Tema 2. Códigos sin prefijos

José A. Montenegro

Dpto. Lenguajes y Ciencias de la Computación
ETSI Informática. Universidad de Málaga
monte@lcc.uma.es

26 de septiembre de 2013

1. El problema de la decodificación
   - Libre de Prefijo
2. Representación códigos mediante árboles
3. El número Kraft-McMillan
   - UD implica $K \leq 1$
   - Principio de Conteo

El problema de la decodificación

Los símbolos en un mensaje aparecen en un orden específico, por lo que podemos establecer que:

Mensaje como una parte de un flujo con un orden establecido por un proceso que ocurre en tiempo real.

$$\xi_1 \xi_2 \xi_3 \ldots$$

- Cada ξ_k es una variable que puede tomar como valor cualquier símbolo del alfabeto S, y su valor actual es el símbolo que ocurre en el tiempo k (k $=$ 1,2,3,...).
- Tenemos una función de codificación $c : S \rightarrow T^*$ que sustituye los símbolos del alfabeto S con una cadena codificada perteneciente al alfabeto T.

Ejemplo 1

Sea $S = \{x, y, z\}$ *y tenemos una cadena inicial:* $yzxxzyxzyy\ldots$
La función de codificación es $c : S \to B^*$ *es definida por las reglas:*

$$x \mapsto 0, y \mapsto 10, z \mapsto 11$$

La codificación es realizada mediante la concatenación de las palabras codificadas para cada uno de los símbolos:

$$10110011100111010\ldots$$

- Sería razonable establecer como requisito que el código c es **unívocamente decodificable**, por lo que cada cadena codificada corresponde únicamente a una cadena original.
- Además es deseable que la cadena sea decodificada de forma secuencial, sin tener que esperar a la *finalización del mensaje*.

Ejemplo 2

Supongamos que recibimos el mensaje codificado: $100011100\ldots$,
y sabemos que la función de codificación es $c : S \to B^*$ *es definida por las reglas:*

$$x \mapsto 0, y \mapsto 10, z \mapsto 11$$

¿Cómo es posible recuperar la cadena original?.

Ejemplo 2

Supongamos que recibimos el mensaje codificado: $100011100\ldots$,
y sabemos que la función de codificación es $c : S \rightarrow B^*$ *es definida por las reglas:*

$$x \mapsto 0, y \mapsto 10, z \mapsto 11$$

¿Cómo es posible recuperar la cadena original?.

<u>*Solución*</u>:
El primer símbolo 1 no está incluido en las codificaciones, por lo que necesitamos el próximo símbolo, 0. La cadena 10 si está incluida (y), por lo que determinamos que el primer símbolo es y. El próximo símbolo 0 es la codificación de x, por lo que establecemos que el próximo símbolo es x.
Continuando de esta forma obtenemos que la cadena original es

$$yxxzy\ldots$$

.

Libre de Prefijo

- Método descodificar para $c : S \to T^*$ es, examinar los símbolos en orden, hasta que una cadena sea reconocida.
- c es una función inyectiva, solamente existe un único símbolo $s \in S$ tal que $c(s) = q$.
- Proceso falla si la palabra codificada q es el prefijo de otra palabra codificada q', es decir, si $q' = qr$ para cualquier palabra no vacía $r \in T^*$. En tal caso, no es posible distinguir entre la palabra q y la parte inicial de q'.

Definición 1 (Libre de Prefijo)

Diremos que un código $c : S \to T^*$ está libre de prefijo (PF) si no existe un par de palabras codificadas $q = c(s), q' = c(s')$ tal que

$$q' = qr, \text{ para alguna palabra no vacía } r \in T^*.$$

Ejemplo 3

Sea $S = \{w, x, y, z\}$ *y define* $c : S \to B^*$ *mediante* $w \mapsto 10$, $x \mapsto 01$, $y \mapsto 11$, $z \mapsto 011$

¿Es el código PF y es UD?

Ejemplo 3

Sea $S = \{w, x, y, z\}$ *y define* $c : S \to B^*$ *mediante* $w \mapsto 10$, $x \mapsto 01$, $y \mapsto 11$, $z \mapsto 011$
¿Es el código PF y es UD?

