Bounds for Learning Lossless Source Coding

Anders Høst-Madsen
Department of Electrical Engineering
University of Hawaii, Manoa
Honolulu, HI, 96822, Email: ahm@hawaii.edu

***Abstract*—This paper asks a basic question: how much training is required to beat a universal source coder? Traditionally, there have been two types of source coders: fixed, optimum coders such as Huffman coders; and universal source coders, such as Lempel-Ziv. The paper considers a third type of source coders: learned coders. These are coders that are trained on data of a particular type, and then used to encode new data of that type. This is a type of coder that has recently become popular for (lossy) image and video coding.**

The paper evaluates two criteria for performance of learned coders: the average performance over training data, and a guaranteed performance for all training except for some error probability P_e, which is PAC learning. In both cases the coders are evaluated with respect to redundancy.

The paper considers the independent identically distributed (IID) binary case and binary Markov chains. In both cases it is shown that the amount of training data required is very moderate: to code sequences of length l the amount of training data required to beat a universal source coder is $m = K\frac{l}{\log l}$, where the constant K depends on the case considered.

I. Introduction

Traditionally, there have been two types of source coders: fixed, optimum coders such as Huffman coders; and universal source coders, such as Lempel-Ziv [1], [2], [3]. We will consider a third type of source coders: learned coders. These are coders that are trained on data of a particular type, and then used to encode new data of that type. Examples could be source coders for English texts, DNA data, or protein data represented as graphs.

In both machine learning and information theory literatures, there has been some work on learned coding. From a machine learning perspective, the paper [4] stated the problem precisely and developed and evaluated some algorithms. A few follow up papers, e.g., [5], [6], [7], [8], [9], [10] have introduced new machine learning algorithms. For lossy coding, in particular of images and video, there has been much more activity recently, initiated by the paper [11] from Google, see for example [12], [13], [14]. Our aim is to find theoretical bounds for how well it is possible to learn coding. In the current paper we will limit ourselves to lossless coding. From an information theory perspective, Hershkovits and Ziv [15] considered learned coding in terms of learning a database of sequences. The results in [15] are quite pessimistic. Basically they state that to code a sequence of length l so as to approach the entropy rate $\mathcal{H}$, a length $2^{l\mathcal{H}}$ training sequence is needed – so that one observes most of the typical sequences. This means that essentially learned coding is infeasible, as the amount of training needed is exponential in the sequence length!

The research was funded in part by the NSF grant CCF-1908957.

Our perspective on learning coding is to compare with universal source coders with redundancy as measure. The redundancy of a coder is is the difference between the entropy of a source and the average length achieved by the coder. Suppose that the sources are (or assumed to be) in some probability class Λ characterized by a parameter vector $\boldsymbol{\theta}$. For a universal source coder with length function L, the redundancy to encode a sequence of length l is defined by [16]

$$R_l(L,\boldsymbol{\theta}) = \frac{1}{l}E_{\boldsymbol{\theta}}[L(X^l)] - H_{\boldsymbol{\theta}}(X),$$

where x^l denotes a sequence of length l. Since $\boldsymbol{\theta}$ is unknown, even in terms of probability law, usually the minimax redundancy is considered [16]

$$R_l^+ = \min_L \sup_{\boldsymbol{\theta}} R_l(L,\boldsymbol{\theta}).$$

A good coder is one that achieves this minimum.

This setup can be generalized to learning. We are given a training sequence x^m; based on the training we develop coders $C(x^l;x^m)$ with length function $L(x^l;x^m)$ for encoding *test sequences* x^l. The codelength is $\frac{1}{l}E_\theta[L(X^l;x^m)|x^m]$ (the expectation here is only over x^l), and the redundancy is

$$R_l(L,x^m,\boldsymbol{\theta}) = \frac{1}{l}E_{\boldsymbol{\theta}}[L(X^l;x^m)|x^m] - H_{\boldsymbol{\theta}}(X). \quad (1)$$

The redundancy depends on the training sequence x^m. *One way* to remove this dependency is to average also over x^m,

$$R_l(L,m,\boldsymbol{\theta}) = \frac{1}{l}E_{\boldsymbol{\theta}}[L(X^l;X^m)] - H_{\boldsymbol{\theta}}(X) \quad (2)$$

$$R_l^+(m) = \min_L \sup_{\theta} R_l(L,m,\boldsymbol{\theta}). \quad (3)$$

The idea of learning to code is to obtain information about the distribution of the source from the training x^m and then apply this to code the test sequence x^l. One can take two approaches to the application phase. First, the the coder can be *frozen* in the sense that it does not further update from the test sequence. There are both practical and theoretical reasons for freezing the coder. Machine learning algorithms usually have a distinct learning phase, and once the algorithm is trained, it is not updated with test samples; the reason for this is both that training is much more computational intensive than application, often run on specialized hardware, and that there are few good algorithms for updating for example neural networks with new data. As a case in point, the LSTM in [4] was not updated after the training phase, and the theoretical work in [15] also considered frozen coders. Here we consider only frozen coders, but the journal version [17] also will consider non-frozen coding.

