Informationstheorie, FS 2021
ETH Zürich, Dept. Informatik

Prof. Joachim M. Buhmann
Fabian Laumer

Lösungen zu Übungsblatt 6

6.1

a) Gegeben sei eine Folge von unabhängigen Zufallsvariablen $X_1, X_2, \ldots$ mit Werten in $\mathcal{X} = \{a, b, c\}$, die alle entsprechend $P(X_i = a) = 0.5$, $P(X_i = b) = 0.3$, $P(X_i = c) = 0.2$ verteilt sind. Geben Sie die beim arithmetischen Codieren verwendeten Intervalle I_{aba}, I_{abb} und I_{abc} an.

$I_{aba} = [0.25, 0.325)$, $I_{abb} = [0.325, 0.37)$, $I_{abc} = [0.37, 0.4)$.

b) Geben Sie die "Binärintervalle" $B_{b_1 \ldots b_N}$ an für $b_1 \ldots b_N \in \{000, 010, 10101\}$ an.

$B_{000} = [0, 0.001)_2 = [0, \frac{1}{8})$, $B_{010} = [0.01, 0.011)_2 = [\frac{1}{4}, \frac{3}{8})$, $B_{10101} = [0.10101, 0.1011)_2 = [\frac{21}{32}, \frac{11}{16})$ (wobei $[\ldots, \ldots)_2$ bedeutet, dass die Intervallgrenzen im Binärsystem angegeben sind).

6.2

Begründen Sie, dass bei arithmetischen Codes im Allgemeinen mit dem Decodieren eines Codeworts begonnen werden kann, bevor dessen Ende bekannt ist. Formulieren und beweisen Sie dazu ein Lemma analog zu dem in der Vorlesung, mit dem die entsprechende Aussage fürs Codieren begründet wurde.

Wie in der Vorlesung nehmen wir an, dass das Alphabet $\mathcal{X} = \{s_1, \ldots, s_M\}$ ein ausgezeichnetes "end of file"-Symbol $s_M = \square$ enthält, und wir schreiben

$$\mathcal{X}^*_\square = \{x_1 \ldots x_{n-1}\square \in \mathcal{X}^* \mid x_i \neq \square \text{ für alle } i = 1, \ldots, n-1\}$$

für die Menge der Wörter, die als Eingaben in Frage kommen.

Lemma. *Seien $b, b' \in \{0,1\}^*$, $x \in \mathcal{X}^*_\square$ und $x' \in (\mathcal{X} \setminus \{\square\})^*$ so, dass Folgendes gilt:*

a) *$B_b \subseteq I_x$ und B_b ist das längste "Binärintervall" mit dieser Eigenschaft,*

b) *$B_{b'} \subseteq I_{x'}$,*

c) *b' ist ein Präfix von b.*

Dann ist x' ein Präfix von x.

Beweis. Da b' Präfix von b ist, gilt $B_b \subseteq B_{b'}$ und damit folgt $B_b \subseteq I_{x'}$. Das impliziert $B_b \subset I_x \cap I_{x'}$ und damit $I_x \cap I_{x'} \neq \emptyset$, und daher muss $I_x \subseteq I_{x'}$ oder $I_{x'} \subseteq I_x$ gelten (das folgt aus der Konstruktion dieser Familie von Intervallen). Also ist x' ein Präfix von x oder x ist ein Präfix von x'. Wegen der Annahmen an die Gestalt von x, x' muss Letzteres der Fall sein. $\square$

Für die Decodierung des Codeworts $b \in \{0,1\}^*$ für ein Wort $x \in \mathcal{X}_\square^*$ bedeutet das: Sobald wir ein Präfix b' von b kennen, so dass $B_{b'} \subseteq I_{x'}$ gilt, wissen wir, dass x' Präfix von x ist und können somit x' als Anfang der decodierten Nachricht ausgeben.

6.3

Beschreiben Sie einen Algorithmus für die arithmetische Codierung von Strings $x_1 \dots x_N$ mit $\sum_{i=1}^N x_i = K$ für gegebene feste Werte von N und K. Geben Sie für den Fall $N = 5$, $K = 2$ die entsprechenden Intervalle für alle Teilstrings der Längen 1 bis 5 an.

