the risk of $G_{\pi,\beta}^N$ and $\tilde{G}_{\pi,\beta}^N$.

Theorem 6: Let

$$\begin{cases} \chi_N = \log(N+a) \\ \tilde{\chi}_N = \log(N+a) + \log(a). \end{cases}$$

Let

$$\beta_N = \frac{1}{\chi_N - 1} \times \left(\sqrt{1 - (\chi_N - 1)\left(2 - \frac{\log \chi_N}{\chi_N}\right)\frac{\log \chi_N}{\chi_N}} - 1 \right) \underset{N \to +\infty}{\sim} \frac{\sqrt{2 \log \log N}}{\log N},$$

and let $\tilde{\beta}_N$ be deduced from $\tilde{\chi}_N$ as β_N is deduced from χ_N.

Let $G^N_{\pi,\,\beta}$ (resp., $\tilde{G}^N_{\pi,\,\beta}$) denote the Gibbs estimator at inverse temperature β_N (resp., $\tilde{\beta}_N$) based on the family of Laplace estimators $\{Q^{K+1}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$ (resp., adaptive Laplace estimators $\{\tilde{Q}^{K+1}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$) and on the prior π defined in Section IV, with the size of the training being set to

$$K = \left[\frac{N\sqrt{a-1} - \sqrt{\beta_N^{-1}(\log(a)+1)}}{\sqrt{a-1} + \sqrt{\beta_N^{-1}(\log(a)+1)}} \right]$$

where [.] denotes greatest integer (resp., to $\tilde{K}$ defined like K with β_N replaced by $\tilde{\beta}_N$).

For any exchangeable distribution P_N on $(\mathcal{X}\times\mathcal{Y})^N$ the risk of $G^N_{\pi,\,\beta}$ satisfies

$$R_{P_N}\left(G^N_{\pi,\beta}\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}} \left\{ R_{P_N}(P_{\mathcal{S},\theta}) + \frac{|\mathcal{S}|C_N}{N+1} \right\}$$

with

$$C_N = \left(\sqrt{(1+\log(a))\,\beta_N^{-1}} + \sqrt{a-1} \right)^2 \left(1 + \frac{1}{N-2}\right).$$

Let γ_{K+1} be defined in as in Theorem 2. The risk of $\tilde{G}^N_{\pi,\,\beta}$ satisfies

$$R_{P_N}\left(\tilde{G}^N_{\pi,\tilde{\beta}}\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D,\theta\in\Sigma^{\mathcal{S}}} \left\{ R_{P_N}(P_{\mathcal{S},\theta}) + \frac{\sum_{s\in\mathcal{S}} \delta_N(s)}{N+1} \right\}$$

with

$$\begin{aligned} \delta_N(s) = &\left(\sqrt{\gamma_{K+1}(s)} + \sqrt{(\log(a)+1)\,\tilde{\beta}_N^{-1}} \right)^2 \\ &+ \sqrt{\frac{(\log(a)+1)\,\tilde{\beta}_N^{-1}}{a-1}} \left(\sqrt{a-1} - \sqrt{\gamma_{K+1}(s)} \right)^2 \\ &+ \frac{\gamma_{K+1}(s)}{(a-1)\,(N-1)} \\ &\times \left(\sqrt{a-1} + \sqrt{(\log(a)+1)\,\tilde{\beta}_N^{-1}} \right)^2. \end{aligned}$$

Remark 6: Asymptotically, the upper bound on the risks of the Gibbs estimators provided by Theorem 6 appear to be worse than the risks of the corresponding progressive mixture estimators given by Theorem 4 because of the factor $(\beta_N)^{-1}$. This is due to the fact that the inverse temperature has to be taken smaller and smaller as N increases in order to prove that (15) holds. However, the conditions imposed on β which involve a uniform bound on the likelihood of the primary estimators might be very conservative in the particular problem we consider. Therefore, larger values of β might also ensure the validity of (15), and the actual performance of this estimator is probably better than the one proven in Theorem 6 (it is reasonable to think from the computations in [18] that $\beta = 1/2$ will work in many cases).

Remark 7: Even though the risk of the Gibbs estimator is worse than the risk of the progressive mixture estimator one might prefer to implement the former because it only involves the computation of one mixture, while the latter involves the computation of $N-K$ Bayesian mixtures which are then averaged.

Proof of Theorem 6: The family of Laplace estimators $\{Q^{K+1}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$ is uniformly bounded by

$$\begin{aligned} &\forall\, z_1^{K+1} \in (\mathcal{X}\times\mathcal{Y})^{K+1}, \forall \mathcal{S}\in\mathcal{C}_D, \\ &\qquad 0 \geq \log Q^{K+1}_{\mathcal{S}}\left(y_{K+1}\,|\,x_{K+1};\, z_1^K\right) \\ &\qquad = \log \frac{\mu_K(s_{\mathcal{S}}(x_{K+1}),\, y_K) + 1}{\nu_K(s_{\mathcal{S}}(x_{K+1})) + a} \\ &\qquad \geq -\log(K+1+a) \\ &\qquad \geq -\log(N+a). \end{aligned}$$

Similarly, the family of adaptive Laplace estimator $\{\tilde{Q}^{K+1}_{\mathcal{S}}\}_{\mathcal{S}\in\mathcal{C}_D}$ is uniformly bounded by

$$\begin{aligned} &\forall\, z_1^{K+1} \in (\mathcal{X}\times\mathcal{Y})^{K+1}, \forall \mathcal{S}\in\mathcal{C}_D, \\ &\qquad 0 \geq \log \tilde{Q}^{K+1}_{\mathcal{S}}\left(y_{K+1}\,|\,x_{K+1};\, z_1^K\right) \\ &\qquad \geq -\log(N+a) - \log(a). \end{aligned}$$

We can, therefore, apply Theorem 5 with χ_N (resp., $\tilde{\chi}_N$) and β_N (resp., $\tilde{\beta}_N$) as defined in Theorem 6 to get

$$R_{P_N}\left(G^N_{\pi,\beta}\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D} \left\{ R_{P_N}\left(Q^{K+1}_{\mathcal{S}}\right) + \frac{1}{\beta_N(N-K)} \log\frac{1}{\pi(\mathcal{S})} \right\}$$

and

$$R_{P_N}\left(\tilde{G}^N_{\pi,\tilde{\beta}}\right) \leq \inf_{\mathcal{S}\in\mathcal{C}_D} \left\{ R_{P_N}\left(Q^{K+1}_{\mathcal{S}}\right) + \frac{1}{\tilde{\beta}_N(N-K)} \log\frac{1}{\pi(\mathcal{S})} \right\}$$

Using these two inequalities instead of (13) the proof of Theorem 6 now follows exactly the proof of Theorem 4. □

VII. Data-Dependent Prior on the Trees

Theorem 1 provides two bounds for the risk of the Laplace estimator on a given tree: the first depends on the design distribution, i.e., the distribution of X_1^N, and reflects the property of adaptiveness of the estimator, while the second does not depend on the design law, and is therefore weaker. The aggregation of these estimators described in Sections V and VI are also distribution-independent because the model risk is chosen *a priori*.

In this section, we present a modification which can be applied to any of the four estimators studied in Sections V and VI. It consists in replacing the prior distribution π on the set of trees $\mathcal{C}_D$ by a *data-dependent* prior $\overline{\pi}$ to aggregate the primary estimators in order to get a better upper bound on the risk, which depends on the design distribution. This modification should be especially useful when the design distribution $P_N(X_1^N)$ is concentrated on a small subspace of $\mathcal{A}^D$, which is, for instance, the case in natural language modeling (see Section IX).

