

Datenkomprimierung

Huffman-Coding

gehalten von Axel Fischer

Seminar für Algorithmen

1. Allgemeines

- David A. Huffman
- Definition Datenkomprimierung
- Datenkomprimierungsprozess
- Datenkomprimierungstypen
- Entropy encoding

2. Huffman-Coding

- Präfix-Code
- Optimaler Präfix-Code
- Generierung optimaler Präfix-Codes
- Kodierung einer Zeichenfolge
- Die Aha-Effekte

David A. Huffman (09.08.1925 – 07.10.1999)

Pionier in den Computerwissenschaften

Allgemein bekannt durch das Huffman-Coding

Weitere wichtige Beiträge für - finite state machines
- switching circuits
- synthesis procedures
- signal design
geliefert.

seit 1953 arbeitete er am MIT in Boston

1967 wechselte er zur Universität von Kalifornien, Santa Cruz

Im Bereich der Computerwissenschaften stellt die Datenkomprimierung einen Prozess zur Kodierung von Informationen dar unter Benutzung von weniger Bits.

Komprimierung von Informationen ist deswegen möglich, da die meisten *realen* Informationen redundant sind bzw. in einer vom Menschen interpretierbaren Form in nicht präziser Weise dargestellt werden.

Def. Redundanz: Das mehrfache Vorhandensein ein und derselben Information.



Unter Verwendung eines Algorithmus, wird eine Kodierungsmatrix erstellt, dem die Kodierung zu Grunde liegt

Die komprimierte Datei enthält selbst die Informationen, um wieder dekomprimiert zu werden.

(Dekodierungsmatrix = reverse Kodierungsmatrix)

Information $X' = \text{Information } X$

Verlustfreie Datenkomprimierung
(lossless data compression)

Information $X' \neq \text{Information } X$

Verlustbehaftete Datenkomprimierung
(lossy data compression)

Verlustfreie Komprimierung (lossless data compression)

Die dekodierte Information ist
Identisch mit dem Original.

Wird benötigt wenn keine
Informationen des Originals verloren
gehen oder verändert werden dürfen.

Zip (Winzip,gzip2,bzip)

Rar (WinRAR)

Ace (WinACE)

Verlustbehaftete Komprimierung (lossy data compression)

Im Encoding-Prozess gehen Teile der
Originalinformationen verloren.
Die dekodierte Information ist nicht
mehr identisch mit dem Original

Findet Anwendung in der Komprimierung
von Bildern, Sound und Filmen

Bilder: JPEG

Filme: MPEG-1,-2,-4, DivX, XviD

Sound: Dolby AC3

MP3

WMA

OGG Vorbis

... um nur einige aufzuzählen

Verlustfreie Komprimierung (lossless data compression)

- Run-length coding
- Dictionary coders
- Burrows-Wheeler transform
- Entropy encoding

Verlustbehaftete Komprimierung (lossy data compression)

- Discrete cosine transform
- Fractal compression
- Wavelet compression

Def. Entropie (Informationstheorie)

Die Entropie als Begriff in der Informationstheorie ist in Analogie zur Entropie in der Thermodynamik und Statistischen Mathematik zu verstehen.

Der Begriff geht auf Claude Shannon zurück

Er definierte die Entropie H einer gegebenen Information I durch

$$H(I) = \sum_{i=1}^n p_i \log_2 p_i$$

wobei p_i die Wahrscheinlichkeit ist, mit der das i -te Symbol des Informationstextes in I auftritt.

H multipliziert mit der Anzahl der Zeichen im Informationstext gibt dann die mindestens notwendige Anzahl von Bits an, die zur Darstellung der Information notwendig sind.

Der Aha-Effekt kommt später ...

Kodierung von Text-Dateien erfolgt in ASCII-Code (8 Bits / Zeichen) oder Unicode (16 Bits / Zeichen).

Alle Zeichen werden mit der gleichen Anzahl von Bits kodiert

⇒ Fixed-length code

Beim Entropy-Encoding werden den Zeichen Codes zugeordnet, bei denen die Codelänge mit der Häufigkeit des Auftretens dieses Zeichens korreliert.

⇒ Variable-length code

Optimale Codelänge (nach Shannon) für ein Symbol : $-\log_b P$

b stellt die Anzahl von Symbolen zur Erstellung der Codes und P die W'keit des Zeichen in der Zeichenfolge dar.

Beim Huffman-Coding wird die Kodierung mit optimaler Länge der Einzelcodes durchgeführt.

