

Finite Memory Universal Coding of Individual Binary Sequences

Eado Meron and Meir Feder
Department of Electrical engineering
Tel Aviv University, Israel
meir@eng.tau.ac.il
eado@eng.tau.ac.il

Abstract

We consider universal coding of individual binary sequences, with the constraint that the universal coder itself is a K-state time-invariant deterministic finite state (TI-DFS) machine. Actually we consider an equivalent problem of sequentially assigning a probability to the next outcome using a TI-DFS machine. We show that unlike the associated prediction problem, counters and finite window machines perform poorly for malicious sequences. We introduce a simple “exponential decaying memory” machine whose normalized code length is greater than the sequence empirical entropy by an $O(K^{-2/3})$ term. As a lower bound, we show that no K-state TI-DFS machine can achieve a redundancy less than $\Theta(K^{-4/5})$ for all sequences.

I. Problem setting

Imagine an observer receiving an arbitrary deterministic binary sequence and wishing to assign at time t a probability p_t to the event that the next bit will be ‘1’ based on the past. Any encoder that assigns a probability of p_t (written on the current state) pays a code length (or log-loss) of $-\log_2(p_t)$ in case the next bit is actually ‘1’ and $-\log_2(1-p_t)$ in case the next bit is ‘0’. We would like to have a single universal encoder U that competes with the best encoder b of a certain comparison class \mathbf{B} for every x^n in the sense that $\frac{\text{codelength}(U(x^n))}{n}$ is as close as possible to $\min_{b \in \mathbf{B}} \frac{\text{codelength}(b(x^n))}{n}$.

The universal predictor must be the same for every x^n whereas the minimizing predictor b may depend on the entire sequence. The comparison class considered in this paper is the class of single state predictors, i.e., predictors assigning a constant p_t . The best static predictor tuned to the sequence chooses p_t to be the empirical probability; thus the reference code length is the empirical binary entropy of the sequence. This leads to the definition of the universal coding redundancy ([2]):

$$R_U = \max_{x^n} \left(\frac{\text{code length}(U(x^n))}{n} - H_{emp}(x^n) \right)$$

where $H_{emp}(x^n)$ is the binary empirical entropy of a sequence.

II. Candidate machines

In this section we discuss several possible K-state universal machines and analyze their performance.

The saturated up-down counter (SUD): This machine is a counter with K states, that goes up after observing ‘1’ and down after ‘0’. It stays at the top state after ‘1’, and stays at the bottom state after ‘0’. Each state is assigned a fixed probability p that defines the prediction of the machine. In ([1]) it

was shown that the SUD whose assigned probabilities to the states are equally spaced $1/K$ apart between $1/(2K)$ and $1-1/(2K)$ is the optimal machine in the closely related 0-1 loss, achieving a regret of K^{-1} compared to the best single-state predictor.

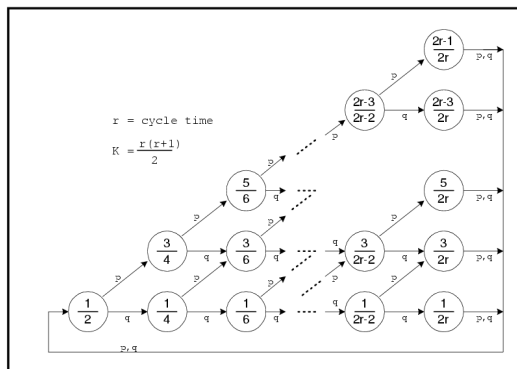
However, in the log-loss case, this predictor performs poorly. The worst-case sequence for the SUD with uniform assigned probabilities is the sequence that begins with $K/2$ ones and continues with the infinite sequence $0101010\dots$. The SUD normalized code length for this sequence approaches $\frac{\log(K)}{2}$ while the empirical entropy approaches 1, causing a redundancy that grows with K .

Modifying the probabilities assigned to the counter states will not help since the counter does not have a “relaxation” feature, meaning that a sequences which starts with consecutive ones/zeros and continues with a balanced sequence will cause the counter to forever toggle around some probability value q that does not match the sequence empirical probability (that is close to $1/2$).

The finite window predictor: This machine keeps track of the last K bits. This requires 2^K states. Suppose even that keeping track only on the number of ones in the window can be done with K states (this can not be done exactly by a K -state machine but e.g., a machine with about K^2 states that approximates this count is given in [5]). Now, at each instant the finite window will assign the probability $\frac{(\#ones\ in\ window)+0.5}{K+1}$, which is the Krichevsky -Trofimov (KT) estimate ([3]), to the next bit.