For clarity, we just show the construction of the estimator $Q_{\overline{\pi}}^N$ which is the modification of Q_{π}^N, the progressive mixture estimator which aggregates Laplace primary estimators and is defined in Section V. Let us, therefore, formally define the density $Q_{\overline{\pi}}^N(y_N \,|\, x_1^N;\, y_1^{N-1})$ for any $z_1^N \in (\mathcal{X} \times \mathcal{Y})^N$.

Let $\mathcal{T}(x_1^N)$ denote the tree (in the sense of Section II-B) whose vertices are the suffixes of the x_i's, i.e.,

$$\mathcal{T}(x_1^N) = \left\{(x_i)_{-l}^0\colon (i, l) \in [1, N] \times [0, D]\right\}$$

and let $\overline{\mathcal{T}}(x_1^N)$ be the graph obtained by removing from $\mathcal{T}(x_1^N)$ the vertices with only one child and merging the two edges starting from a removed node (i.e., the edge toward its parent and the edge toward its single child). A *subtree* of the graph $\overline{\mathcal{T}}(x_1^N)$ is by definition any connex subgraph which contains the root λ as a vertex.

Example 2: Fig. 2(a) shows the graph $\overline{\mathcal{T}}(x_1^N)$ when $D = 4$ and the observation is $x_1^3 =$ (caba,aacc,cbcc). In that case, the set of vertices of $\overline{\mathcal{T}}(x_1^3)$ is $\{\lambda$,caba,cc,aacc,cbcc$\}$. Two possible subtrees of $\overline{\mathcal{T}}(x_1^3)$ are shown on the right-hand sides of Fig. 2(b) and (c), with respective sets of vertices $\{\lambda$,caba,cc$\}$ and $\{\lambda$,cc,cbcc$\}$.

Let $\overline{\mathcal{C}}(x_1^N)$ be the set of subtrees of $\overline{\mathcal{T}}(x_1^N)$. For any $\overline{\mathcal{S}} \in \overline{\mathcal{C}}(x_1^N)$ the suffix functional $s_{\overline{\mathcal{S}}}$ is defined in the same way as when $\mathcal{S}$ is a classical tree (see Section II-B). For any $\overline{\theta} \in \Sigma^{\overline{\mathcal{S}}}$ let $P_{\overline{\mathcal{S}}, \overline{\theta}}$ denote the conditional probability distribution

$$\forall\, (x, y) \in \mathcal{X} \times \mathcal{Y}, \qquad P_{\overline{\mathcal{S}}, \overline{\theta}}(y \,|\, x) = \overline{\theta}_{s_{\overline{\mathcal{S}}}(x)}(y).$$

The counters

$$(\nu_n(s))_{s \in \overline{\mathcal{S}}} \quad \text{and} \quad (\mu_n(s, y))_{(s, y) \in \overline{\mathcal{S}} \times \mathcal{Y}}$$

are defined as before by (3). Therefore, the distribution $Q_{\overline{\mathcal{S}}}^n(y_n \,|\, x_n,\, z_1^{n-1})$ can also be defined as before by (4).

Let $\overline{\pi}_{(x_1^N)}$ be the distribution on $\overline{\mathcal{C}}(x_1^N)$ defined by

$$\forall \overline{\mathcal{S}} \in \overline{\mathcal{C}}(x_1^N), \qquad \overline{\pi}_{(x_1^N)}(\overline{\mathcal{S}}) = c^{|\overline{\mathcal{S}}|}$$

where c is the real number which satisfies

$$\sum_{\overline{\mathcal{S}} \in \overline{\mathcal{C}}(x_1^N)} c^{|\overline{\mathcal{S}}|} = 1.$$

Using this data-dependent prior $\overline{\pi}_{(x_1^N)}$ instead of the data-independent prior π in the definition of Q_{π}^N (see Section IV) we finally obtain a modified estimator $Q_{\overline{\pi}}^N$.

For any tree $\mathcal{S}$ in $\mathcal{C}_D$ recall that $v_N(\mathcal{S})$ denotes the set of visited nodes of $\mathcal{S}$, i.e.,

$$v_N(\mathcal{S}) = \{s \in \mathcal{S}\colon \mu_N(s) > 0\}$$

and let $\overline{v(\mathcal{S})}$ be the smallest subtree $\overline{\mathcal{S}}$ of $\overline{\mathcal{T}}(x_1^N)$ such that for any $s \in v_N(\mathcal{S})$, there is an $s' \in \overline{\mathcal{S}}$ such that s is a suffix of s'.

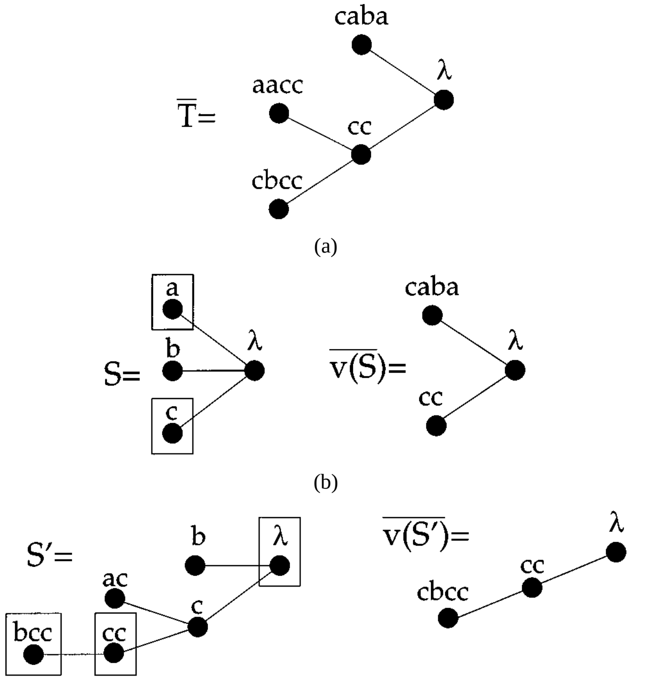

Fig. 2. (a) $\overline{\mathcal{T}}(x_1^3)$ for Example 2. (b) A tree $\mathcal{S}$, squares on its visited nodes, and corresponding subtree of $\overline{\mathcal{T}}(x_1^3)$ (see Example 3). (c) Same as Fig. 2(b) with a different tree $\mathcal{S}$ (see Example 3).

Example 3: As in Example 2, suppose that $D = 4$, $N = 3$, and $x_1^3 =$ (caba,aacc,cbcc). The left-hand sides of Fig. 2(b) and (c) show two trees $\mathcal{S}$ and $\mathcal{S}'$ in $\mathcal{C}_D$. The squares around nodes on $\mathcal{S}$ and $\mathcal{S}'$ indicate the nodes which belong to $v_3(\mathcal{S})$ and $v_3(\mathcal{S}')$. The right-hand sides of the same figures show the corresponding $\overline{v(\mathcal{S})}$ and $\overline{v(\mathcal{S}')}$

We can now give an upper bound on the risk of the estimator $Q_{\overline{\pi}}^N$:

Theorem 7: Let the size of the training set be the same as in Theorem 4. For any exchangeable distribution P_N on $(\mathcal{X} \times \mathcal{Y})^N$, the estimator $Q_{\overline{\pi}}^N$ using the data-dependent prior $\overline{\pi}$ satisfies

$$R_{P_N}\left(Q_{\overline{\pi}}^N\right) \le \inf_{\mathcal{S} \in \mathcal{C}_D, \theta \in \Sigma^{|\mathcal{S}|}} \left\{R_{P_N}(P_{\mathcal{S},\theta}) + \boldsymbol{E}_{P_N}\left(\left|\overline{v(\mathcal{S})}\right|\right) \frac{C_N}{N+1}\right\} \quad (16)$$

with

$$C_N = \left(\sqrt{1 + \log(a)} + \sqrt{a-1}\right)^2 \left(1 + \frac{1}{N-2}\right).$$

Remark 8: For any $\mathcal{S} \in \mathcal{C}_D$, $|\overline{v(\mathcal{S})}|$ is always smaller than $|\mathcal{S}|$. The upper bound in Theorem 7 is therefore smaller than the corresponding upper bound in Theorem 4. The difference can be large in cases when $P_N(X_1^N)$ is concentrated on a small subset of $\mathcal{A}^D$, because in that case $\overline{\mathcal{T}}(X_1^N)$ is a small subtree of $\bigcup_{i=0}^D \mathcal{A}^i$ with high probability.