Seien $0 \le n \le N$ und $0 \le k \le K$. Für $\mathbf{x} = x_1 \dots x_n \in \{0,1\}^n$ mit $\sum_{i=1}^n x_i = k$ gibt es $\binom{N-n}{K-k}$ mögliche Komplettierungen von $\mathbf{x}$ zu einem String der Länge N mit $\sum_{i=1}^N x_i = K$; davon fangen $\binom{N-n-1}{K-k}$ mit 0 und $\binom{N-n-1}{K-k-1}$ mit 1 an. Das natürliche probabilistische Modell, das für alle $0 \le n \le N-1$ die Verteilung von X_{n+1} gegeben $X_1 \dots X_n$ beschreibt, ist daher gegeben durch

$$P(X_{n+1} = 0 | X_1 \dots X_n = x_1 \dots x_n) = \binom{N-n-1}{K-k} / \binom{N-n}{K-k},$$
$$P(X_{n+1} = 1 | X_1 \dots X_n = x_1 \dots x_n) = \binom{N-n-1}{K-k-1} / \binom{N-n}{K-k}.$$

Die Intervalle $I_{x_1 \dots x_n}$ erhält man daraus entsprechend dem allgemeinen Rezept für die Konstruktion arithmetischer Codes.

Betrachtung des Falls $N = 5$ und $K = 2$: Es gibt $\binom{5}{2} = 10$ solcher Strings, und diese haben alle Wahrscheinlichkeit $\frac{1}{10}$; die entsprechenden Intervalle für Strings der Länge 5 sind daher $[a_{i-1}, a_i)$ mit $a_i := \frac{i}{10}$ für $i = 1, \dots 10$. Da jeder solche String durch ein Präfix der Länge 4 schon vollständig bestimmt ist, sind das auch die Intervalle für die Teilstrings der Länge 4. Die Intervalle für die Teilstrings kürzerer Länge sind: $I_{000} = [a_0, a_1)$, $I_{001} = [a_1, a_3)$, $I_{010} = [a_3, a_5)$, $I_{011} = [a_5, a_6)$, $I_{100} = [a_6, a_8)$, $I_{101} = [a_8, a_9)$, $I_{110} = [a_9, a_{10})$, $I_{00} = [a_0, a_3)$, $I_{01} = [a_3, a_6)$, $I_{10} = [a_6, a_9)$, $I_{11} = [a_9, a_{10})$, $I_0 = [a_0, a_6)$, $a_1 = [a_6, a_{10})$.

6.4

Im Folgenden soll der Lempel-Ziv-Algorithmus (wie in der Vorlesung besprochen) verwendet werden, um Strings in $\{a, b, c\}^*$ als Binärstrings zu codieren. Dabei verwendet wir als Codes für die einzelnen Symbole $a \mapsto 00$, $b \mapsto 01$, $c \mapsto 10$.

a) Codieren Sie *cabcccaacabac*.

Aufteilung in Phrasen: $c|a|b|cc|ca|ac|ab|ac$. Der Code ist damit

$$,10|0,00|00,01|01,10|001,00|010,10|010,01|110$$

(ohne die , und |). Die letzte Phrase ist schon im Wörterbuch, daher entfällt hier die Angabe eines Codes für das letzte Symbol.

b) Codieren Sie $aa \dots a \in \{a, b, c\}^n$ für beliebiges n. Zeigen Sie, dass für $n \to \infty$ die Anzahl der benötigten Bits pro Symbol gegen Null geht.

Aufteilung in Phrasen: $a|aa|aaa|\dots$. Sei $M \equiv M(n)$ die Anzahl der Phrasen, die bei der Aufteilung von a^n entstehen. Das Codewort für a^n ist dann

$$C(a^n) = 00[1]_2 00[2]_2 00 \dots [M-2]_2 00 c_M$$

Dabei bezeichnen wir mit $[\cdot]_2$ die Binärdarstellung, und c_M hat entweder die Form $c_M = [s]_2$ oder $c_M = [s]_2 00$ für eine ganze Zahl s mit $1 \leq s \leq M$.

Abschätzung der Länge von $C(a^N)$: Da $[1]_2, \ldots, [M-2]_2$ jeweils maximal $\log(M-2)+1$ Bits lang sind, erfüllt die Länge des Codeworts die Ungleichung

$$|C(a^n)| \leq M(\log(M-2)+3).$$

Da die m-te Phrase in a^n Länge m hat, ist die Länge des Strings vor Beginn der m-ten Phrase gleich $(m-1)m/2$. Daraus folgt die Ungleichung

$$(M-1)M/2 + 1 \leq n.$$

Diese impliziert insbesondere $M^2 \leq 2n$, und daraus folgen die Abschätzungen $\log M \leq \log(2n)$ und $M \leq \sqrt{2n}$. Beide Abschätzungen zusammen ergeben

$$\begin{aligned} \frac{1}{n}|C(a^n)| &\leq \frac{1}{n} M(\log M + 3) \\ &\leq \frac{1}{n}\sqrt{2n}(\log(2n)+3), \end{aligned}$$

und die rechte Seite strebt für $n \to \infty$ gegen 0.