Informatik IV - Tutorium XII & XIII (SR -120)

Tut Nr. 8 – Informationstheorie

David Münch

Universität Karlsruhe (TH)
Fakultät für Informatik
IBDS Prautzsch

19. Juni 2008

Universität Karlsruhe (TH)
Forschungsuniversität • gegründet 1825

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Inhaltsverzeichnis

Organisatorisches

Email: muenchdavid@gmail.com

https://www.stud.uni-karlsruhe.de/~uhbro/

Tutorium 12: Donnerstags 8:00 Uhr - Raum -120
Tutorium 13: Donnerstags 9:45 Uhr - Raum -120

Übungsblattabgabe Donnerstag.

Schein / Übungsblätter

Nur die mit (**K**) gekennzeichneten Aufgaben sind abzugeben.
Diese werden korrigiert und bewertet.
66% der Punkte aller mit (**K**) gekennzeichneten Aufgaben aller Übungsblätter sind notwendig, um einen Schein zu erhalten.

Schein / Übungsblätter

Nur die mit (**K**) gekennzeichneten Aufgaben sind abzugeben.
Diese werden korrigiert und bewertet.
66% der Punkte aller mit (**K**) gekennzeichneten Aufgaben aller Übungsblätter sind notwendig, um einen Schein zu erhalten.

Abgabe meistens Donnerstags.

Schein / Übungsblätter

Nur die mit (**K**) gekennzeichneten Aufgaben sind abzugeben.
Diese werden korrigiert und bewertet.
66% der Punkte aller mit (**K**) gekennzeichneten Aufgaben aller Übungsblätter sind notwendig, um einen Schein zu erhalten.

Abgabe meistens Donnerstags.

Abgabe in Zweiergruppe erlaubt und ausdrücklich erwünscht!

Schein / Übungsblätter

Nur die mit (**K**) gekennzeichneten Aufgaben sind abzugeben. Diese werden korrigiert und bewertet.
66% der Punkte aller mit (**K**) gekennzeichneten Aufgaben aller Übungsblätter sind notwendig, um einen Schein zu erhalten.

Abgabe meistens Donnerstags.

Abgabe in Zweiergruppe erlaubt und ausdrücklich erwünscht!

Um das Übungsteam zu unterstützen bitte folgendes Deckblatt verwenden:
`http://www.stud.uni-karlsruhe.de/~unbdh/deckblatt/index.php?course=5`

Literatur

Boehm, Prautzsch: Numerical Methods. AK Peters 1993. ISBN 3-528-06350-5
http://www.ubka.uni-karlsruhe.de/hylib-bin/suche.cgi?opacdb=UBKA_OPAC&fbt=7319953&nd=3204657

Ash: Information Theory. Dover 1990. ISBN 0-486-66521-6
http://www.ubka.uni-karlsruhe.de/hylib-bin/suche.cgi?opacdb=UBKA_OPAC&nd=9866904

Goos: Vorlesungen über Informatik. Bd. 4, Springer 1998. ISBN 3-540-60650-5
http://www.ubka.uni-karlsruhe.de/hylib-bin/suche.cgi?opacdb=UBKA_OPAC&fbt=9316367&nd=6568301

Was wollen wir heute erreichen?

Was wollen wir heute erreichen?

- Übungsblatt 8 besprechen

Was wollen wir heute erreichen?

- Übungsblatt 8 besprechen
- Wiederholung von Grundbegriffen der Informationstheorie

Was wollen wir heute erreichen?

- Übungsblatt 8 besprechen
- Wiederholung von Grundbegriffen der Informationstheorie
- verschiedene verlustfreie Codierungsverfahren vorstellen

Aufgabe 40 (K, $1 + 2 + 2 = 5$ Punkte)

Wir benutzen Bezeichnungen wie in Kapitel 5.8 der Vorlesung.