The question we consider now is: how many training samples do we need in order to beat a universal source coder, i.e., how large should m be so that

$$R_l^+(m) \leq R_l^+. \tag{4}$$

Learning to code has similarities with universal prediction [18]. The paper [19] developed bounds for universal prediction for IID (independent identically distributed) sources, and [20], [21] for Markov sources. In fact, the paper [19] exactly considers (3), and proves

$$\frac{1}{2m\ln 2} + o\left(\frac{1}{m}\right) \leq R_l^+(m) \leq \frac{\alpha_0}{m\ln 2} + o\left(\frac{1}{m}\right) \tag{5}$$

$$\alpha_0 \approx 0.50922. \tag{6}$$

(The result was improved to $\alpha_0 = \frac{1}{2}$ in [22]). On the other hand, we also have good expressions for R_l^+ [16], which can be expressed as[1] $R_l^+ = \frac{\log l}{2l} + o\left(\frac{1}{l}\right)$. Thus, ignoring o-terms, (4) becomes

$$m \geq \frac{l}{\ln 2 \log l}. \tag{7}$$

The conclusion is that it is very easy to beat a universal coder. For l moderately large $\frac{l}{\ln 2 \log l} < l$, so we need fewer training samples than the length of the sequences we want to encode.

We now return to (2). Averaging over the training x^m might be reasonable for universal prediction. But in learning one usually learns once and applies many times. The average codelength over test sequences in (1) is therefore reasonable, but the averaging over the training less so. Instead one can require that the training is good for most training sequences, or, put another way, that the probability of a bad training sequence is low. So, we consider the criterion

$$E(m,a) = \sup_{\boldsymbol{\theta}} P\left(R_l(L, X^m, \boldsymbol{\theta}) > a\right), \tag{8}$$

where the probability is over X^m. For some given a and small P_e the goal is then to ensure

$$E(m,a) \leq P_e.$$

This criterion will be recognized as the well-known PAC (probability approximately correct) learning criterion [23] applied to lossless coding, with redundancy as loss function. Again, we can consider the bottom line of beating the universal coder, in which case $a = \frac{\log l}{2l}$ for the IID case.

II. IID Case

We consider the IID binary case characterized by the parameter $\theta = p$, where $p = P(X = 1)$ and $q = 1 - p$. Learning a coder boils down to finding an estimator $\hat{p}$. It is then well known [3] that the redundancy of a coder defined by $\hat{p}$ is $R_l(L, x^m, \theta) = D(p\|\hat{p}) = p\log\frac{p}{\hat{p}} + (1-p)\log\frac{1-p}{1-\hat{p}}$ (except for some small constant), and therefore

$$E(m,a) = \sup_p P\left(D\left(p\|\hat{p}\right) \geq a\right).$$

[1]All logaritms in the paper is to base 2 except when ln is explicitly used.

We consider estimators $\hat{p}$. We assume that $\hat{p} = f(\check{p})$, where

$$\check{p} = \frac{k}{m},$$

with k the number of ones, is the minimal sufficient statistic; the function f can depend on m. For convenience, we assume f is invertible. Let

$$P(p,a) = P\left(D(p\|\hat{p}) > a\right)$$

for fixed $p \leq \frac{1}{2}$. The equation $D(p\|\hat{p}) = a$ has two solutions $\hat{p}_\pm$ so that

$$P(p,a) = P\left(\check{p} < f^{-1}(\hat{p}_-)\right) + P\left(\check{p} > f^{-1}(\hat{p}_+)\right),$$

the sum of the *lower* and *upper* tail probabilities.

We consider what can be named the moderate deviations regime. We fix P_e independent of m and require $E(m,a) \leq P_e$ and desire to find the smallest $a(m,P_e)$ that satisfies this inequality. We solve the problem asymptotically as $m \to \infty$; necessarily $a(m,P_e) \to 0$, and we want to find how it converges to zero. This essentially gives the redundancy as a function of m. We can use this to determine how many training samples we need to beat universal source coding: solving $a(m,P_e) < \frac{l}{2\log l}$.

Let $0 \leq \lambda \leq 1$ and put

$$\hat{p}_-(p,m,\lambda P_e) = \inf\{\hat{p} : P\left(\check{p} < f^{-1}(\hat{p})\right) \leq \lambda P_e\}$$
$$\hat{p}_+(p,m,(1-\lambda)P_e) = \sup\{\hat{p} : P\left(\check{p} > f^{-1}(\hat{p})\right) \leq (1-\lambda)P_e\}.$$

Then we can write

$$a(m,P_e) = \min_\lambda \sup_p \max\{D(p\|\hat{p}_-(p,m,\lambda P_e)), D(p\|\hat{p}_+(p,m,(1-\lambda)P_e))\}. \tag{9}$$