Remark 9: The Laplace estimator for a given tree requires no modification because its risk is already bounded in terms of the number of visited nodes (see Theorem 1). Therefore, only the prior π needs to be modified to become data-dependent.

Remark 10: Every tree $\mathcal{S} \in \mathcal{C}_D$ splits the data x_1^N into $|v(\mathcal{S})| = |v(\overline{v(\mathcal{S})})|$ clusters. The number of different separation of the data x_1^N by trees in $\mathcal{C}_D$ is, therefore,

$$\mathcal{N}(x_1^N) = \left|\left\{v\left(\overline{v(\mathcal{S})}\right) : \mathcal{S} \in \mathcal{C}_D\right\}\right|$$

which is equal to $|\overline{\mathcal{C}}(x_1^N)|$ up to the number of trees with unvisited nodes. If we had chosen for $\overline{\pi}$ a uniform prior on $\overline{\mathcal{C}}(x_1^N)$, the model risk would have been of the order of $\boldsymbol{E} \log \mathcal{N}(X_1^N)$. The idea of computing an upper bound involving such a model risk instead of a model risk of order $\log |\mathcal{C}_D|$ (resulting from a uniform prior) is classical in statistical learning theory (see [23]), where the numbers $\mathcal{N}(x_1^N)$ and $\boldsymbol{E} \log(\mathcal{N}(X_1^N))$ are, respectively, known as the *shatter coefficient* and the *annealed entropy*.

Proof of Theorem 7: The random tree $\overline{\mathcal{T}}(X_1^N)$ is invariant under permutation of the indexes $[1, N]$. As a result, for any such tree $\overline{\mathcal{T}}$, the distribution $P_N(Z_1^N \,|\, \overline{\mathcal{T}}(X_1^N) = \overline{\mathcal{T}})$ is exchangeable. In the event that $\{\overline{\mathcal{T}}(X_1^N) = \overline{\mathcal{T}}\}$, the prior $\overline{\pi}$ is independent of the data and therefore Theorem 4 can be applied. As a result, the following holds for any $\overline{\mathcal{S}} \in \overline{\mathcal{C}}(X_1^N)$ and $\overline{\theta} \in \Sigma^{\overline{\mathcal{S}}}$:

$$\begin{aligned} &\boldsymbol{E}_{P_N(dZ_1^N \,|\, \overline{\mathcal{T}}(X_1^N) = \overline{\mathcal{T}})} \log \frac{1}{Q_{\overline{\pi}}^N\left(Y_N \,|\, X_N;\, Z_1^{N-1}\right)} \\ &\quad \le \boldsymbol{E}_{P_N(dZ_1^N \,|\, \overline{\mathcal{T}}(X_1^N) = \overline{\mathcal{T}})} \log \frac{1}{P_{\overline{\mathcal{S}},\overline{\theta}}(Y_N \,|\, X_N)} + |\overline{\mathcal{S}}| \frac{C_N}{N+1} \end{aligned} \tag{17}$$

where C_N is defined in Theorem 7.

For any $\mathcal{S} \in \mathcal{C}^D$ and $\theta \in \Sigma^{\mathcal{S}}$, let $\overline{\theta} \in \Sigma^{\overline{v(\mathcal{S})}}$ be the parameter defined by

$$\forall\, s \in \overline{v(\mathcal{S})}, \qquad \overline{\theta}_s = \theta_{s'}$$

where s' is the longest visited suffix of s in $\mathcal{S}$. For any $i \in [1, N]$ this definition leads to

$$\begin{aligned} P_{\overline{v(\mathcal{S})},\overline{\theta}}(y_i \,|\, x_i) &= \overline{\theta}_{s_{\overline{v(\mathcal{S})}}(x_i)}(y_i) \\ &= \theta_{s_i'}(y_i) \end{aligned}$$

where s_i' is the longest visited suffix of $s_{\overline{v(\mathcal{S})}}(x_i)$ in $\mathcal{S}$. But $s_{\overline{v(\mathcal{S})}}(x_i)$ is by definition a suffix of x_i thus s_i' must also be a suffix of x_i. The largest suffix of x_i in $\mathcal{S}$ is $s_{\mathcal{S}}(x_i)$, which is by definition a suffix of $s_{\overline{v(\mathcal{S})}}(x_i)$. This shows that $s_i' = s_{\mathcal{S}}(x_i)$, and, therefore,

$$\forall\, i \in [1, N], \qquad P_{\mathcal{S},\theta}(y_i \,|\, x_i) = P_{\overline{v(\mathcal{S})},\overline{\theta}}(y_i \,|\, x_i).$$

The parameter $\overline{\theta}$ only depends on Z_1^N through $\overline{\mathcal{T}}$ and, therefore, we can integrate this equality to get

$$\begin{aligned} &\boldsymbol{E}_{P_N(dZ_1^N \,|\, \overline{\mathcal{T}}(X_1^N) = \overline{\mathcal{T}})} \log \frac{1}{P_{\mathcal{S},\theta}(Y_N \,|\, X_N)} \\ &\quad = \boldsymbol{E}_{P_N(dZ_1^N \,|\, \overline{\mathcal{T}}(X_1^N) = \overline{\mathcal{T}})} \log \frac{1}{P_{\overline{v(\mathcal{S})},\overline{\theta}}(Y_N \,|\, X_N)}. \end{aligned}$$

From (17), we deduce that for any $\overline{\mathcal{T}}$, $\mathcal{S}$, and θ the following holds:

$$\begin{aligned} &\boldsymbol{E}_{P_N(dZ_1^N \,|\, \overline{\mathcal{T}}(X_1^N) = \overline{\mathcal{T}})} \log \frac{1}{Q_{\overline{\pi}}^N(Y_N \,|\, X_N;\, Z_1^{N-1})} \\ &\le \boldsymbol{E}_{P_N(dZ_1^N \,|\, \overline{\mathcal{T}}(X_1^N) = \overline{\mathcal{T}})} \log \frac{1}{P_{\mathcal{S},\theta}(Y_N \,|\, X_N)} + |\overline{v(\mathcal{S})}| \frac{C_N}{N+1}. \end{aligned}$$

Taking the expectation of this inequality with respect to P_N yields the upper bound in Theorem 16. □

VIII. Implementation for the Aggregation Using a Gibbs Estimator

In this section, we show how the estimator

$$G_{\pi,\,\beta}^N\left(Y_N \,|\, X_N;\, Z_1^{N-1}\right)$$

using the Gibbs estimator to aggregate Laplace estimators (see Section VI) can be computed using a recursive algorithm in the spirit of the context tree weighting algorithm [7]. The construction we present can be adapted to the other estimators studied in this paper.

