ESTUDIO Y CONSTRUCCIÓN DE PLATAFORMA PARA ENVÍO Y RECEPCIÓN DE INFORMACIÓN POR LASER

GRADO EN INGENIERÍA INFORMÁTICA

Práctica 2

Códigos Huffman

Autores
Adrián Sánchez Cerrillo
adrisanchez@correo.ugr.es

Miguel Ángel López Robles
robles2197@correo.ugr.es

Teoría de la Información y la Codificación

Dpto. de Ciencias de la Computación e Inteligencia Artificial

Escuela Técnica Superior de Ingenierías Informática y de Telecomunicación

—

1. **(0,5 puntos) Implemente una función en C que tenga como entrada el nombre de un fichero, y que como salida proporcione un vector con los caracteres ASCII existentes en el fichero, y su probabilidad de ocurrencia, ordenados de mayor a menor probabilidad.**

   Esta función puede encontrarse en nuestro código con el nombre de *loadFrec*, aunque en nuestro caso, no se ordenan por frecuencias, ya que esto se realizará más tarde cuando creamos la lista de símbolos ordenados. Para realizar esta ordenación se ha usado un tipo de dato set con un functor que ordena por frecuencias, de modo que se irá ordenando a la vez que creamos la lista.

2. **(2.5 puntos) Implemente una función que, teniendo como entrada un vector con caracteres ASCII y su probabilidad de ocurrencia, y proporcione como salida un árbol Huffman que implemente un código binario para el alfabeto dado por los caracteres ASCII**

3. **(1.5 puntos) Implemente una función que, teniendo como entrada un árbol Huffman y un carácter ASCII del código que codifica, proporcione como salida una cadena de "0"s y "1"s con el código asociado al carácter.**

4. **(1.5 puntos) Implemente una función que, teniendo como entrada un árbol Huffman y una cadena de "0"s y "1"s, proporcione como salida una cadena de caracteres ASCII decodificados con el árbol Huffman.**

5. **(1 punto) Implemente el programa emisorHuffman, que haga lo siguiente:**
   a. **Lea el fichero "quijote.txt" dado en la práctica, y calcule el árbol Huffman asociado.**
   b. **A continuación, que pida una cadena de caracteres ASCII válida por teclado, la codifique con el árbol Huffman y la envíe por USB a ArduinoEmisor.**

6. **(1 punto) Implemente el programa receptorHuffman, que haga lo siguiente:**
   a. **Lea el fichero "quijote.txt" dado en la práctica, y calcule el árbol Huffman asociado.**
   b. **A continuación, que lea una cadena de "0"s y "1"s por USB desde ArduinoReceptor, la decodifique con el árbol Huffman y muestre los caracteres ASCII resultantes por consola.**

7. **(1 punto) Suponiendo que un laserbit tarda en enviarse U ms., y que la probabilidad de enviar cualquier símbolo del alfabeto de la fuente es equiprobable, responda a las siguientes preguntas:**

   a. **Considerando el código Huffman creado, ¿Cuántos ciclos de U ms serán necesarios, en promedio, para enviar un mensaje compuesto por un símbolo de la fuente? ¿Cuántos ciclos como mínimo? ¿Cuántos ciclos como máximo?**

Lo primero que vamos a calcular es la longitud promedio de los mensajes enviados

$$\bar{\text{и}} = \sum_{i=0}^{N} p(n_i) * |n_i| = 4.107416283$$

Ya tenemos la longitud promedio ahora calcularemos el tiempo promedio ya que no se tarda lo mismo en mandar un 0 que un 1, en nuestro caso U es 3 ms.

$$t = \frac{t_0 + t_1}{2} = \frac{U * 2 + U * 3}{2} = 2.5 * \ Uciclos = 7.5ms$$

Por tanto, ya podemos calcular el tiempo promedio como:

$$T_{promedio} = \bar{\text{и}} * t = 10.26854 * Uciclos = 30.8056ms$$

Ahora vamos a ver cuál sería el tiempo máximo, para ello observamos la codificación de los mensajes y buscamos el más corto y que esté formado por un mayor número de 0. En nuestro caso el carácter ‘E’, el cual se codifica con 011.