Solución:
Podemos observar que no es PF, ya que c(x) = 01 es un prefijo de c(z) = 011 y tampoco es UD.
Por ejemplo, la cadena 10011011 corresponde a dos mensajes distintos, $wxwy$ y wzz:

$$wxwy \mapsto 10011011$$
$$wzz \mapsto 10011011$$

Teorema 1

Si un código $c : S \to T^*$ *es libre de prefijo, entonces es unívocamente decodificable.*

Ejercicio 1

Un código libre de prefijo binario es definido como $s_1 \mapsto 00, s_2 \mapsto 010, s_3 \mapsto 100, s_4 \mapsto 111$. *Si la cadena codificada es 1001110100011101010000, ¿Cuál es la cadena original?*

Ejercicio 1

Un código libre de prefijo binario es definido como $s_1 \mapsto 00, s_2 \mapsto 010, s_3 \mapsto 100, s_4 \mapsto 111$. *Si la cadena codificada es 1001110100011101010000, ¿Cuál es la cadena original?*

Solución:

$s_3 s_4 s_2 s_1 s_4 s_2 s_3 s_1$

Ejercicio 1

Un código libre de prefijo binario es definido como $s_1 \mapsto 00, s_2 \mapsto 010, s_3 \mapsto 100, s_4 \mapsto 111$. *Si la cadena codificada es 1001110100011101010000, ¿Cuál es la cadena original?*

Solución:

$s_3 s_4 s_2 s_1 s_4 s_2 s_3 s_1$

Ejercicio 2

La codificación $s_1 \mapsto 10, s_2 \mapsto 010, s_3 \mapsto 100$ *no es unívocamente decodificable, ¿Cúal es la razón?*

Ejercicio 1

Un código libre de prefijo binario es definido como
$s_1 \mapsto 00, s_2 \mapsto 010, s_3 \mapsto 100, s_4 \mapsto 111$. *Si la cadena codificada es 10011101000111010100000, ¿Cuál es la cadena original?*

Solución:

$s_3 s_4 s_2 s_1 s_4 s_2 s_3 s_1$

Ejercicio 2

La codificación $s_1 \mapsto 10, s_2 \mapsto 010, s_3 \mapsto 100$ *no es unívocamente decodificable, ¿Cúal es la razón?*

Solución:

s_1 es prefijo de s_3

Ejercicio 3

Si sustituimos la codificación de $s_2 \mapsto 1$*, demuestra que aunque el nuevo código no es libre de prefijo, es unívocamente decodificable.*

Ejercicio 3

Si sustituimos la codificación de $s_2 \mapsto 1$, demuestra que aunque el nuevo código no es libre de prefijo, es unívocamente decodificable.

<u>*Solución*</u>:
Claramente no es libre de prefijo, ya que 1 es el prefijo de 10 y 100. Si realizamos la codificación desde detrás hacia adelante, el bit es 1 será s_2, si no será s_1 o s_3. Por ejemplo 110101100, es decodificado como $s_2 s_1 s_1 s_2 s_3$.

Representación códigos mediante árboles

- Una forma útil de representar un código es mediante una estructura de árbol. La representación puede realizarse para cualquier tamaño b, pero para simplificar su representación nos centraremos en un alfabeto binario $\mathbb{B}$ con b=2.
- El conjunto de nodos de un árbol binario infinito es representado como $\mathbb{B}^*$, el conjunto de todas las palabras en $\mathbb{B}$.
- La raíz del árbol es etiquetada con la palabra vacía, y los nodos son etiquetados recursivamente. Los nodos hijos del nodo etiquetado como (w) son etiquetados como $w0$ y $w1$.