A. Exact Computation

Let $\mathcal{T}_D = \bigcup_{i=0}^D \mathcal{A}^i$ be the *context tree* of depth D, and for every $z_1^N \in (\mathcal{X} \times \mathcal{Y})^N$ let the following counters be attached to the nodes of the context tree, i.e., $\forall\,(s,\, y) \in \mathcal{T}_D \times \mathcal{Y}$:

$$\begin{cases} \mu_T(s,\, y) &= \sum_{i=1}^{K} \mathbf{1}(s \text{ is a suffix of } x_i \text{ and } y_i = y), \\ \mu_V(s,\, y) &= \sum_{i=K+1}^{N-1} \mathbf{1}(s \text{ is a suffix of } x_i \text{ and } y_i = y), \\ \mu_*(s,\, y) &= \mathbf{1}(s \text{ is a suffix of } x_N \text{ and } y_N = y), \\ \nu_T(s) &= \sum_{y \in \mathcal{Y}} \mu_T(s,\, y). \end{cases}$$

The subscripts T and V refer to the training set and the validation set, respectively. Using these counters we can define the following functions attached to each node $s \in \mathcal{T}_D$, and defined for any subset $\mathcal{N} \subset \mathcal{Y}$ and $\xi \in \{0,\, 1\}$:

$$w_{\mathcal{N}}^{(\xi)}(s) \overset{def}{=} c_D \prod_{y \in \mathcal{Y}} \left(\frac{\mu_T(s,\, y) - \sum_{i \in \mathcal{N}} \mu_T(is,\, y) + 1}{\nu_T(s) - \sum_{i \in \mathcal{N}} \nu_T(is) + a} \right)^{\phi(s,y)}$$

where

$$\phi(s,\, y) = \beta \left(\mu_V(s,\, y) - \sum_{i \in \mathcal{N}} \mu_V(is,\, y) \right) + \xi \mu_*(s,\, y).$$

For any $\mathcal{S} \in \mathcal{C}_D$ and $s \in \mathcal{S}$ let

$$\mathcal{N}_{\mathcal{S}}(s) = \{i \in \mathcal{A}\colon is \in \mathcal{S}\}.$$

Let now $\gamma^{(\xi)}$ be defined recursively on $\mathcal{T}_D$ for $\xi = \{0,\, 1\}$ by the formula

$$\begin{cases} \gamma^{(\xi)}(s) = w_{\emptyset}^{(\xi)}(s), & \text{if } l(s) = D \\ \gamma^{(\xi)}(s) = \sum_{\mathcal{N} \subset \mathcal{Y}} w_{\mathcal{N}}^{(\xi)} \prod_{i \in \mathcal{Y} \setminus \mathcal{N}} \gamma^{(\xi)}(is), & \text{otherwise.} \end{cases} \tag{18}$$

The following lemma shows that $\gamma^{(\xi)}(s)$ can be seen as a tensorization of a sum over all subtrees with root s.

Lemma 2:

$\forall (s,\, \xi) \in \mathcal{T}_D \times \{0, 1\}$,

$$\gamma^{(\xi)}(s) = \sum_{\mathcal{S}\in\mathcal{C}_{D-d}} \left(\prod_{s'\in\mathcal{S}} w_{\mathcal{N}_\mathcal{S}(s')}(s's)\right)$$

and the following result gives an effective way of computing the estimator $G^N_{\pi,\,\beta}\,(Y_N\,|\,X_N;\, Z_1^{N-1})$.

Proposition 1:

$$\forall\, z_1^N \in (\mathcal{X}\times\mathcal{Y})^N, \qquad G^N_{\pi,\,\beta}\left(y_N\,|\,x_N;\, z_1^{N-1}\right) = \frac{\gamma^{(1)}(\lambda)}{\gamma^{(0)}(\lambda)}.$$

Proof of Lemma 2: We prove the result by backward induction on $l(s)$. The property is obvious for $l(s) = D$ by definition of $\gamma^{(\xi)}(s)$ in that case. Suppose it is true for any $s' \in \mathcal{T}_D$ such that $l(s') = d + 1$, and let a string $s \in \mathcal{T}_D$ of length $l(s) = d$. Then we get

$$\begin{aligned}\gamma^{(\xi)}(s) &= \sum_{\mathcal{N}\subset\mathcal{Y}} w^{(\xi)}_{\mathcal{N}} \prod_{i\in\mathcal{Y}\setminus\mathcal{N}} \gamma^{(\xi)}(is)\\ &= \sum_{\mathcal{N}\subset\mathcal{Y}} w^{(\xi)}_{\mathcal{N}} \prod_{i\in\mathcal{Y}\setminus\mathcal{N}} \left[\sum_{\mathcal{S}\in\mathcal{C}_{D-d-1}} \left(\prod_{s'\in\mathcal{S}} w_{\mathcal{N}_\mathcal{S}(s')}(s'is)\right)\right]\\ &= \sum_{\mathcal{S}\in\mathcal{C}_{D-d}} \left(\prod_{s'\in\mathcal{S}} w_{\mathcal{N}_\mathcal{S}(s')}(s's)\right). \qquad \square\end{aligned}$$

Proof of Proposition 1: It is easy to check the following equality for any $\mathcal{S} \in \mathcal{C}_D$, $z_1^N \in (\mathcal{X}\times\mathcal{Y})^N$, and $\xi \in \{0, 1\}$, using the definition of $\pi(\mathcal{S})$ and of the Laplace estimator $Q^{K+1}_\mathcal{S}$:

$$\pi(\mathcal{S}) \prod_{i=K+1}^{N-1} Q^{K+1}_\mathcal{S}\left(y_i\,|\,x_i, z_1^K\right)^\beta Q^{K+1}_\mathcal{S}\left(y_N\,|\,x_N, z_1^K\right)^\xi = \prod_{s\in\mathcal{S}} w^{(\xi)}_{\mathcal{N}_\mathcal{S}(s)}(s).$$

As a result, the estimator $G^N_{\pi,\,\beta}$ can be expressed as follows:

$\forall\, z_1^N \in (\mathcal{X}\times\mathcal{Y})^N$,

$$G^N_{\pi,\beta}\left(y_N\,|\,x_N;\, z_1^{N-1}\right) = \frac{\sum\limits_{\mathcal{S}\in\mathcal{C}_D}\left(\prod\limits_{s\in\mathcal{S}} w^{(1)}_{\mathcal{N}_\mathcal{S}(s)}(s)\right)}{\sum\limits_{\mathcal{S}\in\mathcal{C}_D}\left(\prod\limits_{s\in\mathcal{S}} w^{(0)}_{\mathcal{N}_\mathcal{S}(s)}(s)\right)}.$$

Proposition 1 is a direct consequence of this equality and Lemma 2. □

B. Approximation by Model Selection

The implementation suggested by Proposition 1 using the functionals $\gamma^{(\xi)}$ involves the computation of a sum over $\mathcal{N} \subset \mathcal{A}$ at every node (see (18)). In real-world applications, the computation of this sum 2^a terms might be computationally too expensive if the size of the alphabet is too large.

As an alternative, one can observe that the estimator $G^N_{\pi,\,\beta}$ is a mixture of Laplace estimators

$$G^N_{\pi,\,\beta} = \sum_{\mathcal{S}\in\mathcal{C}_D} \rho(\mathcal{S}) Q^{K+1}_\mathcal{S}$$

and that this mixture should usually be unimodal in the space of conditional distributions, by construction of the Gibbs estimator. As a result, an approximation of $G^N_{\pi,\,\beta}$ is the Laplace estimator corresponding to the tree with highest posterior probability, i.e.,

$$G^{(est.)}_{\pi,\,\beta}\left(Y_N\,|\,X_N;\, Z_1^{N-1}\right) = Q^{K+1}_{\overline{\mathcal{S}}(Z_1^{N-1})}\left(Y_N\,|\,X_N;\, Z_1^K\right)$$

with

$$\begin{aligned}\overline{\mathcal{S}}(z_1^{N-1}) &= \arg\max_{\mathcal{S}\in\mathcal{C}_D}\; \rho(\mathcal{S})\\ &= \arg\max_{\mathcal{S}\in\mathcal{C}_D} \left\{\pi(\mathcal{S}) \prod_{i=K+1}^{N-1} Q^{K+1}_\mathcal{S}\left(y_i\,|\,x_i,\, z_1^K\right)^\beta\right\}\\ &= \arg\max_{\mathcal{S}\in\mathcal{C}_D} \left\{|\mathcal{S}|\,\frac{\log c_D}{\beta}\right.\\ &\qquad\left. + \log \prod_{i=K+1}^{N-1} Q^{K+1}_\mathcal{S}\left(y_i\,|\,x_i,\, z_1^K\right)\right\}.\end{aligned} \tag{19}$$

This formulation shows that $G^{(est.)}_{\pi,\,\beta}$ is obtained by a *penalized maximum-likelihood* selection procedure, where the penalization for the log-likelihood of a model $\mathcal{S}$ is $\kappa = (\log c_D)/\beta$ per node.