For achievability, we consider estimators of the well-known form [3], [24], [19]

$$\hat{p} = \frac{k+\alpha}{m+2\alpha}. \tag{10}$$

The main result is

Theorem 1. *For estimators that are functions of the sufficient statistic and P_e sufficiently small,*

$$a(m,P_e) \geq \frac{Q^{-1}(P_e/2)^2}{2m\ln 2} + o\left(\frac{1}{m}\right), \tag{11}$$

with $Q(x) = 1 - \Phi(x)$, Φ being the Gaussian CDF. The estimator (10) has on optimum value of α that satisfies

$$\frac{1}{6}Q^{-1}(P_e/2)^2 - 1 \leq \alpha \leq \frac{1}{6}Q^{-1}(P_e/2)^2 + 1, \tag{12}$$

which gives an achievable $a(m,P_e)$;

$$a(m,P_e) = b(P_e)\frac{Q^{-1}(P_e/2)^2}{2m\ln 2} + o\left(\frac{1}{m}\right), \tag{13}$$

where

$$\lim_{P_e \to 0} b(P_e) = 1.$$

Proof: Space only allows for a proof outline. The complete proof can be found in [17]. It can easily be seen that we

can use convergent sequences in the proof technique. We can divided such sequences into three regimes:

- *CLT regime*: $\lim_{m\to\infty} mp(m) = \infty$. In this regime the central limit theorem (CLT) can be applied.
- *Poisson regime*: $0 < \lim_{m\to\infty} mp(m) < \infty$. In this regime a Poisson approximation can be used.
- *Sub-Poisson regime*: $\lim_{m\to\infty} mp(m) = 0$.

We consider the limit of $ma(m, P_e)$ in each of these regimes, and maximizes over these limits. In the following we will drop the explicit dependency $p(m)$ and just write p. It turns out the worst achievable performance is in the Poisson regime, which gives (13), and we will therefore here only include the analysis for this regime in the conference paper. On the other hand, for the lower bound we will consider the CLT regime – a lower bound in one regime is clearly a lower bound everywhere.

CLT Regime: We can use the central limit theorem, here for the upper tail,

$$\begin{aligned} P\left(\check{p} > f^{-1}(\hat{p}_+)\right) &= P\left(\check{p} - p > f^{-1}(\hat{p}_+) - p\right) \\ &= P\left(\frac{\sqrt{m}}{\sqrt{pq}}(\check{p} - p) > \frac{\sqrt{m}}{\sqrt{pq}}(f^{-1}(\hat{p}_+) - p)\right) \\ &\to Q\left(\lim_{m\to\infty} \frac{\sqrt{m}}{\sqrt{pq}}(f^{-1}(\hat{p}_+) - p)\right) \quad (14) \end{aligned}$$

for $m \to \infty$, which can be seen from Berry-Esseen [25]. Therefore

$$\hat{p}_+ = f\left(\frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}((1-\lambda)P_e + \epsilon(m)) + p\right). \quad (15)$$

In the following we will omit the $\epsilon(m)$ as it does not affect the results.

We derive a converse in the CLT regime, more specifically for p constant rather than a function of m. We use use Pinsker's inequality for relative entropy [26], $D(p\|\hat{p}_+) \geq \frac{2}{\ln 2}(p - \hat{p}_+)^2$ in (9),

$$\begin{aligned} a(m, P_e) \geq \frac{2}{\ln 2} \min_f \min_\lambda \sup_p \max \\ \left\{ \left(f\left(\frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}((1-\lambda)P_e) + p\right) - p\right)^2, \right. \\ \left. \left(f\left(-\frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}(\lambda P_e) + p\right) - p\right)^2 \right\}. \quad (16) \end{aligned}$$

Let $f(x) = x + g_m(x)$, where we have made explicit that f can depend on m. We can then write this as

$$\begin{aligned} a(m, P_e) \geq \frac{2}{m \ln 2} \min_f \min_\lambda \sup_p \max \\ \left\{ \left(\sqrt{pq} Q^{-1}((1-\lambda)P_e) + \sqrt{m} g_m\left(\frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}((1-\lambda)P_e) + p\right)\right) \right. \\ \left. \left(-\sqrt{pq} Q^{-1}(\lambda P_e) + \sqrt{m} g_m\left(-\frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}(\lambda P_e) + p\right)\right)^2 \right\}. \quad (17) \end{aligned}$$

We will argue that $g_m = 0$ and $\lambda = \frac{1}{2}$ is optimum, or more precisely that $\lim_{m\to\infty} \sqrt{m} g_m = 0$. Suppose that for some p, $\lim_{m\to\infty} \sqrt{m} g_m \left(\frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}((1-\lambda)P_e) + p\right) = b$, so that

$$\begin{aligned} \lim_{m\to\infty} \Bigg(\sqrt{pq} Q^{-1}((1-\lambda)P_e) \\ + \sqrt{m} g_m\left(\frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}((1-\lambda)P_e) + p\right)\Bigg)^2 \\ = \left(\sqrt{pq} Q^{-1}((1-\lambda)P_e) + b\right)^2. \end{aligned}$$