Wir betrachten ein vorwärtsgerichtetes Perzeptron mit m Schichten, Zustandsmenge $Q = \mathbb{R}$, n_0 Eingabeneuronen, n_m Ausgabeneuronen, linearer Erregungsfunktion $h(x) = x$, Gewichten ω_{ij}^k, Schwellwerten σ_k und Lernmuster $(\mathbf{e}, \mathbf{a})$, wobei

$$n_0 = n_2 = 2, n_1 = 1, m = 2, \mathbf{e} = [1,\ 2]^t, \mathbf{a} = [2,\ 3]^t, \sigma_k = \mathbf{0}, \omega_{ij}^k = 1.$$

1. Führen Sie einen Schritt des Lernens mit Rückkopplung und mit Lernrate $\eta = 0{,}1$ durch.

Algorithm 1 MUSTERLERNEN DURCH RÜCKKOPPLUNG

1. Vorwärtsrechnen

for $k = 1..m$ **do**

  berechne q_k, p_k und

  $J_k = J(p_k)$

end

2. Rückwärtsrechnen

$d_{m+1} = (q_m - a)$

$W_{m+1} = E$

for *k=m..1* **do**

  berechne $d_k^t = d_{k+1}^t W_{k+1} J_k$ und

  $\frac{\partial E}{\partial W_k}$

end

Aufgabe 41 (K, 0,5 + 0,5 + 1 + 1 + 1 + 1 = 5 Punkte)

Durch Einsatz eines Hopfield-Netzes soll ein ungerichteter Graph $G = (V, E)$ in zwei Teilgraphen partitioniert werden. Die Teilgraphen $G_1 = (V_1, E_1) := (V_1, E \cap V_1^2)$ und $G_2 = (V_2, E_2) := (V \setminus V_1, E \cap V_2^2)$ sollen (in Bezug auf die Anzahl der Knoten) möglichst gleich groß sein ($|V_1| \approx |V_2|$), und es sollen möglichst wenige (nämlich $|E \cap (V_1 \times V_2)| = |E| - |E_1| - |E_2|$) Kanten übrig bleiben.

Konkret sei gegeben:

$$G = (\{A, B, C, D, E, F\}, \{AB, CD, EF, AC, CE, EA, BD, DF, FB\}).$$

1. Welche Topologie (Netzstruktur) hat das zugehörige Hopfield-Netz?
2. Beschreiben Sie anschaulich in Worten, was der Parameter γ angibt.
3. Geben Sie für $\gamma = 1/2$ die Gewichte w_{ij} und Schwellwerte ρ_j des Hopfield-Netzes an. Stellen Sie die Matrizen K und W auf.
4. Sei ($V_1 = \{A\}, V_2 = \{B, C, D, E, F\}$) eine initiale Zerlegung von V. Geben Sie den aktuellen Zustand $\mathbf{q}$ an und simulieren Sie einen asynchronen Schalt-Verlauf des Netzes bis zu Ende. Interpretieren Sie das Ergebnis.
5. Simulieren Sie nun das Ganze noch einmal mit derselben initialen Zerlegung, aber für $\gamma = 2$. Interpretieren Sie das Ergebnis.
6. Welche (optimale) Partitionierung würden Sie für G intuitiv vorschlagen?

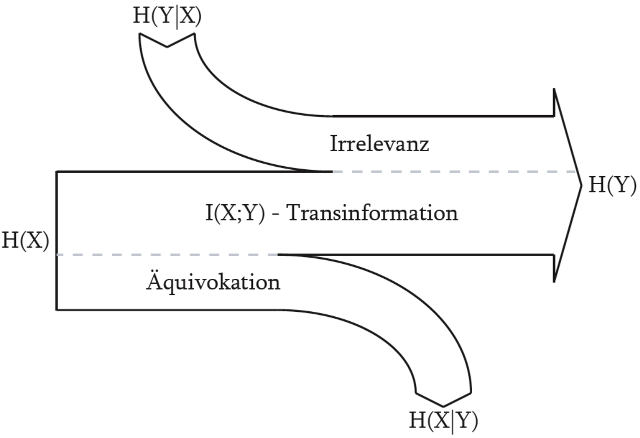
Entropie

Definition: Entropie

Die Entropie (Informationsgehalt) einer Zufallsvariablen X ist
$H(X) = H(p_1, ..., p_m) = -\sum_{i=1}^{m} p_i \log p_i$

Die Entropie ist die Information pro Zeichen, die wir erwarten.
Oder: Die Entropie ist die durchschnittliche Anzahl von Entscheidungen (bits), die benötigt werden, um ein Zeichen aus einer Zeichenmenge zu identifizieren oder zu isolieren.