$$t_{min} = U * 2 + 2 * U * 3 = 8Uciclos = 24ms$$

En el peor caso debemos ver el carácter más largo y codificado con un mayor número de 1s. En nuestro código el carácter ‘W’ que se codifica como 1011011100011.

$$t_{max} = 5 * 2 * U + 8 * 3 * U = 34Uciclos = 102 \ ms$$

   b. **¿Cuál es la tasa de información transmitida por el canal, en términos reales?**

Para realizar este cálculo vamos a usar la fórmula vista en clase para la tasa de información en términos reales

$$R(S) = k \frac{H(S)}{T}$$

Como podemos ver en primer lugar debemos calcular la entropía del canal, para lo cual debemos conocer la probabilidad de mandar un H o un L. Como ya sabemos, esta probabilidad no es igual para ambos símbolos ya que los 0s se codifican como HL y los 1s como HHL. El problema es que ahora no es equiprobable enviar un 1 que un 0. En nuestro código Huffman debido a que se genera sin tener en cuenta que es más costoso enviar un 1 que un 0, genera un código donde se envían 93 ceros y 111 unos. Por tanto tenemos que la probabilidad de mandar un 0 es p(0) = 0.456 y p(1)=0.544. de modo que podemos calcular las probabilidades de L y H como. Para ello es simple pensar que por cada 204 símbolos que se envíen 93 serán ceros y 111 unos y que por tanto tendremos 204L y 315H. si los dividimos entre el número total de H y L que se habían trasmitido por el canal, tenemos:

$$p(S = L) = \frac{204}{519} = 0.393 \quad P(S = H) = \frac{315}{519} = 0.607$$

Como podemos ver, no hay una gran variación en la probabilidad con respecto a la práctica anterior que sería de

$$p(S = L) = \frac{2}{5} = 0.4 \quad P(S = H) = \frac{3}{5} = 0.6$$

Ahora ya podemos calcular la entropía como

$$H(S) = -\sum_{i=1}^{n} p(s_1) * log_2 (p(s_1)) = 0.9667$$

Con esto solo nos falta los símbolos medios enviados, para lo cual usaremos la longitud promedio de un mensaje del alfabeto. El problema es que ahora cada mensaje del alfabeto del emisor no tiene la misma probabilidad de estar formado por 0 que por 1, por lo que los laserbit necesarios en promedio para enviar un 1 o un 0 no pueden ser 2.5. Ahora el promedio será:

$$laserbit_{promedio} = 0.456 * 2 + 0.544 * 3 = 2.544\, laserbit$$

$$R(S) = k \frac{H(S)}{T} = 4.107 * 2.544 \frac{0.9667}{30.8056ms} = 327.87\, bps$$

c. ¿Cuál es la velocidad de señalización, en términos reales, para enviar un mensaje con un único símbolo del alfabeto de la fuente?

Para enviar un símbolo de la fuente usamos el t promedio en enviar un símbolo del alfabeto y el número de símbolos es la longitud promedio de cada mensaje por el promedio de laserbit necesarios para cada símbolo del mensaje, obteniendo como resultado los laserbit promedio que se envían en un mensaje

$$r = \frac{n}{T_{promedio}} = \frac{4.107 * 2.544}{30.8056ms} = 339.165\ baudios$$

d. ¿Cuántos valores m "0" ó "1" se pueden enviar por unidad de tiempo por el canal, en promedio? ¿Cuál es la velocidad máxima posible para enviar un código "0" o "1" por el canal?

Para calcular la velocidad promedio usamos el tiempo promedio calculado en el apartado a, en enviar un 0 o un 1.

$$v = \frac{1}{t_{promedio}} = \frac{1}{7.5ms} = 133.333\ bps$$

Para calcular la velocidad máxima posible debemos usar, el tiempo en enviar un 0 ya que tarda menos que el 1. Para ello calculamos el tiempo en mandar un único 0

$$t_{min} = 2 * 3 = 6ms$$

$$s = \frac{1}{t_{min}} = \frac{1}{6ms} = 166.666\ bps$$

e. ¿Cuál es la capacidad del canal? ¿Es el canal un canal ideal?