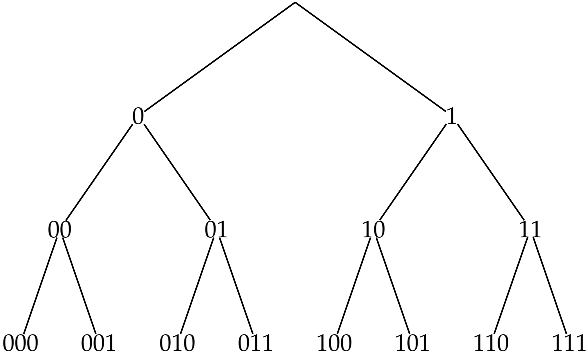


Figura 1 : Los nodos de un árbol binario correspondiente a $\mathbb{B}^3$

Libre de prefijo utilizando árbol

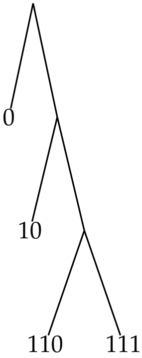
- Partimos de un código $C \subseteq \mathbb{B}^*$, y las palabras codificadas son los nodos del árbol binario.
- Las palabra codificada q es un **prefijo** de otra palabra codificada q' si el único camino de q' a la raíz del árbol pasa a través de q.
- C está **libre de prefijo**, para cada palabra codificada q, ninguno de los descendientes de q puede ser una palabra codificada.
    - Por tanto podemos ignorar todos los nodos que son descendientes de q.
- Si ignoramos todos aquellos nodos que no son palabras codificadas ni prefijos de palabras codificadas, tendremos un árbol binario finito, y las palabras codificadas de C son sus hojas.

Ejemplo 4

Representar el código $C = \{0, 10, 110, 111\}$ mediante un árbol.

Ejemplo 4

Representar el código $C = \{0, 10, 110, 111\}$ mediante un árbol.



Ejercicio 4

Dibuja el árbol que representa el código PF $s_1 \mapsto 00$, $s_2 \mapsto 010$, $s_3 \mapsto 100$, $s_4 \mapsto 111$ *y etiqueta cada hoja con el símbolo correspondiente. ¿Es posible extender este código sin destruir la propiedad PF?*

Ejercicio 5

El alfabeto ternario $T = \{0, 1, 2\}$. *Construya un árbol representando el código ternario con las palabras codificadas 02, 101, 120, 221, 222. ¿Es posible extender este código sin destruir la propiedad PF?*

Ejercicio 6

Tenemos el problema de representar los números 1, 2, 3, 4, 5, 6 mediante un código binario PF, de tal forma que la longitud total de las palabras codificadas sea tan pequeña como sea posible. Construya un código que cumpla las condiciones, utilizando la representación de árbol como guía.

El número Kraft-McMillan

- Dado un código $c : S \to T^*$, sea n_i el número de símbolos en S que son codificados por cadenas de longitud i en T^*.
  - En otras palabras, n_i es el número de $s \in S$ tal que $c(s)$ está en T^i.
- Si M es la longitud máxima de una palabra codificada, nos referimos a los números $n_1, n_2, \ldots, n_M$ como los **parámetros de c**.
- En los diagramas de árbol, n_i es el número de palabras codificadas en el nivel i.
- Cuando $|T| = b$ el número de cadenas de longitud i en T^* es $|T^i| = b^i$.

- Por definición, un código $c : S \to T^*$ es una inyección, por lo que los parámetros deben satisfacer $n_i \leq b^i$.
- La fracción $\frac{n_i}{b^i}$ representa la proporción de las palabras de longitud i que son utilizadas como palabras codificadas.

Definición 2 (Número Kraft-McMillan)

El número Kraft-McMillan asociado con los parámetros $n_1, n_2, \ldots, n_M$*, y la base* b *es definida como:*

$$K = \sum_{i=1}^{M} \frac{n_i}{b^i} = \frac{n_1}{b^1} + \frac{n_2}{b^2} + \ldots + \frac{n_M}{b^M}$$

Por ejemplo si $n_1 = 1, n_2 = 2, n_3 = 1$ y b =2, entonces

$$\mathrm{K} = 1/2 + 2/4 + 1/8 = 9/8.$$

Estableceremos dos importantes resultados sobre la existencia de códigos b-arios.

- Si $K \leq 1$ para un conjunto de parámetros $n_1, n_2, \ldots, n_M$ entonces existe un código PF b-ario con esos parámetros.
- Los parámetros de un código UD b-ario deben satisfacer la condición $K \leq 1$.

Tomando estas dos consideraciones, tenemos que si tenemos un código UD con ciertos parámetros, entonces hay un código PF con los mismos parámetros.