Übertragungskanal



Zusammenfassung

Transinformation: $I(X|Y) = H(X) - H(X|Y) = I(Y|X)$

Zusammenfassung

Transinformation: $I(X|Y) = H(X) - H(X|Y) = I(Y|X)$
Äquivokation: $H(X|Y) = H(X) - I(X|Y)$

Zusammenfassung

Transinformation: $I(X|Y) = H(X) - H(X|Y) = I(Y|X)$
Äquivokation: $H(X|Y) = H(X) - I(X|Y)$
Irrelevanz: $H(Y|X) = H(Y) - I(X|Y)$

Zusammenfassung

Transinformation: $I(X|Y) = H(X) - H(X|Y) = I(Y|X)$
Äquivokation: $H(X|Y) = H(X) - I(X|Y)$
Irrelevanz: $H(Y|X) = H(Y) - I(X|Y)$
Totalinformation: $H(Y|X) + I(X|Y) + H(X|Y)$

Kanaleigenschaften

- deterministisch: $H(Y|X) = 0$

Kanaleigenschaften

- deterministisch: $H(Y|X) = 0$
- verlustfrei: $H(X|Y) = 0$

Kanaleigenschaften

- deterministisch: $H(Y|X) = 0$
- verlustfrei: $H(X|Y) = 0$
- störungsfrei: $H(X|Y) = H(Y|X) = 0$

Kanaleigenschaften

- deterministisch: $H(Y|X) = 0$
- verlustfrei: $H(X|Y) = 0$
- störungsfrei: $H(X|Y) = H(Y|X) = 0$
- nutzlos: $I(X|Y) = 0$

Kanaleigenschaften

- deterministisch: $H(Y|X) = 0$
- verlustfrei: $H(X|Y) = 0$
- störungsfrei: $H(X|Y) = H(Y|X) = 0$
- nutzlos: $I(X|Y) = 0$

Die Zufallsvariablen X und Y sind genau dann stochastisch unabhängig, wenn der Kanal nutzlos ist.
$H(X,Y) = H(X) + H(Y)$

Kanalkapazität

Definition: Kanalkapazität

$C = max_{P(X)}\{I(X|Y)\}$
C ist die höchste Informationsmenge, die unter allen möglichen Quellenverteilungen über den Kanal übertragen werden kann.

Definition: Codierung

Sei $\mathbb{A} = \{a_1, .., a_m\}$ ein Alphabet und A eine Zufallsvariable über $\mathbb{A}$ mit den Wahrscheinlichkeiten $p_i = P(A = a_i)$. Eine Codierung von A über einem Codealphabet $\mathbb{X} = \{x_1, .., x_d\}$ ist eine Abbildung: $C : \mathbb{A} \rightarrow \mathbb{X}^+$. Diese wird für ganze Worte erweitert: $C^* : \mathbb{A}^* \rightarrow \mathbb{X}^*$, $a_{i_1}...a_{i_k} = c_{i_1}...c_{i_k}$

Definition: Codierung

Sei $\mathbb{A} = \{a_1, .., a_m\}$ ein Alphabet und A eine Zufallsvariable über $\mathbb{A}$ mit den Wahrscheinlichkeiten $p_i = P(A = a_i)$. Eine Codierung von A über einem Codealphabet $\mathbb{X} = \{x_1, .., x_d\}$ ist eine Abbildung: $C : \mathbb{A} \to \mathbb{X}^+$. Diese wird für ganze Worte erweitert: $C^* : \mathbb{A}^* \to \mathbb{X}^*$, $a_{i_1}...a_{i_k} = c_{i_1}...c_{i_k}$

