Comunicaciones de Datos

Facultad de Ciencias Exactas y Naturales y Agrimensura.
Universidad Nacional del Nordeste

Guía Serie de Trabajos Prácticos Nº 1

Teoría de la Información y Codificación

1. Introducción

La Teoría de la Información tiene sus orígenes en la publicación de Claude E. Shannon "Una teoría Matemática de la Comunicación" que trata de los soportes de la información, los símbolos. Los símbolos, deben obedecer ciertas leyes si han de ser capaces de transmitir información.

Nos limitaremos a considerar la información binaria y comenzaremos por la definición de medida de información.

En la Tabla 1 se presenta un ejemplo de representación de información no binaria en función de los dígitos binarios 0 y 1. La correspondencia entre los dígitos decimales y binarios definida por la Tabla 1 constituye un ejemplo de código. Las diez secuencias binarias se denominan *palabras código* y los diez dígitos decimales *símbolos mensaje.*

Tabla 1. Codificación binaria de los dígitos decimales

Dígito decimal	Representación binaria
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

2. Cantidad de Información

La cantidad de información I contenida en un mensaje, es un valor matemático medible referido a la probabilidad p de que una información en el mensaje sea recibida. Entendiendo que el valor más alto de probabilidad se le asigna al mensaje menos probable.

Según Shannon:

$$I = log_2 \frac{1}{p}$$

Cuando se toma logaritmo en base 2 la unidad de información es el *bit.*

Ejemplo 1. Se lanza una moneda al aire y se desea calcular la cantidad de información contenida en los mensajes cara o cruz separadamente.

Los mensajes a representar son cara y cruz.

La probabilidad de obtener cara al lanzar una moneda es:

$$P(cara) = \frac{1}{2}$$

La cantidad de información que representa el mensaje es:

$$I(cara) = log_2 \frac{1}{1/2} = 1\ bit$$

Recordar que:

$$log_a x = \frac{1}{log_b a} log_b x$$

De manera similar se obtiene la cantidad de información que representa el mensaje cruz.

Ejemplo 2. Calcular la cantidad de información necesaria para especificar los mensajes A, B, C, E y F con probabilidades $P(A) = \frac{1}{4}; P(B) = \frac{1}{4}; P(C) = \frac{1}{8}; P(E) = \frac{1}{4}\ y\ P(F) = \frac{1}{8}$.

$I(A) = log_2 \frac{1}{\frac{1}{4}} = 2\ bits$

$I(B) = log_2 \frac{1}{\frac{1}{4}} = 2\ bits$

$I(C) = log_2 \frac{1}{\frac{1}{8}} = 3\ bits$

$I(E) = log_2 \frac{1}{\frac{1}{4}} = 2\ bits$

$I(F) = log_2 \frac{1}{\frac{1}{8}} = 3\ bits$

Ejemplo 3. Teniendo en cuenta los mensajes del ejemplo anterior, calcular la cantidad de información suministrada en el mensaje CAFE.

$$P(CAFE) = P(C) \times P(A) \times P(F) \times P(E) = \frac{1}{8} \times \frac{1}{4} \times \frac{1}{8} \times \frac{1}{4} = \frac{1}{1024}$$

$$I(CAFE) = log_2 \frac{1}{1/1024} = 10\ bits$$

O bien:

$$I(CAFE) = I(C) + I(A) + I(F) + I(E) = 3\ bits + 2\ bits + 3\ bits + 2\ bits = 10\ bits$$

3. Fuente de información de memoria nula

Es interesante y útil describir matemáticamente un mecanismo generador de información. Imaginemos una fuente emitiendo una secuencia de símbolos pertenecientes a un alfabeto finito y fijo:

$$S = \{s_1, s_2, \dots, s_q\}$$

Los símbolos emitidos sucesivamente se eligen de acuerdo a una ley fija de probabilidad. En la fuente más sencilla, admitiremos que los símbolos emitidos son estadísticamente independientes. Tal fuente de información se conoce como *fuente de información de memoria nula*.

4. Entropía

Puede calcularse la información media suministrada por una fuente de información de memoria nula:

$$H(S) = \sum_{i=1}^{q} P(s_i) \times I(s_i)$$

Esta magnitud, cantidad de información por símbolo de la fuente, recibe el nombre de *entropía* de la fuente de información de memoria nula.