The implementation of this model selection procedure can follow the spirit of the implementation of the following mixture.

- For any subset $\mathcal{N} \in \mathcal{Y}$ and $s \in \mathcal{T}_D$ let

$$\overline{w}_\mathcal{N}(s) \overset{\text{def}}{=} \kappa + \sum_{y\in\mathcal{Y}} \left(\mu_V(s,\, y) - \sum_{i\in\mathcal{N}} \mu_V(is,\, y)\right) \times \log \frac{\mu_T(s,\, y) - \sum\limits_{i\in\mathcal{N}} \mu_T(is,\, y) + 1}{\nu_T(s) - \sum\limits_{i\in\mathcal{N}} \nu_T(is) + a}.$$

- Let $\overline{\gamma}$ be recursively defined on $\mathcal{T}_D$ by

$$\begin{cases} \overline{\gamma}(s) = \overline{w}_\emptyset(s), & \text{if } l(s) = D\\ \overline{\gamma}(s) = \max\limits_{\mathcal{N}\subset\mathcal{A}} \left\{\overline{w}^{(\xi)}_\mathcal{N} + \sum\limits_{i\in\mathcal{Y}\setminus\mathcal{N}} \overline{\gamma}(is)\right\}, & \text{otherwise.}\end{cases}$$

- For every $s \in \mathcal{T}_D$ if the nodes in the selected subset $\mathcal{N}$ used to compute $\overline{\gamma}(s)$ are marked, then $\overline{\mathcal{S}}$ is the largest tree made of marked nodes.

Remark 11: Another possibility to approximate the estimator $G^N_{\pi,\,\beta}$ would be to use a Monte Carlo Markov chain simulation to approximate the mixture (see [16] for a discussion in the framework of decision trees).

IX. Experiments and Natural Language Processing Applications

As an application for the estimators studied in this paper, we show here how they can be used to model texts written in natural language, and give results from a text clustering experiment based on these statistical models.

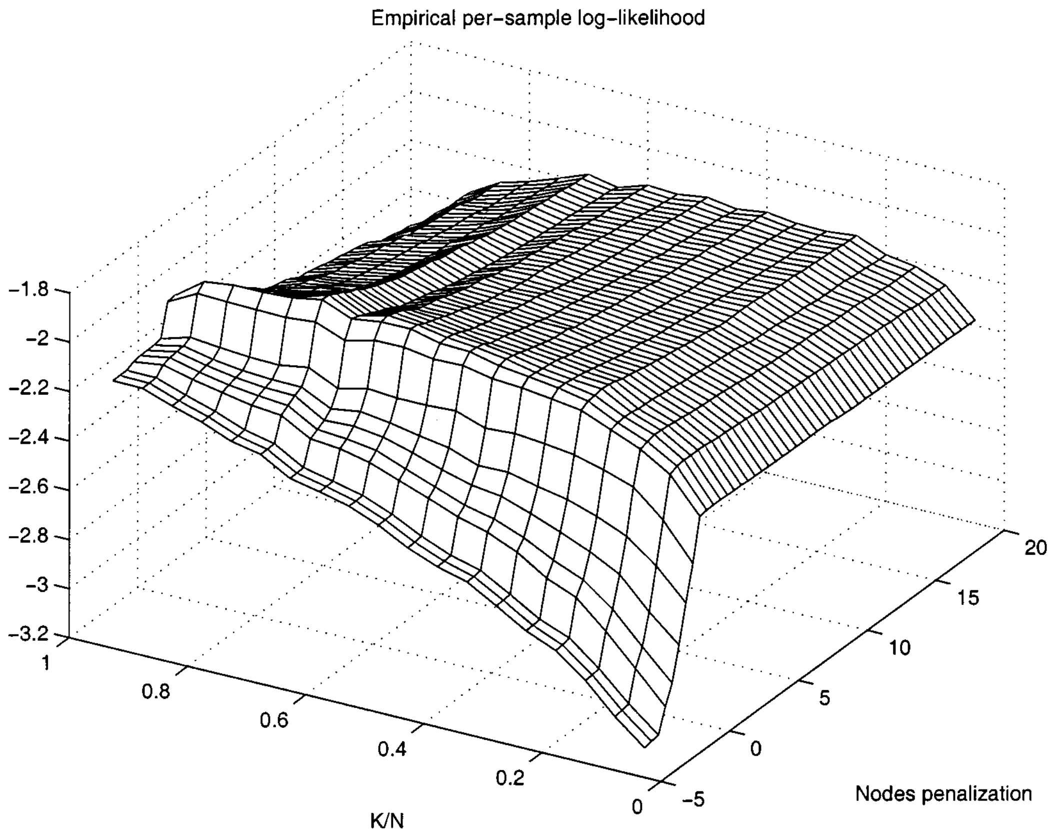

Fig. 3. Log-likelihood with $N = 20\,000$ for various K and $\log(c_D)/\beta$.

For a given alphabet $\mathcal{A}$, a text T written in natural language (e.g., in English or Japanese) is a string which can be parsed into a series of letters. One can think of $\mathcal{A}$ as the letters of the alphabet $\{a, b, \ldots, z\}$, the ASCII symbols set, a dictionary of words, or whatever set of symbols in terms of which the text can be represented as a sequence $(t_1, \ldots, t_{|T|})$ with $\forall i \in [1, |T|], t_i \in \mathcal{A}$.

For a given $D < |T|$, let $(X, Y) \in \mathcal{A}^D \times \mathcal{A}$ be the random variable obtained by randomly choosing an index $i \in [1, |T| - D]$ uniformly and setting

$$\begin{cases} X &= t_i \cdots t_{i+D-1} \\ Y &= t_{i+D}. \end{cases}$$

For a given N, let us consider the statistical experiment that consist in sampling N i.i.d. variables $(X_i, Y_i)_{i\in[1, N]}$ according to this common law. This experiment can be used to train any regression model to infer Y from X, which gives a representation of the initial text as a stochastic model. Note that the initial text is deterministic, and that the random nature of the variables comes from the sampling.

A. Tuning the Parameters

As an example, let us consider the model selection algorithm described in Section VIII-B. Equation (19) shows that the "cost" of adding a node to a model is $\log(c_D)/\beta$, which is a parameter we can try to optimize for a given problem. Note that if we were trying to compute the actual estimator which is a mixture of models, for instance using Monte Carlo simulations, two different parameters could be varied: c_D and β, which influence the shape of the prior and the speed of learning from examples, respectively.

A second parameter can be optimized: K/N, which is related to the relative sizes of the estimation and the validation sets.

In order to observe the effect of these two parameters, Figs. 3 and 4 show results of an experiment carried out from the text *Far from the Madding Crowd* by T. Hardy, which is the file "book1" of the Calgary corpus[1] (used in [24]). The text (in English) was parsed into a sequence of characters using the alphabet $\mathcal{A} = \{a, b, \ldots, z, O\}$, where O represents anything that is not a letter. The estimator was then trained on i.i.d. samples of size $N = 20\,000$ with varying K/N and $\log(c_D)/\beta$, and its likelihood was computed on a test set made of 5000 new i.i.d. samples. Fig. 3 shows the per-sample log-likelihood for varying $\log(c_D)/\beta$ and K/N, and Fig. 4 shows for clarity purpose the same curve for $K/N = 0.7$ being fixed.

