15-859: Information Theory and Applications in TCS **Spring 2013**

Lecture 5: Universal source coding

Lecturer: Mahdi Cheraghchi *Scribes: Yu Zhao*

5.1 Recap

- KL divergence vs. Chernoff Bound
- Data Processing and Markov Chains
- Fano's inequality
- Asymptoti Equipartition Porperty(AEP) :

$$x_1, \ldots, x_n \text{iid } p(x) \Rightarrow -\frac{\log p(x_1, \ldots, x_n)}{n} \to H(X) \text{in probability}$$

- Typical Sets $A_\epsilon^{(n)} = \{(x_1, \ldots, x_n) | p(x_1, \ldots, x_n) \in 2^{-n(H(X)\pm\epsilon)}\} \Rightarrow |A_\epsilon^{(n)}| \leq 2^{n(H(X)+\epsilon)}$

5.2 Lempel-Ziv Algorithm

The Lempel-Ziv Algorithm is described as below and its main idea is keeping a dictionary D of tokens/words in U^*:

1) Initialize D with elements of U
2) Look for the largest possible token w in D
3) Output the index of w using $\lceil \log |D| \rceil$ bits
4) Remove w from D and replace it by all one-letter extensions of w
5) Until the last symbol is reached, **goto** 2)

Nothing is better than an example. Suppose the universe $U = \{a, b, c\}$ and the string $u_1^6 = aaaccb$. Figure 5.1 shows how the Lempel-Ziv Algorithm modifies the dictionary and gets the outputs.

5.3 How good is it

5.3.1 Upper bound of LZ algorithm

Theorem 5.1 (Upper bound of LZ Algorithm) *Suppose u_1^n is parsed into c_{lz} tokens as $u_1^n = \lambda w_1 \ldots w_{c_{lz}}$ by Lempel-Ziv Algorithm and $\lambda, w_1, \ldots, w_{c_{lz}-1}$ are distinct tokens by construction. Suppose $y_i \in \{0,1\}^*$ is the ouput of the token w_i, then $|y_1^{c_{lz}}| = |y_1 \ldots y_{c_{lz}}|$ is just the length of total output. If we define c^* as the max number of distinct tokens that u_1^n can be parsed into, then we have*

$$\lim_{n\to\infty} \sup \frac{|y_1^{c_{lz}}|}{n} \leq \lim_{n\to\infty} \sup \frac{c^* \log_2 c^*}{n}$$

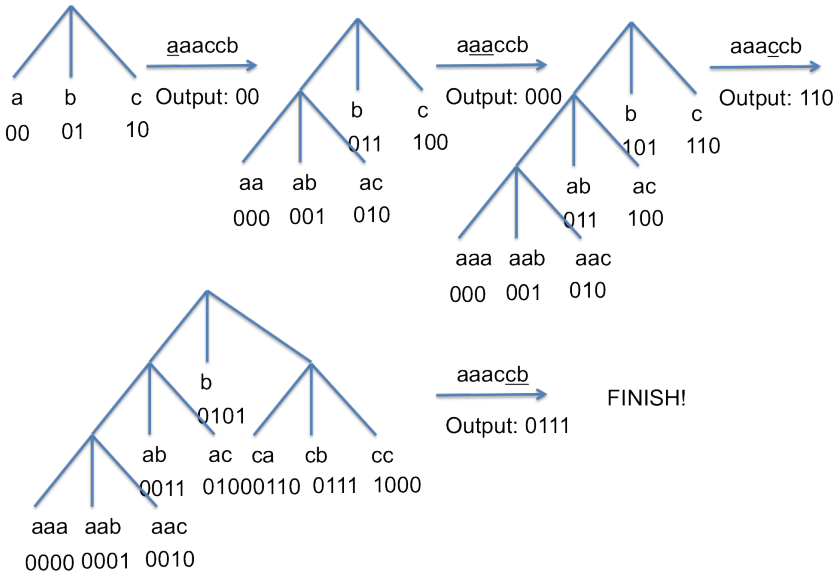

Figure 5.1: An example of Lempel-Ziv Algorithm

Proof: Let $q = |U|$. Assuming that $q \geq 2$ and $c^* \geq 1$, for each token the size of dictionary D increases $(q-1)c_{lz}$,so the size of D in the end is

$$1 + (q-1)c_{lz} \leq 1 + (q-1)c^* \leq qc^*$$

Therefore the length of y_i may be less than $\lceil \log_2(qc^*) \rceil$, so we have

$$|y_1^n| \leq c_{lz} \lceil \log_2(qc^*) \rceil \leq c_{lz} \log_2(2qc^*) \leq c^* \log_2(2qc^*)$$

Therefore we get

$$\lim_{n\to\infty} \sup \frac{|y_1^{c_{lz}}|}{n} \leq \lim_{n\to\infty} \sup \frac{c^*(\log_2 c^* + \log_2(2q))}{n} \leq \lim_{n\to\infty} \sup \frac{c^* \log_2 c^*}{n}$$