Ejemplo 4. Calcular la entropía de la fuente $S = \{s_1, s_2, s_3\}$ con $P(s_1) = \frac{1}{2}; P(s_2) = P(s_3) = \frac{1}{4}$

La entropía de la fuente es:

$$H(S) = \sum_{i=1}^{3} P(s_i) \times I(s_i)$$

Calculamos la cantidad de información de los símbolos:

$$I(s_1) = log_2 \frac{1}{1/2} = 1\ bit$$

$$I(s_2) = I(s_3) = log_2 \frac{1}{1/4} = 2\ bits$$

Resulta, entonces:

$$H(S) = \frac{1}{2} \times 1\ bit + \frac{1}{4} \times 2\ bits + \frac{1}{4} \times 2\ bits = \frac{3}{2}\ bits$$

El valor calculado es un límite teórico, que normalmente no se puede alcanzar. Se puede decir que no existe ninguna codificación que emplee longitudes promedio de mensajes inferiores al número calculado.

El método de *Huffman* que describiremos más adelante, permite obtener codificaciones binarias que se aproximan bastante al óptimo teórico de una forma sencilla y eficiente.

5. Extensiones de orden n **de una fuente de información de memoria nula**

Anteriormente, definimos una fuente de información de memoria nula y consideramos que ésta emitía símbolos aislados, es decir uno a la vez, como en el caso de la fuente S del ejemplo 4. Imaginemos esta misma fuente emitiendo en grupos de a dos símbolos. De esta forma, podemos considerarla como una fuente de nueve símbolos: $s_1s_1, s_1s_2, s_1s_3, s_2s_1, \dots, s_3s_1, s_3s_2$ *y* s_3s_3; a la que llamaremos S^2 o *extensión de orden*

dos de la fuente original. Esta idea puede generalizarse aún más si consideramos la fuente original emitiendo en grupos de a tres símbolos, lo que sería equivalente a una fuente de 27 símbolos, a la que llamaremos S^3*extensión de orden tres de la fuente original*. En general, dada una fuente S con q símbolos, es posible agrupar las salidas en paquetes de n símbolos, con lo que se obtienen q^n secuencias de salidas distintas. En general, llamaremos a esta fuente S^n o *extensión de orden n de la fuente original*.

La entropía de la extensión de orden n de la fuente original se puede calcular:

$$H(S^n) = n \times H(S)$$

Ejemplo 5. Dada la fuente $S = \{a, b\}$ con probabilidades$P(a) = \frac{1}{4}$ y $P(b) = 3/4$ calcular su entropía. Definir una nueva fuente, extensión de orden 2 de la fuente original y calcular su entropía.
La entropía de la fuente original S, es:

$$H(S) = \frac{1}{4} \times 2 + \frac{3}{4} \times 0.41 = 0.81\ bits$$

La extensión de orden 2 de la fuente original es $S = \{aa, ab, ba, bb\}$ con probabilidades $P(aa) = \frac{1}{4} \times \frac{1}{4} = \frac{1}{16}; P(ab) = \frac{3}{16}; P(ba) = \frac{3}{16}; P(bb) = \frac{9}{16}$

La entropía de la fuente S^2, extensión de orden 2 de la fuente original es:

$$H(S^2) = \frac{1}{16} \times 4 + \frac{3}{16} \times 2.41 + \frac{3}{16} \times 2.41 + \frac{9}{16} \times 0.83 = 1.62\ bits$$

O bien:

$$H(S^2) = 2 \times H(S) = 2 \times 0.81\ bits = 0.162\ bits$$

6. Tasa de Información

Se define como *tasa de información*, expresada en bits por segundo (bps), al cociente entre la entropía de la fuente y la duración media de los símbolos que emite:

$$T = \frac{H(S)}{\tau}$$

La duración media de los símbolos (en segundos) se calcula:

$$\tau = \sum_{i=1}^{n} \tau_i \times p_i$$

Dónde:

τ_i es la duración de cada símbolo (en segundos).

p_i es la probabilidad de ocurrencia de cada símbolo.

Ejemplo 6. Calcular la tasa de información de la fuente $S = \{a, b, c\}$ con probabilidades $P(a) = \frac{1}{2}; P(B) = \frac{1}{8}\ y\ P(c) = \frac{3}{8}$ y duraciones $\tau(a) = 0.6\ seg.;\ \tau(b) = 0.8\ seg.\ y\ \tau(c) = 0.4\ seg.$

La tasa de información de la fuente es:

$$T = \frac{H(S)}{\tau}$$

La entropía de la fuente es:

$$H(S) = \frac{1}{2} \times 1\ bit + \frac{1}{8} \times 3\ bits + \frac{3}{8} \times 1.41\ bits = 1.40\ bits$$

La duración media de los símbolos es:

$$\tau = \frac{1}{2} \times 0.6\ seg. + \frac{1}{8} \times 0.8\ seg. \frac{3}{8} \times 0.4\ seg. = 0.55\ seg.$$

Resulta, entonces:

$$T = \frac{1.40\ bits}{0.55\ seg.} = 2.54\ bps.$$

7. Construcción de códigos compactos binarios. Códigos de *Huffman*

El código compacto de una fuente S es el de menor longitud media que se obtiene al codificar los símbolos de uno en uno. En esta apartado se señalará un procedimiento para generar un código compacto binario, problema resuelto por Huffman en 1952.