Definition: Codelänge

Die mittlere Codelänge ist $L(C) = \sum_{i=1}^{m} p_i \cdot l_i$

Codeeigenschaften

- regulär, wenn C injektiv

Codeeigenschaften

- regulär, wenn C injektiv
- dekodierbar, wenn C^* injektiv

Codeeigenschaften

- regulär, wenn C injektiv
- dekodierbar, wenn C^* injektiv
- Präfixcode, wenn kein c_i Präfix eines andern c_j ist.

Konstruktion optimaler Codes

Satz

Für jeden Präfix- und dekodierbaren Code gilt die Kraft-Ungleichung:

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

Wenn diese Gleichung erfüllt ist, gibt es einen Präfix- bzw. dekodierbaren Code mit diesen Längen l_i

Kodierungstheorem

Theorem

Die Länge eines dekodierbaren Codes C ist mindestens so gross wie die Entropie der Zufallsvariablen A:
$L(C) \geq H_d(p)$
mit Gleichheit $\Leftrightarrow d^{-l_i} = p_i$ für alle i.

Kodierungstheorem

Theorem

Die Länge eines dekodierbaren Codes C ist mindestens so gross wie die Entropie der Zufallsvariablen A:
$L(C) \geq H_d(p)$
mit Gleichheit $\Leftrightarrow d^{-l_i} = p_i$ für alle i.

Ein optimaler Präfixcode L^* hat die Länge:
$H_d(p) \leq L^* < L(C) < H_d(p) + 1$

Redundanz

Redundanz ist überflüssige Information. D.h. sie sieht aus wie Information, ist aber keine.

Definition: Absolute Redundanz

$R_{abs} = \tilde{H} - H$ mit

Realinformation (Entropie)
$H = \mathbb{E}(h(p_i)) = -\sum_{i=1}^{n} p_i \cdot h(p_i) = -\sum_{i=1}^{n} p_i \cdot \log p_i$

Nominalinformation
$\tilde{H} = \mathbb{E}(\text{Länge}(code_i)) = \sum_{i=1}^{n} p_i \cdot \text{Länge}(code_i)$

Redundanz

Redundanz ist überflüssige Information. D.h. sie sieht aus wie Information, ist aber keine.

Definition: Absolute Redundanz

$R_{abs} = \tilde{H} - H$ mit

Realinformation (Entropie)

$H = \mathbb{E}(h(p_i)) = -\sum_{i=1}^{n} p_i \cdot h(p_i) = -\sum_{i=1}^{n} p_i \cdot \log p_i$

Nominalinformation

$\tilde{H} = \mathbb{E}(\text{Länge}(code_i)) = \sum_{i=1}^{n} p_i \cdot \text{Länge}(code_i)$

Definition: Relative Redundanz

$R_{rel} = 1 - \frac{H}{\tilde{H}}$

Huffman-Code

Huffman-Code

Ein Huffman-Code liefert einen beweisbar optimalen Code für gegebene Wahrscheinlichkeiten.

Konstruktionsskizze:

- Erstelle einen Wald mit Bäumen, für jedes Zeichen. Diese Bäume enthalten nur einen Knoten: das Zeichen
- suche die beiden Bäume im Wald, die für die Zeichen mit der kleinsten Wahrscheinlichkeit stehen. Entferne diese Bäume aus dem Wald. Erstelle einen neuen Baum, der die beiden entfernten Bäume als Unterbaum hat. Füge diesen Baum in den Wald ein. Benutze dabei die Summe der Wahrscheinlichkeiten der Unterbäume.
- Wiederhole, bis nur noch ein Baum übrig ist.