For any K/N, the value $\log(c_D)/\beta = 0$ corresponds to the classical maximum-likelihood estimator. Negative values correspond to negative penalties and, therefore, favor large models. Positive values are more natural and correspond to penalizing more large models than small ones.

For $\log(c_D)/\beta < -3$, the likelihoods of the models on the test set are very low: this is the classical phenomenon of overfitting, that is favored by the negative penalization. In this region, indeed the selected model appears to be too large for its parameters to be accurately estimated. As $\log(c_D)/\beta$ increases to 0, the performance increases and peaks at a value a bit larger than

[1] Available at ftp://ftp.cpsc.ucalgary.ca/pub/projects/text.compression.corpus/.

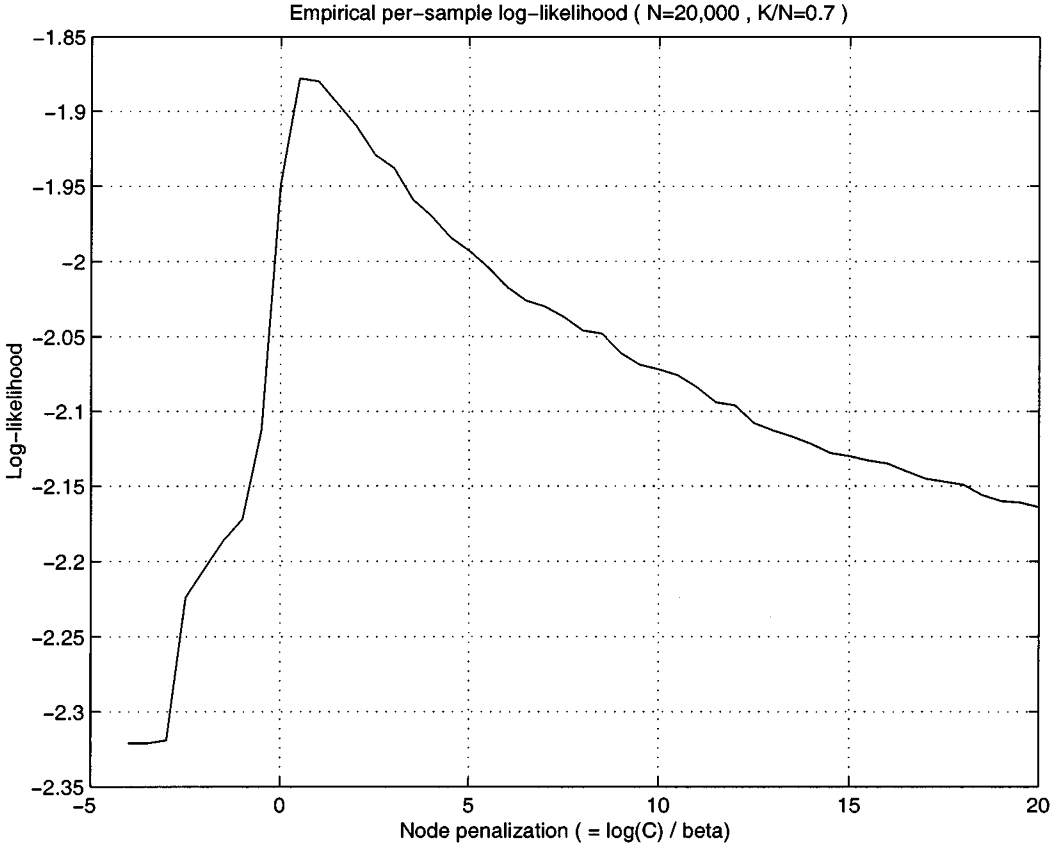

Fig. 4. Log-likelihood with $N = 20\,000$ for $K/N = 0.7$ and various $\log(c_D)/\beta$.

zero, which corresponds to the optimal penalty for the particular unknown probability and the particular sizes considered. Larger penalty values decrease the performance of the selected model on the test set because its dimension becomes too small. In that case, indeed, the gain in the variance term due to decreasing the number of parameters to estimate does not balance the increase of the bias term which corresponds to the distance between P and the selected model.

Fig. 3 also shows that for a given penalty there exists an optimal choice of division between the training set and the validation set, which corresponds to the balance between training the Laplace estimators and choosing the best model: it is better to have a training set a bit larger than the validation set. Naturally, as the penalty increases, the optimal K increases too, because increasing the penalty means giving less importance to each validation sample.

B. Comparison with Other Models

Many other statistical models can be used to characterize the relation between X and Y. In particular, the so-called N-gram models are widely known and used in natural language processing to characterize sequences of characters (e.g., for character recognition purposes) or words (e.g., for speech recognition purposes). In an N-gram model, the distribution of Y is supposed to depend on the suffix of length $N-1$ of X, with N being fixed.

Thus, N-grams are particular regression trees, i.e., complete trees of depth $N-1$. The difficulties arise when one wants to estimate the N^D distributions of Y from a finite training corpus. An adaptive approach, as the one described in this paper, is better at balancing the complexity of the model and the precision of the estimation which basically depends on the size of the training corpus.

As an example, Fig. 5 shows the log-likelihood of different models trained on i.i.d. samples of growing size (between 100 and 10 000) and tested on an i.i.d. sample of size 5 000. The models tested are as follows.

- N-gram models for $N = 1, 2, 3, 4$, with classical nonadaptative Laplace estimators.
- The aggregation using a Gibbs estimator, with classical nonadaptative Laplace estimators.
- The aggregation using a Gibbs estimator, with adaptive Laplace estimators.

Following the results of the first experiment, the parameters for aggregated estimator were set to $\log(c_D)/\beta = 0.5$ and $K/N = 0.65$.

This experiment shows that the estimator obtained by aggregation of Laplace estimators is almost as efficient as the best N-gram models for any training set size. It also shows the improvement gained with the introduction of the adaptive Laplace estimator and the adaptive probability on the model space. Indeed, it is clear that the support of the distributions of Y are often smaller than the whole alphabet (e.g., the character following the letter "q" should almost always be a "u" or a space),