Ejemplo 5

Construya un código binario con parámetros $n_2 = 2, n_3 = 3, n_4 = 1.$

Ejemplo 5

Construya un código binario con parámetros $n_2 = 2, n_3 = 3, n_4 = 1.$

Solución:
Primero verificamos que la condición $K \leq 1$ sea satisfecha:

$$\mathsf{K} = 2/4 + 3/8 + 1/16 = 15/16.$$

- Para construir el código, ya que $n_2 = 2$, comenzamos con dos palabras codificadas de longitud 2, 00 y 01.
- La condición PF significa que no podemos utilizar ninguna palabra de longitud 3 de la forma $00*$ o $01*$, pero nos quedan otras cuatro posibles palabras, de la forma $1 * *$.
- En realidad solo necesitamos tres de las cuatro, por lo que escogemos 100,101 y 110 por ejemplo.
- Para finalizar, necesitamos solamente una palabra de longitud 4, la cual puede ser o 1110 o 1111.

Teorema 2

Si $K \leq 1$ para los parámetros $n_1, n_2, \ldots, n_M$ entonces existe un código PF b-ario con esos parámetros.

Ejercicio 7

Construya códigos binarios PF con los siguientes parámetros:

- $n_2 = 1, n_3 = 4, n_4 = 3;$
- $n_1 = 1, n_2 = 0, n_3 = 2, n_4 = 3, n_5 = 2.$

Teorema 2

Si $K \leq 1$ *para los parámetros* $n_1, n_2, \ldots, n_M$ *entonces existe un código PF b-ario con esos parámetros.*

Ejercicio 7

Construya códigos binarios PF con los siguientes parámetros:

- $n_2 = 1, n_3 = 4, n_4 = 3;$
- $n_1 = 1, n_2 = 0, n_3 = 2, n_4 = 3, n_5 = 2.$

Solución:
Primer caso tendríamos: 00, 010, 011, 100, 101, 1100, 1101, 1110, 1111
Segundo caso tenemos: 0, 100, 101, 1100, 1101, 1110, 11110, 11111

Ejercicio 8

Basándonos en el teorema 2, que podemos decir sobre la existencia de códigos ternarios PF (b=3) con los siguientes parámetros:

- $n_1 = 0, n_2 = 1, n_3 = 12;$
- $n_1 = 0, n_2 = 1, n_3 = 12, n_4 = 40$

Ejercicio 8

Basándonos en el teorema 2, que podemos decir sobre la existencia de códigos ternarios PF (b=3) con los siguientes parámetros:

- $n_1 = 0, n_2 = 1, n_3 = 12;$
- $n_1 = 0, n_2 = 1, n_3 = 12, n_4 = 40$

Solución:
Primer caso: $\frac{0}{3} + \frac{1}{3^2} + \frac{12}{3^3} = \frac{15}{27} \leqslant 1$
Segundo caso: $\frac{0}{3} + \frac{1}{3^2} + \frac{12}{3^3} + \frac{40}{3^4} = \frac{85}{81} > 1$

Ejercicio 8

Basándonos en el teorema 2, que podemos decir sobre la existencia de códigos ternarios PF (b=3) con los siguientes parámetros:

- $n_1 = 0, n_2 = 1, n_3 = 12;$
- $n_1 = 0, n_2 = 1, n_3 = 12, n_4 = 40$

Solución:
Primer caso: $\frac{0}{3} + \frac{1}{3^2} + \frac{12}{3^3} = \frac{15}{27} \leqslant 1$
Segundo caso: $\frac{0}{3} + \frac{1}{3^2} + \frac{12}{3^3} + \frac{40}{3^4} = \frac{85}{81} > 1$

Ejercicio 9

Si la conclusión del anterior ejercicio es que debe existir un código PF, construya uno.

UD implica $K \leq 1$

- Los parámetros de un código UD (n_k) deben satisfacer $K \leq 1$.
- Sea $c : S \to T^*$ un código con parámetros $n_1, n_2, \ldots, n_M$.
- M es la longitud de la palabra codificada más larga para un único símbolo, la máxima longitud del código para una cadena de r símbolos es rM.
- Dado $r \geq 1$, sea $q_r(i)$ el número de cadenas de longitud r que son codificadas por cadenas de longitud i $(1 \leq i \leq rM)$, en particular $q_1(i) = n_i$.
- Mostraremos a continuación que los parámetros n_i determinan el número de $q_r(i)$, para todos los r.