Aufgabe

a) Entwickle einen Huffman-Code zu der Zeichenkette
$ABRACADABRA$

Aufgabe

a) Entwickle einen Huffman-Code zu der Zeichenkette $ABRACADABRA$

b) Berechne die relative Redundanz deines Codes.

Aufgabe

a) Entwickle einen Huffman-Code zu der Zeichenkette $ABRACADABRA$
b) Berechne die relative Redundanz deines Codes.
c) Welche relative Redundanz hätte ein Shannon-Code für diese Zeichenkette?

Aufgabe

a) Entwickle einen Huffman-Code zu der Zeichenkette $ABRACADABRA$
b) Berechne die relative Redundanz deines Codes.
c) Welche relative Redundanz hätte ein Shannon-Code für diese Zeichenkette?
d) Dekodiere ...

Aufgabe

a) Eine Quelle sende Zeichen aus dem Alphabet $X = \{A, B, C, D, E, F\}$ aus.
Die Auftrittswahrscheinlichkeiten seien:
$p_A = 0.25, p_B = 0.06, p_C = 0.04, p_D = 0.1, p_E = 0.35, p_F = 0.2$
Erzeuge einen ternären Huffman-Code C_X mit Symbolen aus $\mathcal{C} = \{1, 2, 3\}$ und berechne die erwartete Codelänge $L(C_X)$.

Aufgabe

a) Eine Quelle sende Zeichen aus dem Alphabet $X = \{A, B, C, D, E, F\}$ aus.
Die Auftrittswahrscheinlichkeiten seien:
$p_A = 0.25, p_B = 0.06, p_C = 0.04, p_D = 0.1, p_E = 0.35, p_F = 0.2$
Erzeuge einen ternären Huffman-Code C_X mit Symbolen aus $\mathcal{C} = \{1, 2, 3\}$ und berechne die erwartete Codelänge $L(C_X)$.

b) Betrachte eine Quelle Y mit Alphabet $\mathcal{Y}, |\mathcal{Y}| = 7$. Für diese Quelle wurde ein optimaler Code C_Y gefunden für den gilt: $L(C_Y) = H_3(Y)$.
Bestimme eine mögliche Wahrscheinlichkeitsverteilung $p(y)$ für die Quelle Y.

Kullback-Leibler-Distanz

Definition: Kullback-Leibler-Distanz

Die Kullback-Leiber-Distanz, oder auch relative Entropie genannt, ist folg. definiert:

$$D(p||q) = \sum_i p_i \log \frac{p_i}{q_i}$$

Kullback-Leibler-Distanz

Definition: Kullback-Leibler-Distanz

Die Kullback-Leiber-Distanz, oder auch relative Entropie genannt, ist folg. definiert:

$$D(p||q) = \sum_i p_i \log \frac{p_i}{q_i}$$

Sie gibt aus informationstheoretischer Sicht an, wieviele Bits durchschnittlich verschwendet werden, wenn eine eigentlich auf q basierende Kodierung auf Ereignisse angewendet wird, die p folgen.

LZW-Codierung

LZW-Codierung ist ein populäres Codierungsverfahren nach Lempel, Ziv und Welch.

Die Idee besteht darin, dass ein Wörterbuch zur Speicherung wiederkehrender Wörter zur Laufzeit generiert und in die Datei implizit gschrieben wird.

Algorithm 2 LZW COMPRESSOR ALGORITHM

```
w = NIL
while read a char c do
    if wc exists in dictionary then
        w = wc
    else
        add wc to the dictionary
        output the code for w
        w = c
    end
end
output the code for w
```

