1. Data Compression & Entropy

- Lossless ← ci occupiamo di questa
- Lossy (multimedia)

Osservazione

Non può esistere un super compressore che comprime qualsiasi cosa di un passo

Esempio: prendiamo tutti i file di lunghezza $\leq N$ bit

non è possibile che tutti i file siano compressi a $< N$.

$2 + 4 + 8 + \dots$ $\quad 2^{N} = 2^{N+1} - 2$

TIPI DI COMPRESSORI LOSSLESS

1. SYMBOL SUBSTITUTION (HUFFMAN, MORSE)
2. PARSING (LZ77 LZ78 GZIP, XZ, ...)
3. TRASFORMAZIONI

ESEMPIO TRASFORMAZIONI USATE PER LA COMPRESSIONE

1. MTF MOVE-TO-FRONT

STRINGA DA COMPRIMERE

BACCADACCADACCAAAAAA

LISTA CARATTERI. SI PARTE CON ABCD →

BACD → ABCD → DABC → ADBC → CADB

OUTPUT:

1 1 4 1 4 0 ....

2. DELTA CODING

INVECE DI COMPRIMERE LA SEQUENZA

1001 1005 1007 1010 ....

CONSIDERO IL PRIMO VALORE E LE DIFFERENZE:

1001 4 2 3 ....

3. BWT

LA VEDREMO IN SEGUITO

SYMBOL SUBSTITUTION

CLASSIFICAZIONE DI CODICI BASATI SU SYMBOL SUBSTITUTION

TABLE 5.1 Classes of Codes

X	Singular	Nonsingular, But Not Uniquely Decodable	Uniquely Decodable, But Not Instantaneous	Instantaneous
A	0	0	10	0
B	0	010	00	10
C	0	01	11	110
D	0	10	110	111

CI OCCUPPEREMO PRINCIPALMENTE DI QUELLI ISTANTANEI (PREFIX-FREE)

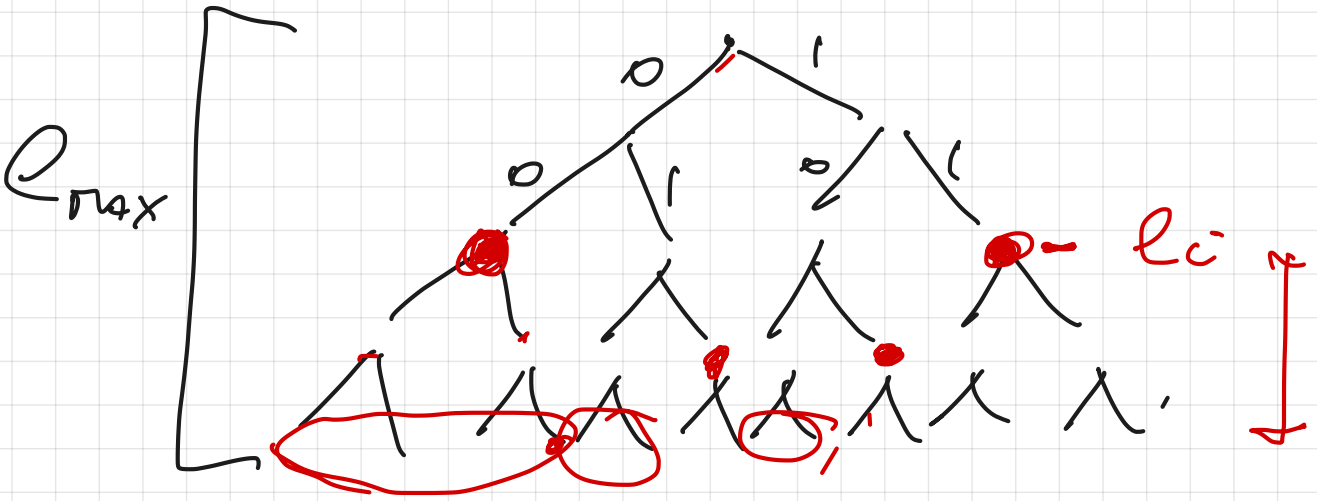
Theorem 5.2.1 (*Kraft inequality*) *For any instantaneous code (prefix code) over an alphabet of size D, the codeword lengths* $l_1, l_2, \ldots, l_m$ *must satisfy the inequality*

$$\sum_i D^{-l_i} \leq 1. \tag{5.6}$$

Conversely, given a set of codeword lengths that satisfy this inequality, there exists an instantaneous code with these word lengths.