Consideremos una fuente S con q símbolos, $s_1, s_2, \dots, s_q$ y sus respectivas probabilidades $P(s_1), P(s_2), \dots, P(s_q)$. Supongamos los símbolos ordenados de tal forma que $P(s_1) \geq P(s_2) \geq \cdots \geq P(s_q)$. Combinando los dos últimos símbolos de la fuente S en un único símbolo, se obtiene una nueva fuente con $q - 1$ símbolos, denominada fuente reducida S_1. Los símbolos de la fuente reducida pueden reordenarse, agrupando de nuevo los dos de menor probabilidad para formar una nueva fuente reducida S_2. Continuando de esta forma, se obtendrá una secuencia de fuentes, cada una con un símbolo menos que la anterior, hasta llegar a una fuente reducida de solamente dos símbolos.

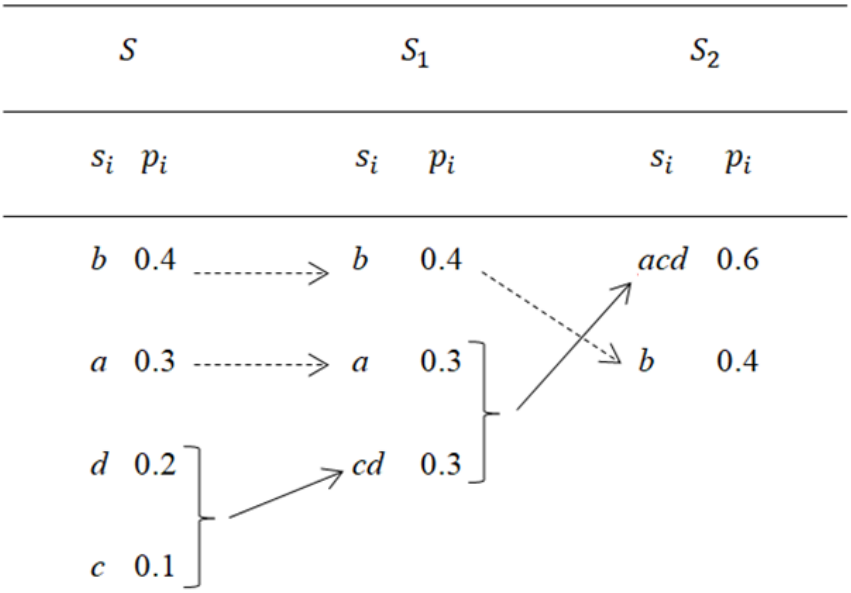
La formación de la secuencia de fuentes reducidas constituye el primer paso en la creación del código compacto correspondiente a la fuente original S. El segundo paso consiste simplemente en observar que el código compacto binario de la última fuente reducida (fuente de dos símbolos) está formado por los códigos 0 y 1. Finalmente, el código compacto de cada una de las fuentes de la secuencia se deduce fácilmente conocido el de la fuente inmediata siguiente. Comenzando por la última fuente y el código compacto hallado, se irá ascendiendo hasta encontrar el código compacto correspondiente a la fuente original.

Ejemplo 7. Obtener un código compacto para la fuente $S = \{a, b, c, d\}$ con probabilidades $P = \{0.3; 0.4;\ 0.1;\ 0.2\}$.

En primer lugar ordenamos los símbolos en orden decreciente de probabilidades: $b\ (0.4); a\ (0.3); d(0.2); c(0.1)$.

En la Tabla 2, se ilustra la secuencia de obtención de fuentes reducidas. Los símbolos *c* y *d* se combinan en un nuevo símbolo *acd*, cuya probabilidad es la suma de probabilidades de los dos símbolos, se obtiene así la fuente reducida S_1, cuyos símbolos se encuentran ordenados en orden decreciente de probabilidades. Los símbolos *cd* y *a* de la fuente reducida S_1 se combinan en un nuevo símbolo *acd* para obtener la fuente reducida S_2 con únicamente dos símbolos.