Ahora nos referimos al canal por lo cual estamos hablando del tiempo mínimo en mandar un laserbit. En nuestro caso todos los laserbit tardar lo mismo ya sean H o L. Por tanto, el t mínimo es U.

$$c = \frac{1}{t_{min}} log_2\,(m) = \frac{1}{3ms} * log_2(2) = 333{,}33\ bps$$

Estamos ante un canal sin ruido ya que todos los símbolos emitidos por la fuente se reciben sin distorsión en el receptor. Aunque nuestra implementación no alcanza la capacidad total del canal, es decir R = C, se acerca considerablemente al máximo.

8. **(1 punto) Responda a las siguientes cuestiones:**

a. Siendo X=emisor, calcule la entropía H(X) del emisor, en términos de la probabilidad de enviar H o L por el láser en el canal. ¿Cuántos bits son necesarios, en promedio, para enviar un símbolo del alfabeto de la fuente (caracteres ASCII) por el canal?

Para calcular la entropía, realizamos el mismo calculo explicado en el apartado anterior, donde se tenía en cuenta que no se tiene la misma probabilidad de enviar 0 y 1, y que por tanto esto influye en la probabilidad de enviar H o L. Con las probabilidades de 0 y 1 calculamos además las probabilidades de H o L en nuestro código Huffman obteniendo los siguientes resultados:

$$p(S = L) = \frac{204}{519} = 0.393 \; y \; p(S = H)) \frac{315}{519} = 0.607$$

Ahora ya podemos calcular la entropía en el emisor con la fórmula vista en clase

$$H(S) = -\sum_{i=1}^{n} p(s_1) * log_2 (p(s_1)) = 0.9667$$

Para saber el número de bits que son necesarios en promedio para enviar un símbolo del alfabeto de caracteres ASCII, realizamos el cálculo de la longitud promedio:

$$\bar{n} = \sum_{1}^{N} p(n_i) * |n_i| = 4.107416283 \; bit$$

Este sería el cálculo y la longitud promedio en bit, si queremos saberlo en laserbit, debemos realizar el producto con el promedio de laserbits por bit:

$$4.107416283 \; bit * 2.544 = 10.44923702 \; laserbit \; por \; caracter \; ASCII$$

b. Calcule la matriz de probabilidades conjuntas de enviar o recibir H o L por el canal. Visualizando esta matriz.

X\Y	**H**	**L**	
H	0.607	0	0.607
L	0	0.393	0.393
	0.607	0.393	1

c. Siendo X=emisor e Y=receptor, calcule la entropía conjunto H(X,Y), en términos de enviar y recibir H o L por el canal.

$$H(X,Y) = \sum_{i} \sum_{j} p\ (X = x_i, Y = y_j) \cdot log\ 2\ (p\ (X = x_i, Y = y_j))$$

$$H(X,Y) = -\ 0.607 * \ log_2(0.607) + 0.393 * \ log_2(0.393) = 0.96671$$

d. ¿Cuál sería la entropía de H(Y|X)? ¿Y la información mutua?

Usamos la ley de las entropías totales

H (X, Y) = H (X) + H (Y | X)

H (Y | X) = H (X, Y) - H (X)

H (Y | X) = 0

I (X; Y) = H (Y) - H (Y | X)

Tenemos que calcular H(Y) para poder calcular la información mutua.

H(Y) = $-\sum_{i=1}^{n} p(s_1) * log_2 (p(s_1)) = 0.9667$

I (X ; Y) = 0.9667+ 0 = 0.9667

e. A partir de los resultados de los apartados anteriores, ¿Qué similitudes y diferencias observa con respecto al código uniforme desarrollado en la práctica 1?

Como podemos ver el tener un H(Y|X) = 0 nos está confirmando que tenemos un canal sin ruido, ya que sabiendo lo que enviamos, podemos conocer con certeza lo que se va a recibir. Por último tenemos que I(X; Y) = H(Y) = H(X), lo que nos indica que el emisor y el receptor tienen una correspondencia total, comparten la información máxima. Por tanto, después de todos los cálculos realizados podemos decir que se trata de un canal binario (H o L), sin ruido (determinista y sin pérdida), simétrico e ideal.