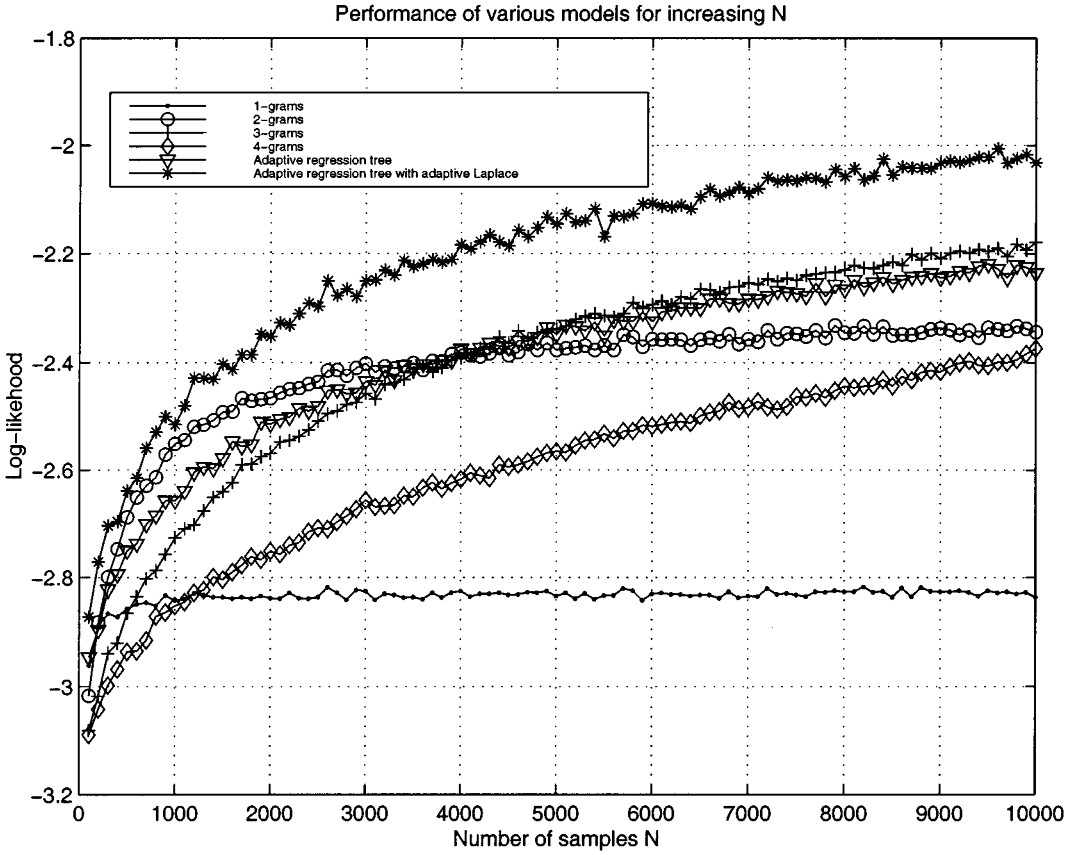

Fig. 5. Comparison with other models.

and that the strings X observed only form a small subset of the set of sequences of D characters.

C. Unsupervised Text Clustering

While the distribution of a letter following a string might have straightforward applications as such (e.g., for disambiguation purpose in optical character recognition systems), the estimator we study can be considered more generally as a way of *representing* a text because it is able to "learn" various statistical features very quickly.

As an example, it can be used to define and measure a notion of *distance* between texts. Indeed, let T_1 and T_2 be two given texts that one wants to compare. Using them to generate statistical experiments, it is natural to say they are close to each other if *the model that has been trained to explain the first statistical experiment is good at explaining the second one*, and far from each other otherwise.

This can be quantified as follows. Suppose each text is used to generate a statistical experiment on which an estimator is trained. This generates two models $Q_1(Y \mid X)$ and $Q_2(Y \mid X)$ which can be used afterwards to compute the likelihood of any sample $(x_i, y_i)_{i=1}^N$. In particular, one can define a pseudodistance between the two texts with the following formula:

$$d(T_1, T_2) = \log \frac{Q_1(\exp_1)}{Q_1(\exp_2)} + \log \frac{Q_2(\exp_2)}{Q_2(\exp_1)} \qquad (20)$$

where $\exp_i$ means the experiment that consists in sampling N i.i.d. pairs (x, y) from text T_i. This pseudodistance is symmetric and satisfies $d(T, T) = 0$ for any text T.

Let now a set of p texts $\{T_1, \ldots, T_p\}$ be given. The unsupervised text clustering problem is the problem of grouping these texts into a number of categories according to their similarities. Most existing clustering algorithms require a distance-like functional to be defined between any two elements to be clustered, that can be the pseudodistance defined by (20).

To illustrate this we took a series of eight books from each of which we extracted five texts, and computed the distance between any two of the resulting 40 texts (see Table I).

Each text was 12 000 characters long and was used to generate three files by i.i.d. sampling. The first two files (8000 and 4000 samples) were used as estimation and validation set, while the third file (5000 samples) was used as a test set to measure the likelihoods used in (20). The parameter $\log(c_D)/\beta$ was set to 0.5.

Fig. 6 is a typical profile of distances between one text (here the text number 23, extracted from Spinoza's *Political Treaty*) and all other texts. It shows that the distance with the four texts extracted from the same book (i.e., texts 21, 22, 24, and 25) are clearly smaller than the distances with the rest of the database, and that it could "recognize" the similarity within the texts extracted from the same book.

In Fig. 7, we plotted a "o" as soon as the distance between two texts was smaller than 1.03. Clusters corresponding to the books already appear with this naive thresholding method.

TABLE I
TEXT DATABASE

Text Number	Extracted from
1-5	Wintson Churchill (*The Crossing*)
6-10	Joseph Conrad (*The Arrow of gold*)
11-15	Arthur Conan Doyle (*The hound of the Baskervilles*)
16-20	Karl Marx (*Manifesto of the communist party*)
21-25	Baruch Spinoza (*Political treatise*)
26-30	Jonathan Swift (*Gulliver's travel*)
31-35	Francois Marie Arouet Voltaire (*Candide*)
36-40	Virginia Woolf (*Night and day*)

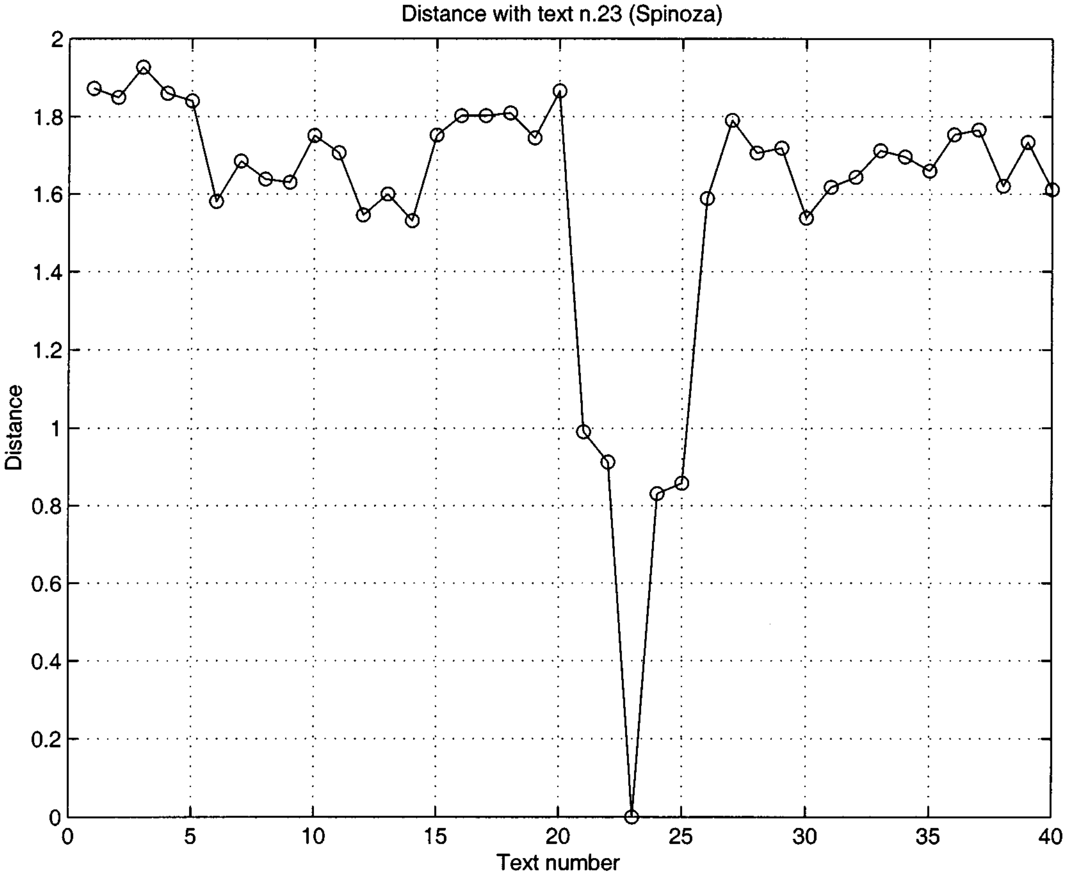

Fig. 6. Distance between text number 23 (Spinoza) and the other texts.

One should remark that no dictionary or preprocessing of the text was used. The usual way of representing a text as a "bag of words" in the literature about natural language processing is limited as far as statistical estimation is concerned because the number of possible words is much larger than the size of the text itself. On the other hand, we experimented on models based on characters only which lead to less risky estimations and encouraging results.