Let p_m be the solution to

$$-\frac{\sqrt{p_m q_m}}{\sqrt{m}} Q^{-1}(\lambda P_e) + p_m = \frac{\sqrt{pq}}{\sqrt{m}} Q^{-1}((1-\lambda)P_e) + p.$$

Then

$$\begin{aligned} \lim_{m\to\infty} \Bigg(-\sqrt{p_m q_m} Q^{-1}(\lambda P_e) \\ + \sqrt{m} g_m\left(-\frac{\sqrt{p_m q_m}}{\sqrt{m}} Q^{-1}(\lambda P_e) + p_m\right)\Bigg)^2 \\ = \left(-\sqrt{pq} Q^{-1}(\lambda P_e) + b\right)^2. \end{aligned}$$

Thus the maximum in (17) as $m \to \infty$ becomes

$$\max\left\{ \left(\sqrt{pq} Q^{-1}((1-\lambda)P_e) + b\right)^2, \left(-\sqrt{pq} Q^{-1}(\lambda P_e) + b\right)^2 \right\}. \quad (18)$$

Since there is a minimization over f and λ, we can choose λ and b. It is now easily seen that (18) is minimized for $\lambda = \frac{1}{2}$ and $b = 0$ due to the convexity of $Q^{-1}(x)$ for $x < \frac{1}{2}$.

Thus, in (16) the minimum is achieved for f the identity and $\lambda = \frac{1}{2}$, while the maximum over p is achieved for $p = q = \frac{1}{2}$. This gives (13) as a lower bound.

Poisson regime: Let $p = \frac{\gamma}{m}$. We also set $\kappa_\pm = m\hat{p}_\pm$. Then

$$D\left(\frac{\gamma}{m}\middle\|\frac{\kappa_\pm}{m}\right) = \frac{\kappa_\pm - \gamma + \gamma \ln \gamma - \gamma \ln \kappa_\pm}{m \ln 2} + o\left(\frac{1}{m}\right) \quad (19)$$

We define

$$d(x, y) = y - x + x \ln \frac{x}{y}.$$

Now

$$\begin{aligned} P\left(\check{p} \leq f^{-1}(\hat{p}_-)\right) &= P\left(k \leq \kappa_-\left(1 + \frac{2\alpha}{m}\right) - \alpha\right) \\ &\to \mathbb{P}_\gamma(\kappa_- - \alpha), \end{aligned}$$

where $\mathbb{P}_\gamma$ is the Poisson CDF. Similarly

$$P\left(\check{p} > f^{-1}(\hat{p}_-)\right) \to 1 - \mathbb{P}_\gamma(\kappa_+ - \alpha).$$

Figure 1 illustrates the proof.

We will first analyze the lower tail probability corresponding to κ_-. Let γ_k, $k = 0, 1, \ldots$ be the sequence of solutions $\mathbb{P}_{\gamma_k}(k) = \frac{P_e}{2}$ – these correspond to the peaks in the solid blue curve in Fig. 1. Notice that if $\gamma_{k-1} < \gamma \leq \gamma_k$, $|\gamma - k| \leq |\gamma_k - k|$, and this is also true for other distance measures. Now, according to [27][2],

$$\frac{P_e}{2} = \mathbb{P}_{\gamma_k}(k) < \Phi\left(\text{sign}(k + 1 - \gamma_k)\sqrt{2d(k+1, \gamma_k)}\right),$$

[2]While [27] states the bounds for $k = 1, 2, \ldots$, it is easy to see that the bounds are also valid for $k = 0$, and the upper bound is also valid or non-integer values of k.

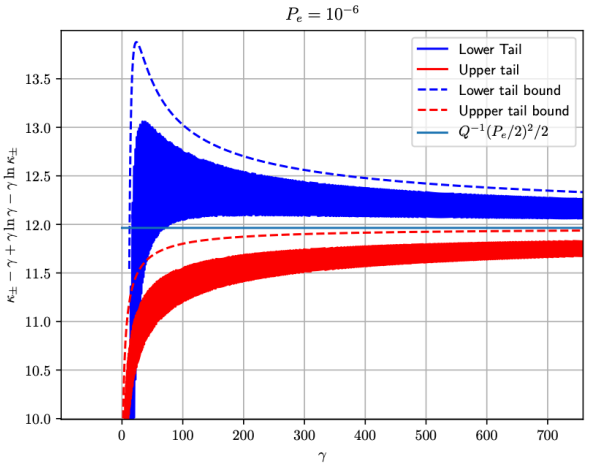

Figure 1. Plot of $d(\tilde{\gamma}, \tilde{\kappa}_-), d(\tilde{\gamma}, \tilde{\kappa}_+)$ for $P_e = 10^{-6}$ for $\alpha = \frac{1}{6}Q^{-1}(P_e/2)^2 - 1$. The solid curves are for the exact values of $\kappa_\pm$, while the dashed curves are the bounds. The solid curves are sawtooth like, but this cannot be seen at the scale of the figure. The bounds are for the peaks of the solid curves.