Die Grundlage zur Optimierung der Einzelcodelänge ist die unterschiedliche Häufigkeit der zu kodierenden Symbole

In diesem Zusammenhang spricht man auch vom Präfix-Code

Ein Präfixcode ist eine Menge von Codes in denen kein Code Anfang (Präfix) eines anderen ist.

0; 11 ; 101; 100 -> Präfixcode

0; 11 ; 101; 110 -> kein Präfixcode

Optimaler Präfixcode:

Ein Präfixcode $c(a_n)$ für Alphabet $\{a_1, a_2, \dots, a_n\}$, die jeweils mit der Häufigkeit $p(a_n)$ in einem Text vorkommen, gilt dann als optimal, wenn die mittlere Codewortlänge

$$\sum_{i=1}^n p(a_i) \cdot |c(a_i)|$$

minimal ist.

Auch dazu kommt der Aha-Effekt später ...

Gegebenen sei die Zeichenfolge **ataatactactag**

Zur Erstellung des optimalen Präfix-Codes werden die Häufigkeiten benötigt

	a	t	c	g
Häufigkeit	6	4	2	1
W'keit	0,46	0,31	0,15	0,08

0,46

a

0,31

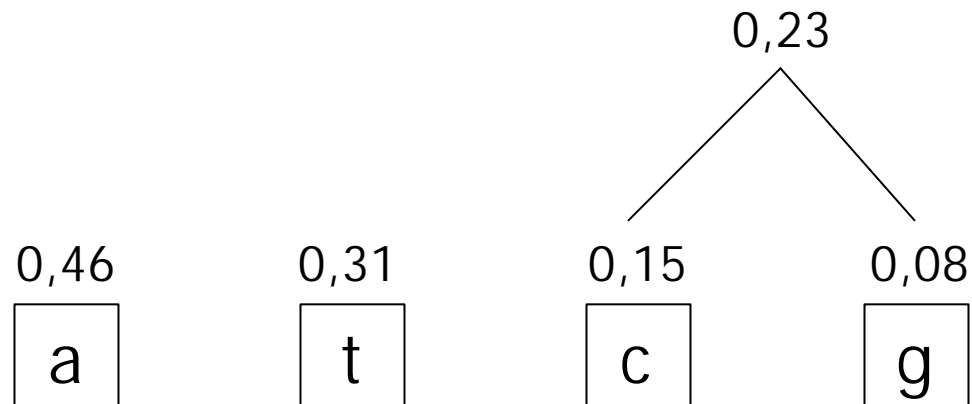
t

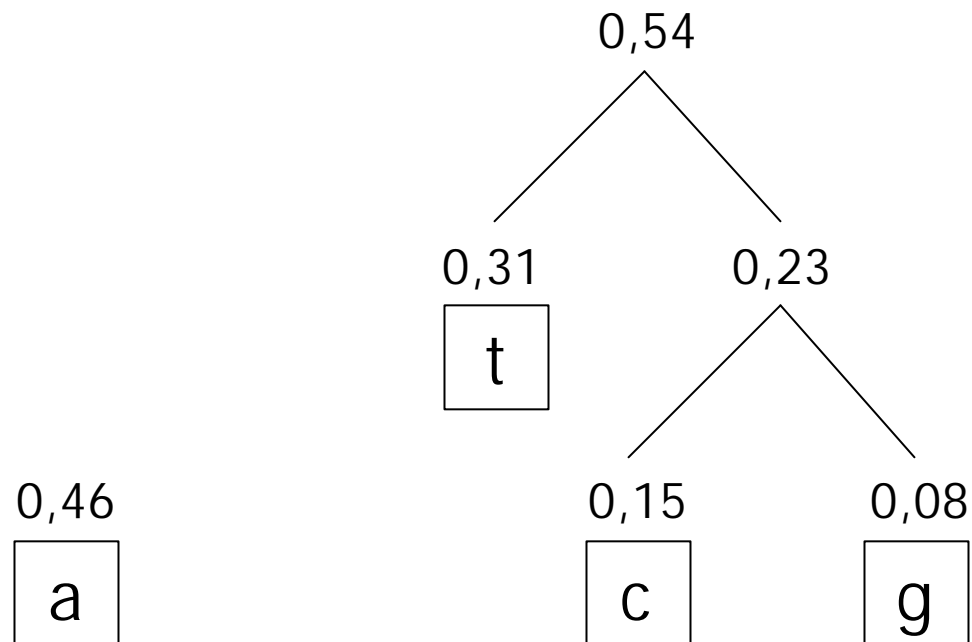
0,15

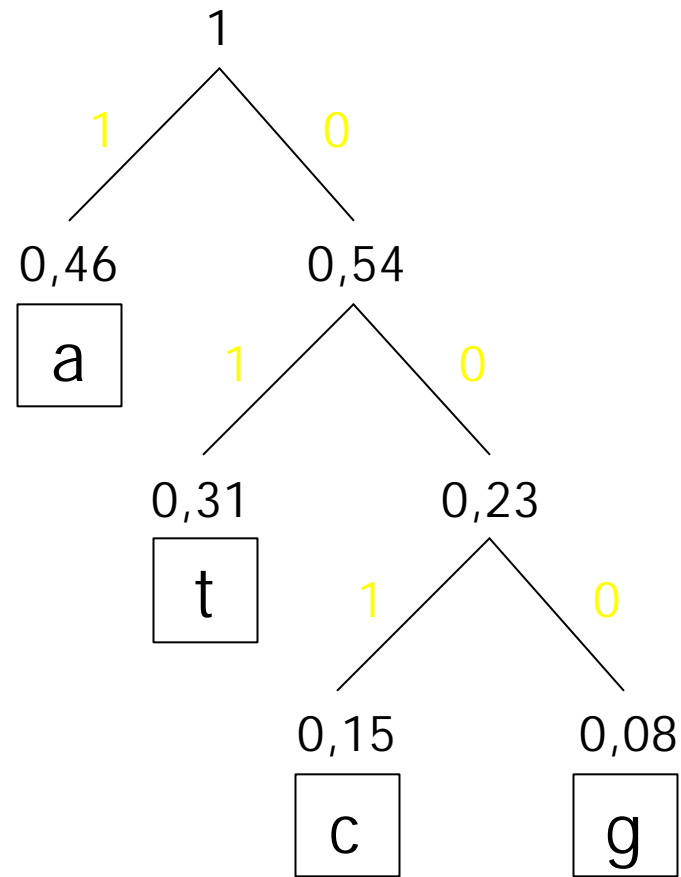
c

0,08

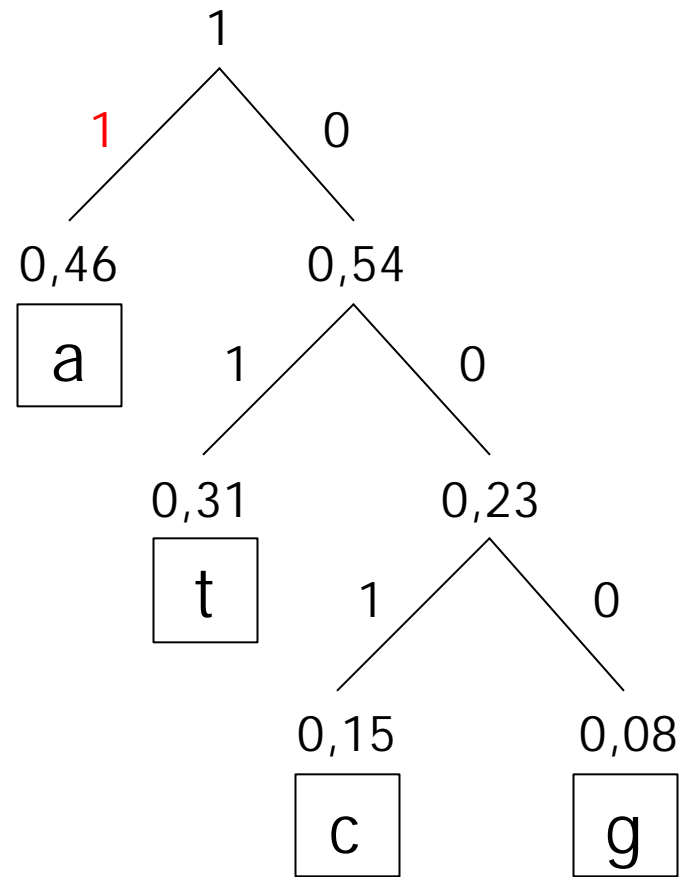
g





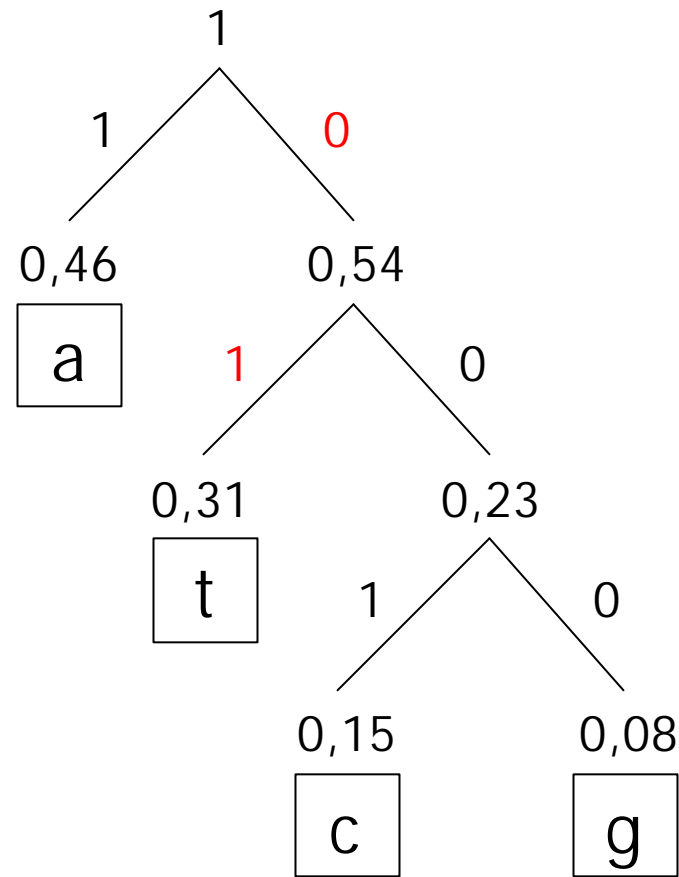


a = 1



a = 1

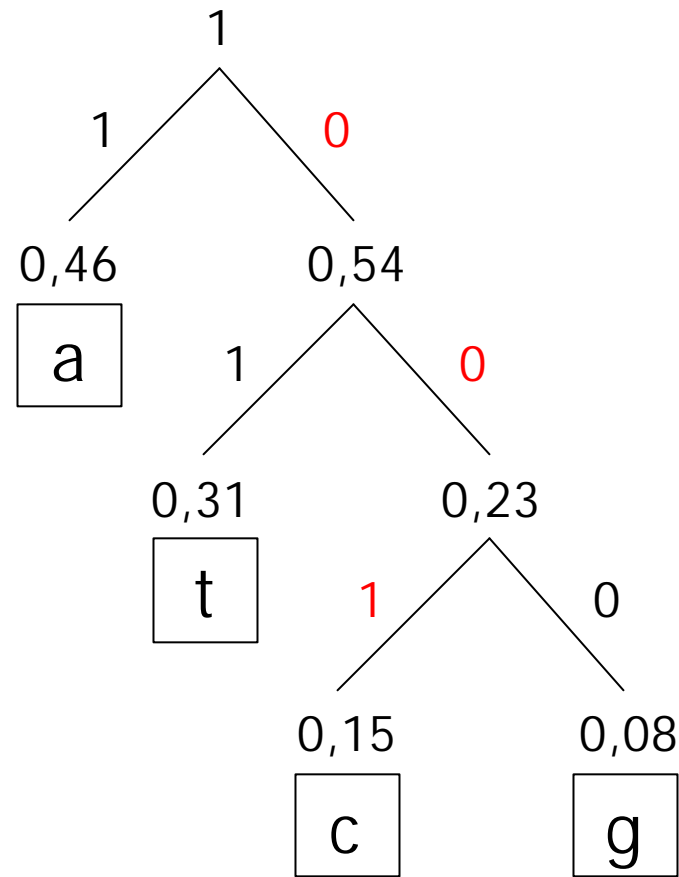
t = 01

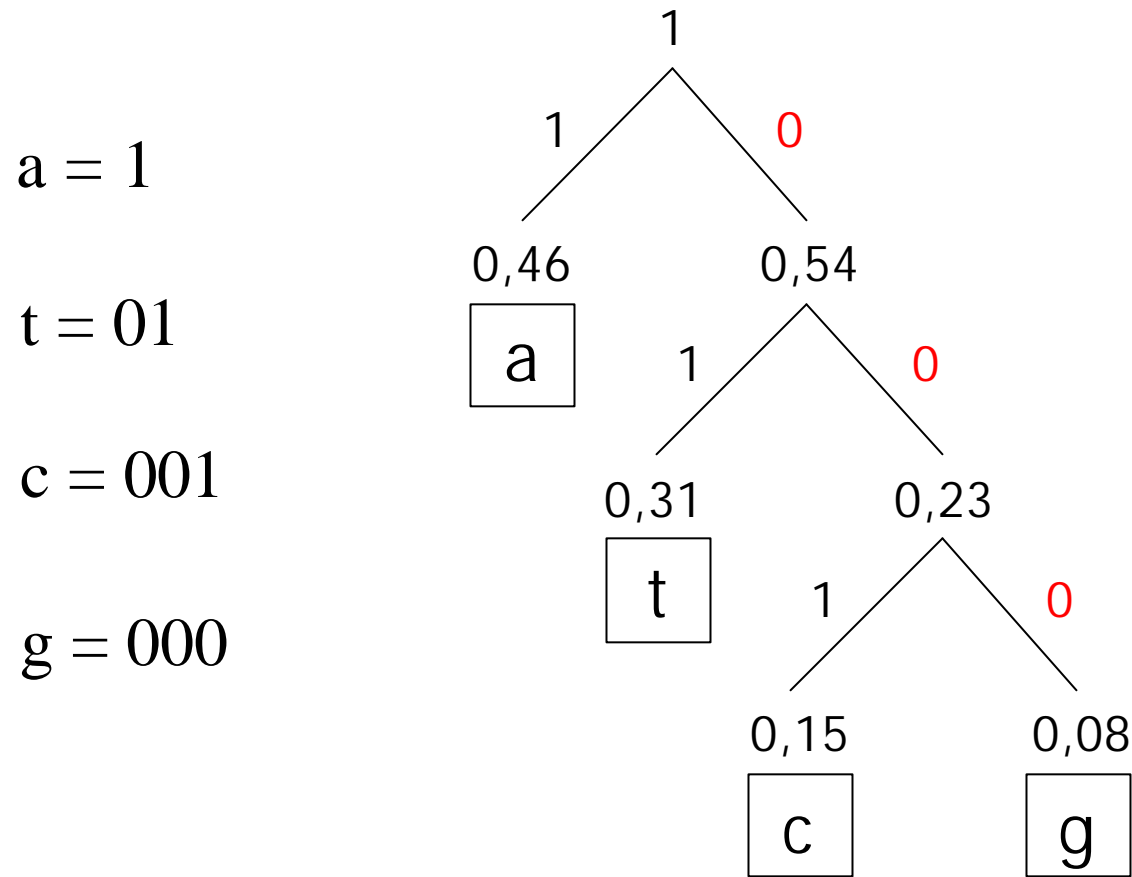


a = 1

t = 01

c = 001





Gegebenen sei die Zeichenfolge **ataactactag**

