Grenzen der Datenkompression

Referat von Ingo Rohloff

10. Dezember 1996

Definition: Eine Funktion C, die eine Zufallsvariable X auf die Worte eines Alphabetes A abbildet, heißt *(source) Code*.
d.h. : $C : \Omega \rightarrow A^*$

Definition: Die erwartete Länge $L(C)$ eines Codes $C(x)$ für eine Zufallsvariable X mit der Verteilungsfunktion $p(x)$ ist durch:

$$L(C) = \sum_{x \in \Omega} p(x) l(x)$$

gegeben, wobei $l(x)$ die Länge des Codewortes für x ist.

Definiton: Ein Code heißt *nicht-singulär (non-singular)*, wenn jedem Element aus Ω ein anderes Codewort zugewiesen wird:

$$x_i \neq x_j \ \Rightarrow \ C(x_i) \neq C(x_j)$$

d.h. C ist injektiv.

Definition: Eine *Extension* C^* eines Codes C, ist eine Funktion die jedem Wort aus Ω^* ein Wort aus A^* durch folgende Vorschrift zuweist:

$$C^*(x_1 x_2 x_3 \cdots x_n) = C(x_1)C(x_2)C(x_3) \cdots C(x_n)$$

wobei $C(x_1)C(x_2) \cdots C(x_n)$ die Konkatenation der Codewörter ist. C^* ist also eine Funktion:
$C^* \ : \ \Omega^* \ \rightarrow \ A^*$

Definiton: Ein Code heißt *eindeutig decodierbar*, wenn seine Extension nicht-singulär ist.

Definition: Ein Code heißt *Präfix-Code* (oder *instantaneous code*), wenn kein Codewort das Präfix irgend eines anderen Codewortes ist. (Diese Bedingung nennt man auch Fano-Bedingung.)

Beispiel: Sei $\Omega = \{1, 2, 3, 4\}$ und $A = \{0, 1\}$

X	Singulär	Nicht-Singulär, nicht eind. decodierbar	eind. decodierbar, nicht Präfix-Code	Präfix-Code
1	0	0	10	0
2	0	010	00	10
3	0	01	11	110
4	0	10	110	111

Kraft-Ungleichung: Jeder Präfix-Code über einem Alphabet mit D Zeichen, mit den Codewortlängen $l_1, l_2, l_3, ..., l_m$, erfüllt die folgende Ungleichung:

$$\sum_{i=1}^{m} D^{-l_i} \leq 1$$

Beweis: Sei ohne Einschränkung der Allgemeinheit das Alphabet mit D Zeichen $\{0, 1, ..., D-1\}$. Jedem $x \in \Omega$ wird dann ein Codewort $y_1 y_2 \cdots y_{l(x)}$ mit $y_i \in \{0, 1, ..., D-1\}$ zugewiesen. Dadurch läßt sich jedem Codewort durch folgende Vorschrift eine reelle Zahl zwischen 0 und 1 zuweisen:

$$0.y_1 y_2 \cdots y_{l(x)} = \sum_{i=1}^{l(x)} y_i D^{-i}$$

Zu jedem Codewort gehört dann das Intervall:

$$\left[0.y_1 y_2 \cdots y_{l(x)}, 0.y_1 y_2 \cdots y_l(x) + \frac{1}{D^{l(x)}}\right[$$

Dessen Länge ist damit genau $D^{-l(x)}$. Weil es ein Präfix-Code ist, überschneiden sich die Intervalle nicht, und daraus folgt die Kraft-Ungleichung.
Umgekehrt kann man zu jeder Menge von Codewörtern, die die Kraft-Ungleichung erfüllen einen Präfix-Code angeben: Man sortiert die Längen in aufsteigender Reihenfolge und weist dann die Intervalle zu, beginnend bei 0.0...0. Beispiel:
$l_1 = 1, l_2 = 2, l_3 = 3, l_4 = 3 \Rightarrow$ Die Kraft-Ungleichung ist erfüllt:

1. Intervall: $[0.0 - 0.1[$ Code: 0
2. Intervall: $[0.10 - 0.11[$ Code: 10
3. Intervall: $[0.110 - 0.111[$ Code: 110
4. Intervall: $[0.111 - 1.000[$ Code: 111

McMillan Theorem: Die Codewortlängen $l_1, ..., l_m$ eines eindeutig decodierbaren Codes erfüllen die Kraft-Ungleichung:

$$\sum_{i=1}^{m} D^{-l_i} \leq 1$$

D ist dabei wieder die Anzahl der Zeichen im Codewortalphabet.

Beweis: Wir betrachten, um diese Ungleichung zu beweisen, die k-te Potenz davon:

$$\begin{aligned}\left(\sum_{x\in\Omega} D^{-l(x)}\right)^k &= \sum_{x_1\in\Omega}\sum_{x_2\in\Omega}\cdots\sum_{x_k\in\Omega} D^{-l(x_1)}D^{-l(x_2)}\cdots D^{-l(x_k)} = \\ &= \sum_{x_1\in\Omega}\sum_{x_2\in\Omega}\cdots\sum_{x_k\in\Omega} D^{-[l(x_1)+l(x_2)+\cdots+l(x_k)]}\end{aligned}$$