so that $\frac{1}{2}\Phi^{-1}(P_e/2)^2 > d(k+1, \gamma_k)$ (since $k+1 < \gamma_k$). As $\kappa_- = k + \alpha$, we therefore have

$$d(\kappa_- - \alpha + 1, \gamma) < \frac{1}{2}\Phi^{-1}(P_e/2)^2 = \frac{1}{2}Q^{-1}(P_e/2)^2$$

By normalizing by $Q^{-1}(P_e/2)^2$ we get

$$d(\tilde{\kappa}_- - \tilde{\alpha}_-, \tilde{\gamma}) < \frac{1}{2}, \tag{20}$$

where specifically $\tilde{\alpha}_- = \frac{\alpha - 1}{Q^{-1}(P_e/2)^2}$. From (19) it can be seen that $a(m, P_e)$ is determined by the swapped relative entropy, $d(\tilde{\gamma}, \tilde{\kappa}_-)$. Solving (20) with equality we get

$$\tilde{\kappa}_- - \tilde{\alpha}_- = r_-(\tilde{\gamma})\tilde{\gamma} = \frac{\frac{1}{2\tilde{\gamma}} - 1}{W_{-1}\left(\frac{1}{e}\left(\frac{1}{2\tilde{\gamma}} - 1\right)\right)}\tilde{\gamma}, \tag{21}$$

where W_{-1} is the Lambert W-function of order -1.

For the upper tail probability, we instead define λ_k, $k = 0, 1, \ldots$ as the sequence of solutions $\mathbb{P}_{\lambda_k}(k) = 1 - \frac{P_e}{2}$. We use the lower bound from [27],

$$1 - \frac{P_e}{2} = \mathbb{P}_{\gamma_k}(k) > \Phi\left(\text{sign}(k - \gamma_k)\sqrt{2d(k, \gamma_k)}\right)$$
$$d(k, \gamma_k) < \frac{1}{2}\Phi^{-1}(1 - P_e/2)^2 = \frac{1}{2}Q^{-1}(P_e/2)^2.$$

We then have

$$\tilde{\gamma} - (\tilde{\kappa}_+ - \tilde{\alpha}_+) + (\tilde{\kappa}_+ - \tilde{\alpha}_+)\ln\frac{\tilde{\kappa}_+ - \tilde{\alpha}_+}{\tilde{\gamma}} < \frac{1}{2}, \tag{22}$$

where now $\tilde{\alpha}_+ = \frac{\alpha + 1}{Q^{-1}(P_e/2)^2}$. The solution of (22) with equality is

$$\tilde{\kappa}_+ - \tilde{\alpha}_+ = r_+(\tilde{\gamma})\tilde{\gamma} = \frac{\frac{1}{2\tilde{\gamma}} - 1}{W_0\left(\frac{1}{e}\left(\frac{1}{2\tilde{\gamma}} - 1\right)\right)}\tilde{\gamma}. \tag{23}$$

The problem is now reduced to finding

$$\sup_{\gamma > 0} \max\{d(\tilde{\gamma}, \tilde{\kappa}_-), d(\tilde{\gamma}, \tilde{\kappa}_+)\} \triangleq \frac{1}{2}b(P_e) \tag{24}$$
$$d(\tilde{\kappa}_- - \tilde{\alpha}_-, \tilde{\gamma}) = \frac{1}{2}, d(\tilde{\kappa}_+ - \tilde{\alpha}_+, \tilde{\gamma}) = \frac{1}{2}.$$

We notice that $d(\tilde{\gamma}, \tilde{\kappa}_-)$ is decreasing with α while $d(\tilde{\gamma}, \tilde{\kappa}_+)$ is increasing. We will first show that if $\tilde{\alpha}_+ \le \tilde{\alpha}_+^*$ for some $\tilde{\alpha}_+^*$, then $\sup_\gamma d(\tilde{\gamma}, \tilde{\kappa}_+) \le \frac{1}{2}$.

By inserting (23) in $d(\tilde{\gamma}, \tilde{\kappa}_+)$ we find that (Mathematica)

$$\lim_{\tilde{\gamma}\to\infty} \tilde{\kappa}_+ - \tilde{\gamma} + \tilde{\gamma}\ln\frac{\tilde{\gamma}}{\tilde{\kappa}_+} = \frac{1}{2}. \tag{25}$$

It can be proven that if $\tilde{\alpha}_+ \le \frac{1}{6}$ *and* $d(\tilde{\gamma}, \tilde{\kappa}_+) > \frac{1}{2}$, then $d(\tilde{\gamma}, \tilde{\kappa}_+)$ is increasing in $\tilde{\gamma}$, which would then contradict the limit (25). The proof is technical and can be found in [17].