■

5.3.2 Comparing with finite state machines

Here we want to show that the lower bound of output length by FSM(finite state machine) is equal to the upper bound of output length by Lempel-Ziv algorithm, which means that Lempel-Ziv algorithm is an optimal. Huffman can be implemented as a FSM and achieves the optimum entropy rate for iid sources (even "ergodic"). We first start from a lemma:

Lemma 5.2 *Let $Z = Z_1 \dots Z_c$, where $Z_i \in U^*$ are distinct and $|U| = q$, then*

$$|Z| > c \log_q \frac{c}{q^3}$$

Proof: We write

$$c = \sum_{k=0}^{m-1} q^k + r$$

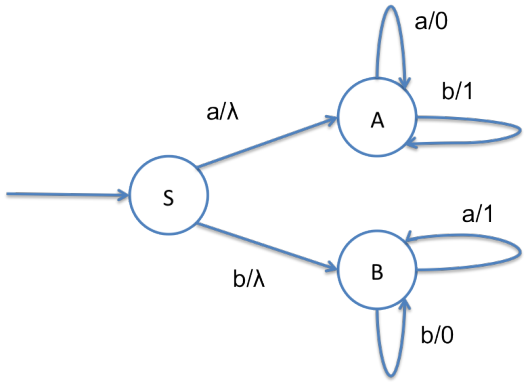

Figure 5.2: An example of FSM which is not uniquely decodable

where $m \geq 0$ and $0 \leq r < q^m$. And we use

$$\sum_{k=0}^{m-1} kq^k = m\frac{q^m}{q-1} - \frac{q}{q-1}\frac{q^m-1}{q-1} = (m - \frac{q}{q-1})\sum_{k=0}^{m-1} q^k + \frac{m}{q-1} = (m - \frac{q}{q-1})(c-r) + \frac{m}{q-1}$$

Since Z is shortest if we let Z_i are as short as possible, we get

$$|Z| \geq \sum_{k=0}^{m-1} kq^k + mr = (m - \frac{q}{q-1})(c-r) + \frac{m}{q-1} + mr \geq (m - \frac{q}{q-1})c > (m-2)c$$

But here we have the upper bound

$$c = \sum_{k=0}^{m-1} q^k + r < \sum_{k=0}^{m} q^k < q^{k+1}$$

Therefore we have $m + 1 > \log_q c$. And finally we get

$$|Z| > (m-2)c > c \log_q \frac{c}{q^3}$$

■

Theorem 5.3 (FSU lower bound) *For any uniquely decodable FSM (Figure 5.2 shows a FSM wich is not uniquely decodable) with s states, if we define c^* as the max number of distinct tokens that u_1^n can be parsed into and $y_i \in \{0,1\}^*$ as the output of FSM after reading u_i, then the output of FSM*

$$|y_1^n| \geq c^* \log_2 \frac{c^*}{8s^2}$$

Proof: Here we define c_{ij} $(i, j \in \{1, \ldots, s\})$ to be the number of words which find FSM in state i and leave it in j. The unique decoding assumes that ouput sequences corresponding to each c_{ij} must be distinct. We define L_{ij} as the total length of encodings of words in c_{ij}. By Lemma 5.2 where $U = 0, 1$ and $|U| = 2$, we have

$$L_{ij} \geq c_{ij} \log_2(\frac{c_{ij}}{8})$$

Therefore the total length

$$|y_1^n| = \sum_{i,j} L_{ij} \geq \sum_{i,j} c_{ij} \log_2(\frac{c_{ij}}{8})$$

Here RHS is a symmetric convex function and $\sum_{i,j} c_{ij} = c^*$, so RHS get the minimum value when $c_{ij} = c^*/s^2$ for all i, j.

$$|y_1^n| \geq \sum_{i,j} c_{ij} \log_2(\frac{c_{ij}}{8}) \geq c^* \log_2 \frac{c^*}{8s^2}$$

■

Suppose we define $\rho_{\text{FSM}}(u_1^n) = |y_1^n|/n$ for a specfic FSM and

$$\rho_s(u_1^n) = \min_{\text{FSM with} \leq s \text{ states}} \rho_{\text{FSM}}(u_1^n)$$

$$\rho_s(u) = \lim_{n\to\infty} \sup \rho_s(u_1^n)$$

$$\rho(u) = \lim_{s\to\infty} \rho_s(u)$$

By Theorem 5.3, we have

$$\rho_s(u) \geq \lim_{n\to\infty} \sup \frac{c^*}{n} \log_2(\frac{c^*}{8s^2}) = \lim_{n\to\infty} \sup \frac{c^* \log_2 c^*}{n} - O(\frac{c^*}{n})$$

But by Lemma 5.2, $n > c^* \log_q(c^*/q^3)$ which means $c^* = O(n/\log n)$. So we have

$$\rho_s(u) \geq \lim_{n\to\infty} \sup \frac{c^* \log_2 c^*}{n}$$

for all s, which means

$$\rho(u) = \lim_{s\to\infty} \rho_s(u) \geq \lim_{n\to\infty} \sup \frac{c^* \log_2 c^*}{n}$$

Therefore the lower bound for any FSM is the upper bound for Lempel-Ziv algorithm, which means Lempel-Ziv algorithm is optimal.