Beispiel

Kompression

Eingabe	Ausgabe	w
a		
b		
a		
b		
a		
b		
a		

Beispiel

Kompression

Eingabe	Ausgabe	w
a	a	
b		
a		
b		
a		
b		
a		

Beispiel

Kompression

Eingabe	Ausgabe	w
a	a	256=ab
b		
a		
b		
a		
b		
a		

Beispiel

Kompression

Eingabe	Ausgabe	w
a	a	256=ab
b	b	
a		
b		
a		
b		
a		

Beispiel

Kompression

Eingabe	Ausgabe	w
a	a	256=ab
b	b	257=ba
a		
b		
a		
b		
a		

Beispiel

Kompression

Eingabe	Ausgabe	w
a	a	256=ab
b	b	257=ba
a	256	
b		
a		
b		
a		

Beispiel

Kompression

Eingabe	Ausgabe	w
a	a	256=ab
b	b	257=ba
a	256	258=aba
b		
a		
b		
a		

Beispiel

Kompression

Eingabe	Ausgabe	w
a	a	256=ab
b	b	257=ba
a	256	258=aba
b		
a	258	
b		
a		

Beispiel

Dekompression

Eingabe	Ausgabe	w
a		
b		
256		
258		

Beispiel

Dekompression

Eingabe	Ausgabe	w
a	a	
b		
256		
258		

Beispiel

Dekompression

Eingabe	Ausgabe	w
a	a	
b	b	
256		
258		

Beispiel

Dekompression

Eingabe	Ausgabe	w
a	a	
b	b	256=ab
256		
258		

Beispiel

Dekompression

Eingabe	Ausgabe	w
a	a	
b	b	256=ab
256	ab	
258		

Beispiel

Dekompression

Eingabe	Ausgabe	w
a	a	
b	b	256=ab
256	ab	257=ba
258		

Beispiel

Dekompression

Eingabe	Ausgabe	w
a	a	
b	b	256=ab
256	ab	257=ba
258	aba	258=aba

Ist der zu decodierende Eintrag noch nicht im Wörterbuch, so besteht er aus der Ausgabe $a_1 \dots a_k a_1$, wobei $a_1 \dots a_k$ die Ausabe im vorhergegangenen Schritt war.

Aufgabe

Über dem Alphabet $X = \{A, B, C\}$ sei eine Zeichenkette w gegeben durch die LZW-Codierung

$$0, 1, 2, 3, 5, 4, 6$$

Dabei sind die Wörterbucheinträge anfänglich gegeben durch $0 = A, 1 = B, 2 = C$

a) Wie lang ist w?

Aufgabe

Über dem Alphabet $X = \{A, B, C\}$ sei eine Zeichenkette w gegeben durch die LZW-Codierung

$$0, 1, 2, 3, 5, 4, 6$$

Dabei sind die Wörterbucheinträge anfänglich gegeben durch $0 = A, 1 = B, 2 = C$

a) Wie lang ist w?

b) Um wie viele Bits differiert die Länge der LZW-Codierung von w von der Länge einer Huffman-Codierung von w?

Lösung

Eingabe	Ausgabe	w
0	A	3=AB
1	B	4=BC
2	C	5=CA
3	AB	6=ABC
4	CA	7=CAB
5	BC	8=BCA
6	ABC	

$$w = ABCABCABCABC$$

Lösung

Eingabe	Ausgabe	w
0	A	3=AB
1	B	4=BC
2	C	5=CA
3	AB	6=ABC
4	CA	7=CAB
5	BC	8=BCA
6	ABC	

$w = ABCABCABCABC$
Huffman-Codierung: $code(w) = 4 \cdot (1bit + 2bit + 2bit) = 20bit$
LZW-Codierung: $0..6 \leq 8 = 2^3 \rightarrow 3bit/Zeichen \rightarrow 7 \cdot 3bit = 21bit$
Differenz ist dann $1bit$.

Burrows-Wheeler-Transformation

Motivation: BWT macht Texte etc. für eine anschliessende Lauflängencodierung geeigneter. Bei einer Lauflängencodierung werden Blöcke beliebiger Länge codiert mit Beachtung der Abhaengigkeiten der einzelnen Zeichen.
Die BWT komprimiert noch nicht, sie permutiert die Zeichen.

Burrows-Wheeler-Transformation: $w \to v$

Gegeben: Blöcke w der Länge n: $w = x_1 ... x_n$
Gesucht: Eine Permutation der Indizes $i \in \{1, ..., n\}$

Aufgabe

Führe die BWT auf der Zeichenkette $w = BLUBBLUB$ durch. Gib als Ergebnis alle Informationen an, die die Rücktransformation benötigt um w wieder herzustellen.