We now argue by contradiction. Let $\tilde{\alpha}_+ \le \frac{1}{6}$. If at some time $\tilde{\kappa}_+ - \tilde{\gamma} + \tilde{\gamma}\ln\frac{\tilde{\gamma}}{\tilde{\kappa}_+} > \frac{1}{2}$ then $\tilde{\kappa}_+ - \tilde{\gamma} + \tilde{\gamma}\ln\frac{\tilde{\gamma}}{\tilde{\kappa}_+}$ is increasing, thus it stays strictly above $\frac{1}{2}$. But then the limit (25) cannot be achieved. Thus we conclude that we must have $\tilde{\kappa}_+ - \tilde{\gamma} + \tilde{\gamma}\ln\frac{\tilde{\gamma}}{\tilde{\kappa}_+} \le \frac{1}{2}$.

For the lower tail, the above argument can be repeated, where we now have $\tilde{\alpha}_- \ge \frac{1}{6}$.

Since $\tilde{\alpha}_- < \tilde{\alpha}_+$ we cannot have both $\tilde{\alpha}_- \ge \frac{1}{6}$ and $\tilde{\alpha}_+ \le \frac{1}{6}$. However, we can choose α so that both $\lim_{P_e\to 0}\tilde{\alpha}_- = \lim_{P_e\to 0}\tilde{\alpha}_+ = \frac{1}{6}$. We can use this to conclude that in the limit both $\sup_{\gamma>0} d(\tilde{\gamma}, \tilde{\kappa}_-), \sup_{\gamma>0} d(\tilde{\gamma}, \tilde{\kappa}_+) \le \frac{1}{2}$. To reach this conclusion, notice that explicitly

$$d(\tilde{\gamma}, \tilde{\kappa}_+) = r_+(\tilde{\gamma})\tilde{\gamma} + \tilde{\alpha} + \tilde{\gamma}\ln\frac{\tilde{\gamma}}{r_+(\tilde{\gamma})\tilde{\gamma} + \tilde{\alpha}}$$
$$\frac{\partial}{\partial\alpha}d(\tilde{\gamma}, \tilde{\kappa}_+) = 1 - \frac{\tilde{\gamma}}{r_+(\tilde{\gamma})\tilde{\gamma} + \tilde{\alpha}}$$

Since $\lim_{\tilde{\gamma}\to\infty} r_+(\tilde{\gamma}) = 1$, the derivative is bounded over $\tilde{\gamma}$. Thus, for any $\epsilon > 0$ we can find a $\delta > 0$ so that if $\tilde{\alpha} < \tilde{\alpha}_0 + \delta$, $|d(\tilde{\gamma}, \tilde{\kappa}_+)|_{\tilde{\alpha}=\tilde{\alpha}_0} - d(\tilde{\gamma}, \tilde{\kappa}_+)|_{\tilde{\alpha}=\tilde{\alpha}_0+\delta}| < \epsilon$ for all $\tilde{\gamma}$, a kind of uniform continuity. We now conclude that when $\tilde{\alpha}_+ \to \frac{1}{6}$, $\sup_{\gamma>0} d(\tilde{\gamma}, \tilde{\kappa}_+) \to \frac{1}{2}$. This shows that $b(P_e)$ given by (24) converges to 1 as $m \to \infty$.

■

The first thing to notice from this result is that as for average performance, the performance increases as $\frac{1}{m}$. Specifically, to beat universal coding of sequences of maximum length l with probability $1 - P_e$ the number of training samples is approximately

$$m \ge \frac{Q^{-1}(P_e/2)^2}{2\ln 2}\frac{l}{\log l}. \tag{26}$$

The other thing to remark is that the upper and lower bounds are only tight in the limit: they are separated by a factor $b(P_e)$. An upper bound on $b(P_e)$ is

$$b(P_e) \le 2\max_{\tilde{\gamma}>0} \tilde{\kappa}_- - \tilde{\gamma} + \tilde{\gamma}\ln\tilde{\gamma} - \tilde{\gamma}\ln\tilde{\kappa}_-, \tag{27}$$

where $\tilde{\kappa}_-$ is given by (21). A plot of this upper bound on $b(P_e)$ can be seen in Fig. 2 below.

III. Extension to Markov Chains

The results for the IID case can be extended to Markov chains. We will provide the results here, and refer to [17] for proofs. We consider a binary Markov chain with states $0, 1$. Let $p_i = p(i|i)$ be the probability of staying in state i when the current state is i. The stationary probability is $\pi_i = \frac{p_i}{p_0+p_1}$.Based on training, estimates $\hat{p}_0$ and $\hat{p}_1$ are generated. The redundancy for coding then is [3]

$$\pi_0 D(p_0\|\hat{p}_0) + \pi_1 D(p_1\|\hat{p}_1).$$

We consider the two measures of performance

$$R_l^+(m) = \sup_{p_0,p_1} E\left[\pi_0 D(p_0\|\hat{p}_0) + \pi_1 D(p_1\|\hat{p}_1)\right] \quad (28)$$

$$E(m,a) = \sup_{p_0,p_1} P\left(\pi_0 D(p_0\|\hat{p}_0) + \pi_1 D(p_1\|\hat{p}_1) \geq a\right). \quad (29)$$