DIMOSTRAZIONE KRAFT INEQUALITY

(IDEA) ℓ_{MAX} LUNGHEZZA MASSIMA DEI CODICI



A OGNI CODICE CORRISPONDE UN NODO DELL'ALBERO; ASSEGNO A QUEL CODICE TUTTE LE FOGLIE SOTTO QUEL NODO. OGNI FOGLIA È ASSEGNATA AD AL + UN CODICE

SOMMO IL NUMERO DI FOGLIE ASSOCIATO AD OGNI CODICE

$$\sum_i 2^{\ell_{max} - \ell_i} \leq 2^{\ell_{max}}$$

$$\boxed{\sum_i 2^{-\ell_i} \leq 1}$$ ← KRAFT INEQUALITY

PROBLEMA: STRINGA DI LUNGH N I SINGOLI CARATTERI APPAIONO

$$n_1 \quad n_2 \quad n_3 \cdots n_m$$

VOLTE

$$n_1 + n_2 + n_3 \cdots + n_m = N$$

CONSIDERO TUTTE LE POSSIBILI CODIFICHE, UNA GENERICA CODIFICA USA CODICI DI LUNGHEZZA

$$\ell_1 \quad \ell_2 \quad \ell_3 \cdots \ell_m$$

LA DIMENSIONE DEL FILE CODIFICATO SARA'

$$O = n_1 \ell_1 + n_2 \ell_2 + \cdots + n_m \ell_m$$

HO VINCOLO DELLA DIS. DI KRAFT

$$\sum 2^{-\ell_i} \leq 1$$

PROBLEMA DI OTTIMIZZAZIONE VINCOLATA

RISOLVIAMO CON MOLTIPLICATORI DI LAGRANGE

$$F(\ell_1, \ell_2 .. \ell_m, \lambda)$$

$$= \sum n_i \ell_i + \lambda \left(\sum 2^{-\ell_i} - 1 \right)$$

LA TEORIA DICE CHE I PUNTI DI MIN O MAX SONO DOVE $\partial F = 0$

OSSERVAZIONE

$$\frac{\partial F}{\partial \lambda} = \left(\sum_i 2^{-\varepsilon_i} - 1 \right) = 0$$

QUESTO È IL VINCOLO

$$\frac{\partial F}{\partial \varepsilon_i} = n_i + \lambda \frac{\partial}{\partial \varepsilon_i} 2^{-\varepsilon_i}$$

$$= n_i - \lambda \, 2^{-\varepsilon_i} \cdot \ln 2 = 0$$

$$= \boxed{\lambda 2^{-\varepsilon_i} = \frac{n_i}{\ln 2}}$$

PER OGNI i

$$2^{-\varepsilon_i} = \frac{1}{\lambda} \frac{n_i}{\ln 2}$$

IMPONGO IL VINCOLO:

$$\boxed{\sum 2^{-\varepsilon_i} = 1}$$

OTTENGO:

$$\sum_i \frac{1}{\lambda} \frac{n_i}{\ell_{ge}^2} = 1$$

$$\Rightarrow \quad \sum_i n_i = \lambda \ell_{ge}^2$$

$$\Rightarrow \quad n = \lambda \ell_{ge}^2$$

CHE UNITO A $\boxed{\lambda 2^{-\ell_i} = \frac{n_i}{\ell_{ge}^2}}$

MI DA LA FORMULA PER LA

LUNGHEZZA OTTIMALE DEI CODICI

$$2^{-\ell_i} = \frac{n_i}{n}$$

EQUIVALENTEMENTE

$$\ell_i = -\ell_{g_2} \frac{n_i}{n}$$

LA LUNGHEZZA MINIMA

DELLA CODIFICA DI S

$$\sum_i - n_i \log \frac{n_i}{n} = n \cdot H_0(S)$$

ENTROPIA DI ORDINE $\emptyset$ DI S

$$H_0(S) = \sum_i - \frac{n_i}{n} \log \frac{n_i}{n}$$

DEFINIAMO

$$\frac{n_i}{n} = P_i$$

PROBABILITÀ EMPIRICA DEL SIMBOLO i

ESISTE UNA CATEGORIA DI CODICI BASATI SULLE FREQUENZE $\frac{n_i}{n}$ CHE RIESCONO A COMPRIMERE FINO A

$$n H_0(S) + \epsilon n$$

PER FARE MEGLIO DI H_0 UNO DEI METODI È FARE LA COMPRESSIONE BASATA SUL CONTESTO.