X. Conclusion

We presented a family of statistical estimators of a conditional distribution and proved upper bounds on their risk. The main characteristic of these estimators is their ability to find a good tradeoff between the bias of different models and the risk of their estimation for a given number of observations.

Such estimators are interesting in cases when the "real" law P_N is complicated, but progressively approximated by models of increasing dimensions. As an example we considered the issue of modeling texts written in natural language, for which classical Markovian models like N-grams are limited in depth because of the size of the training corpus that is needed. In spite of the simplicity of our models, encouraging experimental results lead us to believe that important improvement could be obtained by carefully designing pertinent models for a partic-

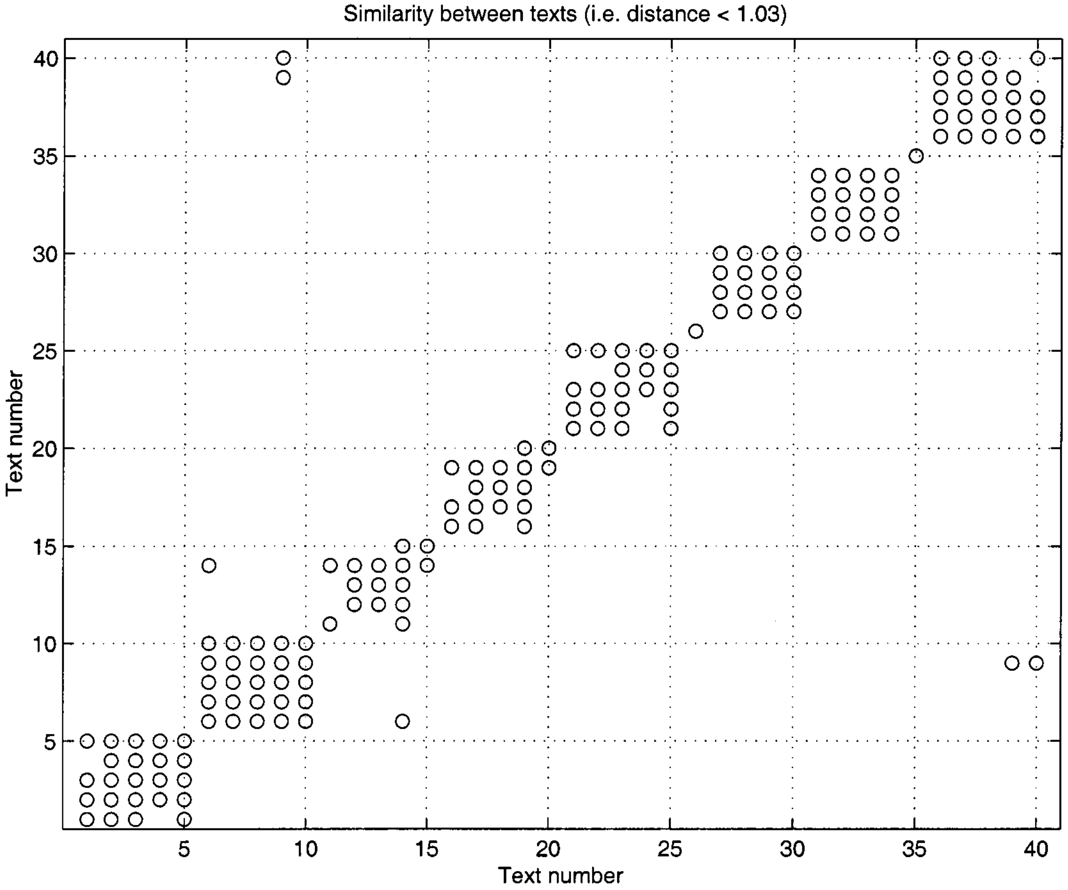

Fig. 7. Similarity between texts.

ular application while keeping in mind the necessity of efficient statistical estimations.

Acknowledgment

The author wishes to thank O. Catoni for his numerous suggestions and advice.

References

[1] G. Salton, A. Wong, and C. S. Yang, "A vector space model for automatic indexing," *Commun. ACM*, vol. 18, no. 11, pp. 613–620, Nov. 1975.

[2] R. Cole, J. Mariani, H. Uszkoreit, G. B. Varile, A. Zaenen, and A. Zampolli, Eds., *Survey of the State of the Art in Human Language Technology*. Cambridge, U.K.: Cambridge Univ. Press, 1998.

[3] L. D. Davisson, "Universal noiseless coding," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 783–795, Nov. 1973.

[4] J. Rissanen and G. G. Langdon, Jr., "Universal modeling and coding," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 12–23, Jan. 1981.

[5] L. D. Davisson, "Minimax noiseless universal coding for Markov sources," *IEEE Trans. Inform. Theory*, vol. IT-29, pp. 211–215, Mar. 1983.

[6] B. Y. Ryabko, "Twice-universal coding," *Probl. Inform. Transm.*, vol. 20, no. 3, pp. 24–28, July 1984.

[7] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens, "The context-tree weighting method: Basic properties," *IEEE Trans. Inform. Theory*, vol. 41, pp. 653–664, May 1995.

[8] M. Feder and N. Merhav, "Hierarchical universal coding," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1354–1364, Sept. 1996.

[9] A. R. Barron, J. Rissanen, and B. Yu, "The minimum description length principle in coding and modeling," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2743–2760, Oct. 1998.

[10] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[11] N. Merhav and M. Feder, "Universal prediction," *IEEE Trans. Inform. Theory*, vol. 44, pp. 2124–2147, Oct. 1998.

[12] B. Y. Ryabko, "A fast adaptive coding algorithm," *Probl. Inform. Transm.*, vol. 26, no. 4, pp. 305–317, 1990.

[13] R. E. Krichevskyi, "Laplace's law of succession and universal encoding," *IEEE Trans. Inform. Theory*, vol. 44, pp. 296–303, Jan. 1998.

[14] F. Topsoe, "Instances of exact prediction and a new type of inequalities obtained by anchoring," in *Proc. 1999 IEEE Information Theory and Communication Workshop*, Kruger National Park, South Africa, 1999, p. 99.

[15] O. Catoni, "'Universal' aggregation rules with exact bias boundst," *Ann. Statist.*, to be published.

[16] G. Blanchard, "The 'progressive mixture' estimator for regression trees," *Ann. Inst. Henri Poincaré, Probabilités et Statistiques*, vol. 35, no. 6, pp. 793–820, 1999.

[17] L. Devroye, L. Gyorfi, and G. Lugosi, *A Probabilistic Theory of Pattern Recognition*. New York: Springer-Verlag, 1996.

[18] O. Catoni, "Gibbs estimators," Ecole Normale Supérieure, Dept. Math. Applications, preprint LMENS-98-21, May 1998.

[19] B. Y. Ryabko, "Prediction of random sequences and universal coding," *Probl. Inform. Transm.*, vol. 24, no. 2, pp. 87–96, 1988.

[20] F. M. J. Willems, Y. M. Shtarkov, and T. J. Tjalkens, "Context weighting for general finite-context sources," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1514–1520, Sept. 1996.

[21] A. R. Barron, "Are Bayes rules consistent in information?," in *Open Problems in Communication and Computation*, T. M. Cover and B. Gopinath, Eds. New York: Springer-Verlag, 1987.

[22] A. R. Barron and Y. Yang, "Information-theoretic determination of minimax rates of convergence," *Ann. Statist.*, vol. 27, no. 5, pp. 1564–1599, 1999.

[23] V. N. Vapnik, *Statistical Learning Theory*. New York: Wiley, 1998.

[24] T. C. Bell, J. G. Cleary, and I. H. Witten, *Text Compression*. Englewood Cliffs, NJ: Prentice-Hall, 1990.