Lösung

$$
\left[\begin{array}{c|cccccc|c}
B & B & L & U & B & B & L & U \\
B & B & L & U & B & B & L & U \\
B & L & U & B & B & L & U & B \\
B & L & U & B & B & L & U & B \\
L & U & B & B & L & U & B & B \\
L & U & B & B & L & U & B & B \\
U & B & B & L & U & B & B & L \\
U & B & B & L & U & B & B & L
\end{array}\right]
\begin{array}{l}
(1) \\
(2) \\
(3) \Leftarrow \text{Original} \\
(4) \Leftarrow \text{Original} \\
(5) \\
(6) \\
(7) \\
(8)
\end{array}
$$

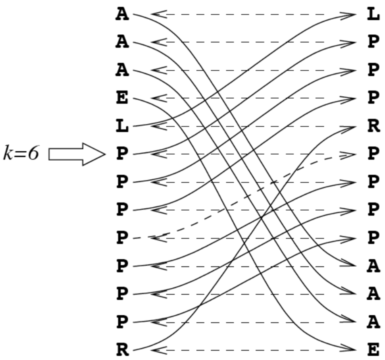
$L = UUBBBBLL$ und $k = 3$ oder $k = 4$

Aufgabe

Dekodiere die lauflängencodierte Folge
$(L,1),(P,3),(R,1),(P,4),(A,3),(E,1)$.
Führe für die Ergebniszeichenkette v die inverste BWT mit Startindex $k=6$ durch und gib das Ergebnis w an.

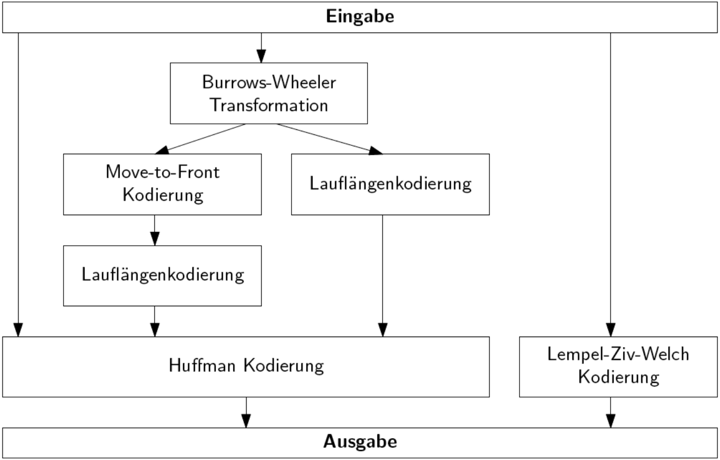
Lösung

$$v = LPPPRPPPPAAAE$$



$$w = PAPPERLAPPAPP$$

Verlustfreie Codierungen



Quellen

Pajor - Informatik 4 Tutorium SS2007

Prautzsch - Skript Informatik 4 SS2008

Ash: Information Theory. Dover 1990. ISBN 0-486-66521-6
http://www.ubka.uni-karlsruhe.de/hylib-bin/suche.cgi?opacdb=UBKA_OPAC&nd=9866904

Wikipedia

Reflexion

Was haben wir heute gelernt?

Reflexion

Was haben wir heute gelernt?

- Übungsblatt 8 besprochen

Reflexion

Was haben wir heute gelernt?

- Übungsblatt 8 besprochen
- Grundbegriffe der Informationstheorie wiederholt

Reflexion

Was haben wir heute gelernt?

- Übungsblatt 8 besprochen
- Grundbegriffe der Informationstheorie wiederholt
- Verschiedene verlustfreie Codierungen behandelt

Noch Fragen?

Vorschau

Vorschau

- Fehlerkorrigierende Codes

Bis zum nächsten Mal