In [20], [21] universal prediction (called estimation) for Markov chains was considered, as an extension of [19]. It was shown that the estimation error decreases as $\frac{\log\log m}{m}$, which is an interesting contrast to (5). However, for learned coding the redundancy does not decrease at all with the length of the training sequence,

Proposition 1. *Assume that the training data consists of a single sequence. Then*

$$R_l^+(m) \geq \frac{1}{2}$$
$$E(m,a) = 1 \quad \textit{for } a < \frac{1}{2}.$$

The issue is that if the Markov Chain is slowly mixing, the training sequence might see only a single state, whereas the test sequence might be from the other state. It is clear that multiple training sequences are required for learning how to code. For achievability, we let the training data consist of n sequences each of length s with $m = ns$, where each sequence has an *independent* initial state according to the stationary distribution. For the converse we allow optimization of initial states.

Theorem 2. *Consider a binary Markov chain. Assume that the training consists of a set of sequences, so that both the size of the set and the length of each sequence approach infinity. For the estimator (10), with $\alpha = \alpha_0$ (6) we get*

$$R_l^+(m) = \frac{2\alpha_0}{m\ln 2} + o\left(\frac{1}{m}\right) \quad (30)$$

while a lower bound is

$$R_l^+(m) \geq \frac{1}{m\ln 2} + o\left(\frac{1}{m}\right). \quad (31)$$

Theorem 3. *Consider a binary Markov chain. Assume that the training consists of a set of sequences, so that both the size of the set and the length of the sequences approach infinity. Using the estimator from Theorem 1, the following decay is achievable*

$$a(m,P_e) = 2b(P_e)\frac{Q^{-1}((1-\sqrt{1-P_e})/2)^2}{2m\ln 2} + o\left(\frac{1}{m}\right).$$

This is achievable for

$$\left|\frac{1}{6}Q^{-1}((1-\sqrt{1-P_e})/2)^2 - \alpha\right| \leq 1.$$

Theorem 4. *A lower bound is*

$$a(m,P_e) \geq \frac{F_{\chi_2^2}^{-1}(1-P_e)}{2m\ln 2} + o\left(\frac{1}{m}\right) \quad (32)$$

where $F_{\chi_2^2}$ is the CDF for a χ^2-distribution with two degrees of freedom.

As opposed to the IID case, Theorem 1, the upper and lower bounds are not tight as $P_e \to 0$. There is a factor about 2 between the bounds. Fig. 2 shows the different bounds.

Our bottom-line comparison was with universal source coding. The redundancy of universal source coding of a Markov chain with 2 states is about $R_l^+ \approx \frac{\log l}{l}$ [28], a factor 2 increase over IID sources. For the achievable rate we also have about a factor 2 increase, and therefore approximately

$$m \geq \frac{Q^{-1}(P_e/2)^2}{2\ln 2}\frac{l}{\log l},$$

the same as (26). Thus no more samples are required than for the IID case.

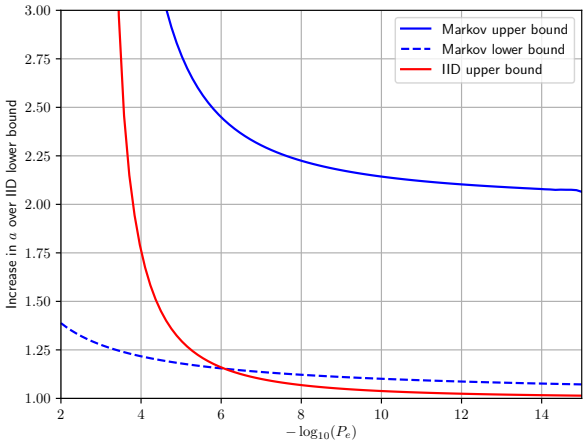

Figure 2. Plot of the ratio between the IID lower bound and the Markov bounds.

IV. Conclusions

The central question of this paper can be thought of as: how much training is required to beat universal source coding? The answer for both IID sources and Markov chains is: not many. To code a sequence of length l the number of training samples is proportional to $\frac{l}{\log l}$. This optimistic conclusion is totally opposite to the pessimistic conclusion of [15]. The reason is due to the viewpoint – and perhaps that we so far only consider very simple sources. While [15] focuses on approaching entropy rate, we just want to beat the redundancy of universal source coding. Additionally, [15] considers learning to be that of building a dictionary, inspired by Lempel-Ziv coding [1], [2]. However, the exceptional performance of modern machine learning can be seen as being achieved through learning soft information.

REFERENCES

[1] J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *Information Theory, IEEE Transactions on*, vol. 23, no. 3, pp. 337 – 343, may 1977.

[2] ——, "Compression of individual sequences via variable-rate coding," *Information Theory, IEEE Transactions on*, vol. 24, no. 5, pp. 530 – 536, sep 1978.