Tabla 2. Secuencia de obtención de fuentes reducidas



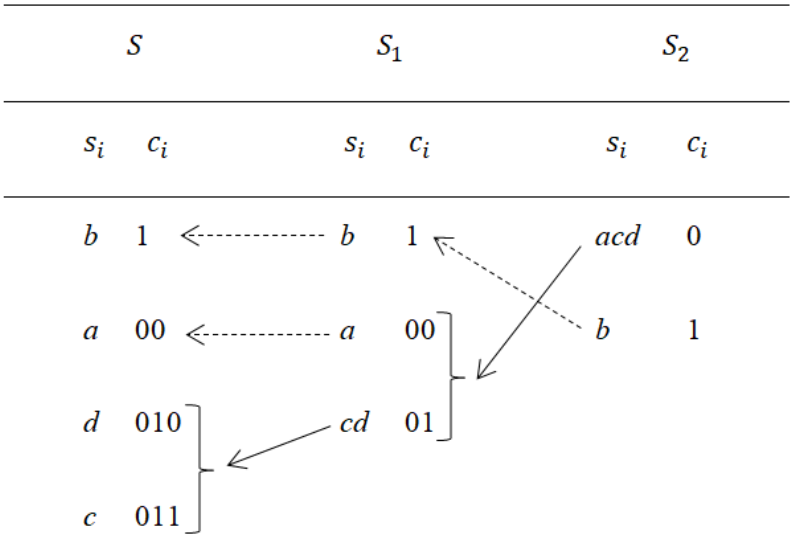
En la Tabla 3, se ilustra el procedimiento de codificación regresiva para obtener el código compacto de la fuente original *S*.

El código compacto de la fuente reducida S_2 (última fuente de dos símbolos) está formado por los códigos 0 y 1. Si observamos los símbolos de la fuente S_2, notaremos que el símbolo *acd* está formado por los símbolos *cd* y *a* de la fuente precedente S_1. El símbolo restante de S_2, en este caso *b*, se identifica con un único símbolo de la fuente precedente S_1. El código compacto correspondiente a la fuente S_1se deduce asignando al símbolo *b* la palabra asignada al mismo símbolo en la fuente S_2. Las palabras correspondientes a los símbolos *a* y *cd* se forman añadiendo 0 y 1 respectivamente a la palabra asignada al símbolo *acd* en la fuente S_2.

De manera similar, se obtiene el código compacto para los símbolos de la fuente original *S*, asignando a los símbolos *b* y *a* de la fuente *S* las palabras asignadas a los mismos símbolos en la fuente S_1. Las palabras correspondiente a los símbolos *c* y *d* se forman añadiendo 0 y 1 respectivamente a la palabra asignada al símbolo *cd* en la fuente S_1.

Es indiferente cuál de cada una de las dos palabras formadas se asigna a cada símbolo de la fuente, lo que significa que la asignación de los símbolos 0 y 1 a las distintas palabras del código se hace de forma completamente arbitraria.

Tabla 3. Codificación regresiva para obtener el código compacto de la fuente original



El código compacto resultante es:

$$C = \{00; 1; 011; 010\}$$

Si se hubiesen codificado los símbolos *acd* y b con las palabras código 1 y 0 respectivamente, se obtendría el siguiente código:

$$C' = \{11; 0; 100; 101\}$$

Es posible obtener otros códigos compactos para la fuente S.

Se define la *longitud media del código* mediante la ecuación:

$$L = \sum_{i=1}^{q} p_i \times l_i$$

Dónde:

p_i es la probabilidad de los símbolos de la fuente S.

l_i longitud de las palabras código.

La longitud media del código $C = \{00; 1; 011; 010\}$; obtenido para la fuente $S = \{a, b, c, d\}$ con probabilidades $P = \{0.3; 0.4;\ 0.1;\ 0.2\}$, del ejemplo 7, es:

$$L = 0.3 \times 2 + 0.4 \times 1 + 0.1 \times 3 + 0.2 \times 3 = 1.9\ bits$$

El *rendimiento del código* se define como:

$$\eta = \frac{H(S)}{L}$$

Si la entropía de la fuente del ejemplo 7 es:

$$H(S) = 1.84\ bits$$

Se puede calcular el rendimiento del código $C = \{00; 1; 011; 010\}$:

$$\eta = \frac{1.84\ bits}{1.90\ bits} = 0.96$$

Según se codifiquen extensiones de mayor orden, el rendimiento se acerca a la unidad.

La *tasa de compresión* se define como la relación de compresión lograda frente a un código de longitud fija:

$$R_{max} = \frac{log_2 q}{H(S)}$$

Dónde:

q es el número de símbolos de la fuente.

La tasa de compresión para el código del ejemplo 7 resulta:

$$R_{max} = \frac{log_2 4}{1.84} = \frac{2}{1.84} = 1.08$$

Bibliografía recomendada

[1] David Luis La Red Martínez. Presentaciones de Clases Teóricas. Comunicaciones de Datos, Facultad de Ciencias Exactas y Naturales y Agrimensura. Universidad Nacional del Nordeste.
[2] N. Abramson. *Teoría de la Información y la Codificación*, 5ta Edición, Parainfo, Madrid, 1981.