The worst-case sequence for this finite window predictor toggles between K consecutive ones and K consecutive zeros, leading to a normalized code length that approaches $\log_2 e$. The empirical entropy of this sequence approaches 1, and so for this sequence the redundancy of the ideal K -window machine does not diminish as K becomes larger. Therefore, even though the window does have a relaxation feature it fails to adapt quickly enough to the malicious sequence.

The counter with Reset: The best known TI-DFS machine so far was given in ([4]). This machine is shown in the figure below:



Deterministic machine for the single-state reference setting.

It counts the zero and ones and uses the KT estimates for prediction. It resets itself every \sqrt{K} steps.

As shown in [4] it achieves a redundancy of $\Theta\left(\frac{\log K}{\sqrt{K}}\right)$ for all sequences.

III. The “exponentially decaying memory” (EDM) machine

We now present a novel machine that outperforms the “reset counter” above. The motivation for this machine is to simulate an exponentially decaying memory for the past data. In this EDM machine the

probabilities assigned to the states are based on non-uniform quantization of the probability axis between $K^{-2/3}$ and $1 - K^{-2/3}$ so that the density of states having an assigned probability in the vicinity of p is proportional to $(p(1-p))^{-1/2}$ (this is motivated by Jefferys' prior and follows [4]). Suppose that at time t the machine is at a state with assigned probability x_t (i.e. the machine being at that state predict a probability x_t that the next outcome is '1'). If the next bit is '1' the machine moves to the state whose assigned probability is the closest to $x_t + (1 - x_t)K^{-2/3}$. If the next bit is a '0' the machine shifts to the state whose value is nearest to $x_t - x_t K^{-2/3}$.

Theorem 1: The EDM machine achieves a redundancy of: $O(K^{-2/3})$.

Outline of the proof: Traversing the states as the sequence progresses, we divide uniformly the code length obtained at each step, between the state-gaps that were leaped over during each transition between states. For example, suppose the current state probability is 0.25, the incoming next bit is '1', and suppose that this induces a transition that jumps 10 states upwards. The added code length is 2 bits ($-\log 0.25$), which is divided between the 10 state-gaps, so that each is associated with a code length of 0.2 bits and a 1/10 up-step. Summing along the sequence, each state-gap is associated with an accumulated code length and a number of up- and down-steps. It can be shown that the average code length (code length divided by number of steps) in each state-gap is the binary entropy $H(x)$, within $O(K^{-2/3})$, where x is the center value of that gap. It can also be shown that the ratio between up-steps and down-steps is close to x so that the divergence between this ratio and x is bounded by $O(K^{-2/3})$, and therefore the optimal code length for that gap is $H(x)$ within $O(K^{-2/3})$. Thus the machine is close to optimal for each state-gap, and by Jensen's inequality it is close to optimal for the whole sequence.

IV. Lower bound

Theorem 2: Any TI-DFS machine achieving a redundancy less than M^{-1} for all sequences must have at least $\Theta(M^{5/4})$ states.

Outline of the proof: We describe machine dependent sequences each called "threshold sequence(x)"-TS(x). A TS(x) is a sequence that traverses a given TI-DFS machine as follows: if the probability written on the current state is above x its next outcome is '0' otherwise its next outcome is '1'. This construction causes the machine to circle in a cycle of states. The majority of these states must have probabilities within M^{-1} of the value x . The frequency of ones in the cycle should also be in the vicinity of x . Thus the cycle length should be large enough to provide the necessary precision. Summing these required cycle lengths for M different uniformly-spaced values of x adds up to $\Theta(M^{5/4})$ states.

Remark: The discussion here considers *deterministic* finite state machines. It was conjectured in [4] that the optimal *randomized* finite state machine has an $O(K^{-1})$ redundancy.

References:

- [1] E. Meron and M. Feder "Finite memory universal predictability of binary sequences", submitted to ISIT2003.
- [2] Y. M. Shtar'kov, "Universal sequential coding of single messages," *Problems of Inform. Trans.*, vol. 23, no. 3, pp. 175–186, July/Sept. 1987.
- [3] R. E. Krichevsky, V. K. Trofimov, "The Performance of Universal Encoding", *IEEE Transactions on Information Theory*, VOL. IT-27, No. 2, pp. 199-207, March 1981.
- [4] D. Rajwan, M. Feder, "Universal Finite Memory Machines for Coding Binary Sequences", *Proceedings of the 2000 Data Compression Conference*, Snowbird, Utah, USA, pp. 113–122, March 2000.
- [5] M. Datar, A. Gionis, P. Indyk, R. Motwani: "Maintaining Stream Statistics Over Sliding Windows (Extended Abstract)", in *Proceedings of Thirteenth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'02)*