	a	t	c	g
Häufigkeit	6	4	2	1
W'keit	0,46	0,31	0,15	0,08
Präfix-Code	1	01	001	000

Kodierung: 10111011001011001011000

Es wurden nur **23 Bit** zur Kodierung benötigt.

Vergleich: 13 Zeichen => ASCII (8 Bit/Zeichen) = 104 Bit
Unicode (16 Bit/Zeichen) = 208 Bit

Huffman-Coding

Aha-Effekte

0,46	0,31	0,15	0,08
a	t	c	g

Präfixcode 1 01 001 000

Bitlänge_{errechnet} 1,1 1,7 2,7 3,6 $-\log_2 p(i)$

Huffman-Coding

Aha-Effekte

0,46	0,31	0,15	0,08
a	t	c	g

Präfixcode 1 01 001 000

Bitlänge_{errechnet} 1 2 3 4

	0,46	0,31	0,15	0,08
	a	t	c	g

Präfixcode	1	01	001	000
Bitlänge _{errechnet}	1	2	3	4
Bitlänge _{opt}	1	2	3	3

$$H(I) = \sum_{i=1}^n p_i \log_2 p_i$$

	0,46	0,31	0,15	0,08
	a	t	c	g
Präfixcode	1	01	001	000
Bitlänge _{errechnet}	1	2	3	4
Bitlänge _{opt}	1	2	3	3

$$H(I) = \sum_{i=1}^n p(a_i) \cdot |c(a_i)| = 0,46 \cdot 1 + 0,31 \cdot 2 + 0,15 \cdot 3 + 0,08 \cdot 3 = 1,77$$

$H(I) \cdot n(I) =$ minimale Bitanzahl zur Kodierung

ataatactactag

$$1,77 \cdot 13 = 23$$

Enzyklopädien-Homepages

<http://en.wikipedia.org/wiki/> or <http://de.wikipedia.org/wiki/>

<http://www.nationmaster.com/encyclopedia/>

<http://www.indexlist.de/>

<http://www.data-compression.com/>

<http://www.madeasy.de/>

<http://www.searchbites.com/docs/>