[3] T. Cover and J. Thomas, *Information Theory, 2nd Edition*. John Wiley, 2006.

[4] J. Schmidhuber and S. Heil, "Sequential neural text compression," *IEEE Transactions on Neural Networks*, vol. 7, no. 1, pp. 142–146, 1996.

[5] J.-L. Zhou and Y. Fu, "Scientific data lossless compression using fast neural network," in *ISNN 2006*, 2006, pp. 1293–1298.

[6] A. Kattan, "Universal intelligent data compression systems: A review," in *2010 2nd Computer Science and Electronic Engineering Conference (CEEC)*, Sept 2010, pp. 1–10.

[7] M. V. Mahoney, "Fast text compression with neural networks." in *FLAIRS Conference*, 2000, pp. 230–234.

[8] ——, "Adaptive weighing of context models for lossless data compression," Texas A&M University, Tech. Rep., 2005.

[9] D. Cox, "Syntactically informed text compression with recurrent neural networks," *CoRR*, vol. abs/1608.02893, 2016. [Online]. Available: http://arxiv.org/abs/1608.02893

[10] K. Tatwawadi, "Deepzip: Lossless compression using recurrent networks," Stanford University, Tech. Rep.

[11] G. Toderici, D. Vincent, N. Johnston, S. J. Hwang, D. Minnen, J. Shor, and M. Covell, "Full resolution image compression with recurrent neural networks," *CoRR*, vol. abs/1608.05148, 2016. [Online]. Available: http://arxiv.org/abs/1608.05148

[12] Q. Li and Y. Chen, "Lossy source coding via deep learning," in *2019 Data Compression Conference (DCC)*. IEEE, 2019, pp. 13–22.

[13] Z. Cheng, H. Sun, M. Takeuchi, and J. Katto, "Deep convolutional autoencoder-based lossy image compression," in *2018 Picture Coding Symposium (PCS)*. IEEE, 2018, pp. 253–257.

[14] D. Liu, Y. Li, J. Lin, H. Li, and F. Wu, "Deep learning-based video coding: A review and a case study," *ACM Comput. Surv.*, vol. 53, no. 1, Feb. 2020. [Online]. Available: https://doi.org/10.1145/3368405

[15] Y. Hershkovits and J. Ziv, "On fixed-database universal data compression with limited memory," *IEEE Transactions on Information Theory*, vol. 43, no. 6, pp. 1966–1976, Nov 1997.

[16] G. Shamir, "On the mdl principle for i.i.d. sources with large alphabets," *Information Theory, IEEE Transactions on*, vol. 52, no. 5, pp. 1939–1955, May 2006.

[17] A. Høst-Madsen, "Bounds for learning lossless source coding," *IEEE Transactions on Information Theory*, under preparation, preliminary version available at https://arxiv.org/abs/2009.08562.

[18] N. Merhav and M. Feder, "Universal prediction," *IEEE Transactions on Information Theory*, vol. 44, no. 6, pp. 2124–2147, 1998.

[19] R. E. Krichevskiy, "Laplace's law of succession and universal encoding," *IEEE Transactions on Information Theory*, vol. 44, no. 1, pp. 296–303, Jan 1998.

[20] M. Falahatgar, A. Orlitsky, V. Pichapati, and A. T. Suresh, "Learning markov distributions: Does estimation trump compression?" in *2016 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2016, pp. 2689–2693.

[21] Y. Hao, A. Orlitsky, and V. Pichapati, "On learning markov chains," in *Advances in Neural Information Processing Systems 31*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Curran Associates, Inc., 2018, pp. 648–657. [Online]. Available: http://papers.nips.cc/paper/7345-on-learning-markov-chains.pdf

[22] D. Braess and T. Sauer, "Bernstein polynomials and learning theory," *Journal of Approximation Theory*, vol. 128, no. 2, pp. 187–206, 2004.

[23] M. Mohri, A. Rostamizadeh, and A. Talwalkar, *Foundations of Machine Learning*. MIT Press, 2018.

[24] R. Krichevsky and V. Trofimov, "The performance of universal encoding," *IEEE Transactions on Information Theory*, vol. 27, no. 2, pp. 199–207, Mar 1981.

[25] R. J. Serfling, *Approximation Theorems of Mathematical Statistics*. Wiley-Interscience, 2001.

[26] S. S. Dragomir and V. Gluščevic, "Some inequalities for the kullback-leibler and χ^2-distances in information theory and applications," *Tamsui Oxford Journal of Mathematical Sciences*, vol. 17, no. 2, pp. 97–111, 2001.

[27] M. Short, "Improved inequalities for the poisson and binomial distribution and upper tail quantile functions," *ISRN Probability and Statistics*, vol. 2013, p. 412958, 2013.

[28] J. Rissanen, "Complexity of strings in the class of markov sources," *Information Theory, IEEE Transactions on*, vol. 32, no. 4, pp. 526 – 532, jul 1986.