LA COMPRESSIONE CHE POSSIAMO OTTENERE CON CODEWORD CHE DIPENDONO DAL SIMBOLO PRECEDENTE

$$\sum_{i \in A} \sum_{j} - n_{ij} \log \frac{n_{ij}}{n_i}$$

n_{ij} = # DI j SUBITO DOPO LA i

$$= \sum_{i \in A} n_i \sum_{j \in A} - \frac{n_{ij}}{n_i} \log \frac{n_{ij}}{n_i}$$

$$= \sum_{i \in A} n_i \; H_0(S_i)$$

S_i → INSIEME DEI SIMBOLI CHE SEGUONO IMMEDIATAMENTE IL SIMBOLO i

DEFINIAMO

$$H_1(S) = \sum_{i \in A} \frac{n_i}{n} H_0(S_i)$$

L'OUTPUT DEL METODO SOPRA È AL MEGLIO

$$n H_1(S)$$

$H_1(S)$: ENTROPIA DI ORDINE 1 DI S

VALE

$$H_0(S) \leq H_1(S) \leq H_2(S) \leq \cdots$$

METODI DI TIPO PPM

PERMETTONO DI USARE CODEWORD CHE DIPENDONO DAI K SIMBOLI PRECEDENTI (ES $K=4$)

SENZA DOVER TRASMETTERE IN ANTICIPO LE CODEWORD.

MECCANISMO: COSTRUISCO LA TABELLA DELLE FREQUENZE DURANTE LA COMPRESSIONE

ESEMPIO. CONTESTO "COMP" HO VISTO "COMP" 10 VOLTE ED ERA SEGUITO DA

R	5 VOLTE	O
E	3 VOLTE	LO
O	1 VOLTA	LLLO ...
A	1 VOLTA	LLLL ...

DEVE ESISTERE LA CODEWORD PER INDICARE "NUOVO"

COMPRESSORI BASATI SU PARSING

VENGONO CODIFICATE SOTTOSTRINGHE

I METODI DIFFERISCONO PER COME VIENE FATTO IL PARSING

E COME VENGONO CODIFICATE LE SINGOLE FRASI DEL PARSING.

LA FAMIGLIA DI PARSING PIÙ NOTA È QUELLA LZ77 DI CUI ESISTONO NUMEROSE VARIANTI CONSIDERIAMO LA SEGUENTE:

(1) Find the longest prefix $S[i..j]$ of $S[i..n]$ that occurs in S starting before position i.
(2) If $j \geq i$, that is, $S[i..j]$ is nonempty, then the next phrase is $S[i..j]$, and we set $i \leftarrow j + 1$.
(3) Otherwise, the next phrase is the explicit symbol $S[i]$, which has not appeared before, and we set $i \leftarrow i + 1$.
(4) If $i \leq n$, continue forming phrases.

ESEMPIO

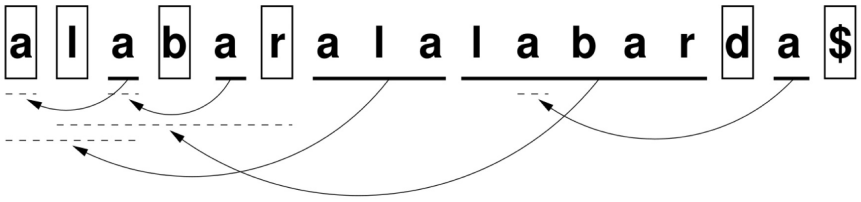


Fig. 2. Lempel-Ziv parse of $S =$ alabaralalabarda$. Each phrase is either an underlined string, which appears before, or a boxed symbol. The arrows go from each underlined string to one of its occurrences to the left (which is underlined with a dashed line).