Jetzt summiert man über die Länge der codierten Zeichenkette, anstatt über die einzelnen Codewörter:

$$\left(\sum_{x\in,\Omega} D^{-l(x)}\right)^k = \sum_{l=1}^{k\cdot l_{max}} a(l) D^{-l}$$

wobei l_{max} die maximale Länge eines einzelnen Codewortes ist und $a(l)$ die Anzahl der Kombinationen aus k Codewörtern ist, die zusammengenommen l als Länge haben. Weil der Code eindeutig decodierbar ist muß $a(l) \leq D^l$ sein, denn es gibt nur D^l *verschiedene* Zeichenketten mit Länge l. Daraus folgt:

$$\begin{aligned}\left(\sum_{x\in\Omega} D^{-l(x)}\right)^k &= \sum_{l=1}^{k\cdot l_{max}} a(l) D^{-l} \\ &\leq \sum_{l=1}^{k\cdot l_{max}} D^l D^{-l} = k \cdot l_{max}\end{aligned}$$

und daraus:

$$\sum_{i=1}^{m} D^{-l_i} \leq (k \cdot l_{max})^{1/k}$$

Da diese Gleichung auch für $k \to \infty$ gilt folgt:

$$\sum_{i=1}^{m} D^{-l_i} \leq 1$$

was zu beweisen war.

Theorem: Die erwartete Länge jedes eindeutig decodierbaren Codes für eine Zufallsvariable X ist größer oder gleich der Entropie $H_D(X)$.

Beweis: Man betrachtet die Differenz zwischen der erwarteten Länge und der Entropie:

$$L(C) - H_D(X) = \sum_{i=1}^{m} p_i l_i \; - \; \sum_{i=1}^{m} p_i \log_D \frac{1}{p_i} = \sum_{i=1}^{m} p_i \log_D D^{l_i} \; - \; \sum_{i=1}^{m} p_i \log_D \frac{1}{p_i}$$

substituiert man $r_i = D^{-l_i} / \sum D^{-l_j}$, so ist r_i eine Wahrscheinlichkeitsverteilung und man erhält:

$$\begin{aligned} L - H &= \sum_{i=1}^{m} p_i \log_D \frac{p_i}{r_i} - \log_D \sum_{i=1}^{m} D^{-l_i} \\ &= D(p \| r) - \log_D \sum_{i=1}^{m} D^{-l_i} \\ &\geq 0 \end{aligned}$$

Shannon-Codes: Darunter versteht man Codes, deren Codewortlängen durch die Formel $l(x) = \lceil \log 1/p(x) \rceil$ gegeben sind, wobei $p(x)$ die Wahrscheinlichkeitsverteilung ist. Für solche Codes gilt:

$$H_D(X) \leq L(C) < H_D(X) + 1$$

Beweis: Zunächst einmal muß man beweisen, daß die Kraft Ungleichung gilt:

$$\sum_{i=1}^{m} D^{-\lceil \log \frac{1}{p_i} \rceil} \leq \sum_{i=1}^{m} D^{-\log \frac{1}{p_i}} = \sum_{i=1}^{m} p_i = 1$$

Durch die Wahl der Codewortlängen gilt:

$$\log_D \frac{1}{p_i} \leq l_i < \log_D \frac{1}{p_i} + 1$$

und daraus folgt dann:

$$H_D(X) \leq L(C) < H_D(X) + 1$$

Optimaler Code: Aus dieser Abschätzung und der Tatsache, daß die erwartete Länge des optimalen Codes $L(C_{Opt})$ immer noch über der Entropie liegen muß, folgt für den optimalen Code ebenso:

$$H_D(X) \leq L(C_{Opt}) < H_D(X) + 1$$

Diese obere Grenze läßt sich noch verkleinern, wenn man mehrere Zeichen auf einmal codiert, d.h. wenn man mit einer Abbildung $C : \Omega^n \to D^*$ codiert. Für eine Abschätzung führt man die Größe $L_n(C)$ ein:

$$L_n(C) = \frac{1}{n} \sum_{x_1, \ldots, x_n \in \Omega} p(x_1, ..., x_n) l(x_1, ..., x_n)$$

$L_n(C)$ ist damit die erwartete Länge pro Zeichen, wenn man n Zeichen auf einmal codiert. Aus den vorhergehenden Überlegungen folgt:

$$H(X_1, X_2, ..., X_n) \leq n \cdot L_n(C) < H(X_1, X_2, ..., X_n) + 1$$

nimmt man an, daß die Zufallsvariablen unabhängig und gleichverteilt sind, folgt:

$$n \cdot H(X) \leq n \cdot L_n(C) < n \cdot H(X) + 1$$

und daraus:

$$H(X) \leq L_n(C) < H(X) + \frac{1}{n}$$

Am Schluß noch ein paar Anmerkungen zu den Formeln: Die Entropie als untere Grenze für die erwartete Länge eines Codes, sagt *überhaupt nichts* über die Codierung einer speziellen Nachricht oder eines speziellen Zeichens aus, denn man kann natürlich einer speziellen Nachricht im Extremfall als Code einfach ein einzelnes Bit zuweisen. Die Entropie gibt Auskunft über eine Nachrichtenquelle und nicht über eine einzelne Nachricht. Das bedeutet die Frage "Wie klein kann ich diese Datei packen?" ist einfach falsch gestellt, denn eine Datei stellt eine *einzelne* Nachricht dar, die im Extremfall als einzelnes Bit codiert werden kann. Daraus folgt: Wenn man beurteilen will, wie gut ein Packverfahren ist, darf man es nicht an einer einzelnen Datei ausprobieren, sondern der Durchschnitt der Packrate ist entscheidend.