Ejemplo 6

Supongamos que $c : S \to T^*$ *es un código con parámetros* $n_1 = 2$, $n_2 = 3$.
¿Cuales son los valores para $q_2(i)$*?*

Ejemplo 6

Supongamos que $c : S \to T^*$ es un código con parámetros $n_1 = 2$, $n_2 = 3$. ¿Cuales son los valores para $q_2(i)$?

Solución:

- Sea xy una cadena de longitud 2 en S. Ya que c(x) y c(y) tienen o longitud 1 o 2, c(xy) tendrá una longitud 2,3, o 4.
- Si c(xy) tiene longitud 2, c(x) y c(y) deben tener ambos longitud 1. Ya que $n_1 = 2$, hay 2×2 cadenas posibles xy, Por tanto $q_2(2) = 4$
- Si c(xy) tiene longitud 3, c(x) debería tener longitud 1 o 2, y c(y) debería tener longitud 2 o 1, respectivamente. Debido a que $n_1 = 2$ y $n_2 = 3$, hay $2 \times 3 + 3 \times 2$ posibles cadenas xy. Por tanto $q_2(3) = 6 + 6 = 12$
- Si c(xy) tiene longitud 4, c(x) y c(y) deben tener ambos longitud 2. Ya que $n_2 = 3$, existe 3×3 posibles cadenas xy, por tanto $q_2(4) = 9$.

Definiremos la función de generación para la secuencia $q_r(1)$,$q_r(2)$,... $q_r(M)$ como sigue:

$$Q_r(x) = q_r(1)x + q_r(2)x^2 + \ldots + q_r(rM)x^{rM}.$$

Lema 1

Para todos $r \geq 1$, $Q_r(x) = Q_1(x)^r$

Teorema 3

Si existe un código b-ario univocamente decodificable, con una serie de parámetros, entonces su Kfraft-McMillan número satisface que $K \leq 1$

Teorema 3 puede ser combinado con el teorema 2 de la siguiente forma:

$$\text{código UD existe} \Longrightarrow K \leqslant 1 \Longrightarrow \text{código PF existe}$$

la existencia un código UD con unos parámetros dados implica la existencia de un código PD con los mismos parámetros.

Existe un código UD con ciertos parámetros sii existe un código PF con los mismos parámetros y esto sucede sii $K \leqslant 1$.

Ejercicio 10

Supongamos que queremos construir un código UD para 12 símbolos utilizando palabras binarias que no excedan de longitud 4. Establezca una lista de todos los conjuntos de parámetros n_1, n_2, n_3, n_4 *para el cual existe un código apropiado.*

Ejercicio 10

Supongamos que queremos construir un código UD para 12 símbolos utilizando palabras binarias que no excedan de longitud 4. Establezca una lista de todos los conjuntos de parámetros n_1, n_2, n_3, n_4 para el cual existe un código apropiado.

Solución:
Los parámetros n_1, n_2, n_3, n_4 deben satisfacer:

$$n_1 + n_2 + n_3 + n_4 = 12 \; n_1/2 + n_2/4 + n_3/8 + n_4/16 \leqslant 1$$

Despejando n_4 tenemos que $7n_1 + 3n_2 + n_3 \leqslant 4$, por tanto $n_1 = 0$ y $n_2 \leqslant 1$.
Ahora podemos establecer una lista de posibilidades:

n_2 :	1	1	0	0	0	0	0
n_3 :	1	0	4	3	2	1	0
n_4 :	10	11	8	9	10	11	12

Ejercicio 11

Sea $S = \{s_1, s_2, \ldots, s_6\}$ *y T* $=$ $\{$*a,b,c*$\}$, *y tenemos un código definido por* $s_1 \mapsto a$, $s_2 \mapsto ba$, $s_3 \mapsto bb$,$s_4 \mapsto bc$, $s_5 \mapsto ca$,$s_6 \mapsto cb$.
Establezca la función generadora $Q_1(x)$ *y* $Q_2(x)$

Ejercicio 11

Sea $S = \{s_1, s_2, \ldots, s_6\}$ *y* $T = \{a,b,c\}$*, y tenemos un código definido por* $s_1 \mapsto a$, $s_2 \mapsto ba$, $s_3 \mapsto bb$, $s_4 \mapsto bc$, $s_5 \mapsto ca$, $s_6 \mapsto cb$.
Establezca la función generadora $Q_1(x)$ *y* $Q_2(x)$

Solución:
Tenemos $n_1 = 1, n_2 = 5$, por lo que $Q_1(x) = x + 5x^2$ y por ende $Q_2(x) = x^2 + 10x^3 + 25x^4$

Ejercicio 11

Sea $S = \{s_1, s_2, \ldots, s_6\}$ y T = $\{$a,b,c$\}$, y tenemos un código definido por $s_1 \mapsto a$, $s_2 \mapsto ba$, $s_3 \mapsto bb$,$s_4 \mapsto bc$, $s_5 \mapsto ca$,$s_6 \mapsto cb$.
Establezca la función generadora $Q_1(x)$ y $Q_2(x)$

Solución:
Tenemos $n_1 = 1, n_2 = 5$, por lo que $Q_1(x) = x + 5x^2$ y por ende $Q_2(x) = x^2 + 10x^3 + 25x^4$

Ejercicio 12

En el ejercicio anterior, ¿Qué significa el coeficiente x^4 en $Q_2(x)$?

Ejercicio 11

Sea $S = \{s_1, s_2, \ldots, s_6\}$ y T = $\{$a,b,c$\}$, y tenemos un código definido por $s_1 \mapsto a$, $s_2 \mapsto ba$, $s_3 \mapsto bb$,$s_4 \mapsto bc$, $s_5 \mapsto ca$,$s_6 \mapsto cb$.
Establezca la función generadora $Q_1(x)$ y $Q_2(x)$

Solución:
Tenemos $n_1 = 1, n_2 = 5$, por lo que $Q_1(x) = x + 5x^2$ y por ende $Q_2(x) = x^2 + 10x^3 + 25x^4$

Ejercicio 12

En el ejercicio anterior, ¿Qué significa el coeficiente x^4 en $Q_2(x)$?

Solución:
El coeficiente x^4 en $Q_2(x)$ es el número de palabras S de longitud 2 que son representadas por palabras T de longitud 4. Esas 25 palabras son todas las palabras $s_i s_j$ con i,j $\in \{2, 3, 4, 5, 6\}$.

Ejercicio 13

Sea $S = \{a, b, c, d, e, f, g\}$ *y tenemos un código binario definido como* $a \mapsto 00$*,* $b \mapsto 010$*,* $c \mapsto 011$*,*$d \mapsto 1000$*,* $e \mapsto 1001$*,*$f \mapsto 1101$*,*$g \mapsto 1111$*.*
Establezca la función generadora $Q_1(x)$ *y* $Q_2(x)$

Ejercicio 13

Sea $S = \{a, b, c, d, e, f, g\}$ *y tenemos un código binario definido como* $a \mapsto 00$*,* $b \mapsto 010$*,* $c \mapsto 011$*,*$d \mapsto 1000$*,* $e \mapsto 1001$*,*$f \mapsto 1101$*,*$g \mapsto 1111$*.*
Establezca la función generadora $Q_1(x)$ *y* $Q_2(x)$

Solución:
Tenemos $n_2 = 1, n_3 = 2, n_4 = 4$, por lo que $Q_1(x) = x^2 + 2x^3 + 4x^4$ y por ende $Q_2(x) = x^4 + 4x^5 + 12x^6 + 16x^7 + 16x^8$

Principio de Conteo

Teorema 4 (Principio de Conteo)

Sea $c : S \to T^*$ un código que para todo $s \in S$, la longitud c(s) no es mayor que M. Dado $r \geq 1$, sea $q_r(i)$ el número de cadenas de longitud r en S que son codificados por cadenas de longitud i en T ($1 \leq i \leq rM$). Sea $Q_r(x)$ la función de generación

$$Q_r(x) = q_r(1)x + q_r(2)x^2 + \ldots + q_r(rM)x^{rM}.$$

Entonces $Q_r(x) = Q_1(x)^r$.

José A. Montenegro Montes
Dpto. Lenguajes y Ciencias de la Computación
ETSI Informática. Universidad de Málaga
monte@lcc.uma.es

twitter

UNIVERSIDAD DE MÁLAGA

E.T.S. INGENIERÍA INFORMÁTICA

LENGUAJES Y CIENCIAS DE LA COMPUTACIÓN UNIVERSIDAD DE MÁLAGA