

Verfahren der Datenkomprimierung

Verlustfreie Kompression

- Lauflängenkodierung (RLE)

- Wörterbuchkodierung

- Huffman-Kodierung

•
•
•

Verlustbehaftete Kompression

- Farbquantisierung

- Transformationsverfahren

- Psychoakustische Kompression

•
•
•



Verfahren der Datenkomprimierung

Wörterbuchkodierung am Beispiel von LZ77

LZ77 erkennt wiederkehrende Zeichenfolgen in bereits gelesenen Daten und ersetzt diese durch Verweise auf frühere Positionen.

Beispiel 1:

Originaltext: ABCABC
Komprimierter Text: ABC(3,3)

(3,3) → „Gehe 3 Zeichen zurück,
kopiere die nächsten 3 Zeichen
und füge sie hier () ein.“

Beispiel 2:

Originaltext: ABCABCDABC
Komprimierter Text: ABC(3,3)D(4,3)

Für Beispiel 2 existiert mindestens eine weitere sinnvolle Lösung für LZ77. Finden Sie diese heraus.

ABC(3,3)D(7,3)

Um schnell vergleichbare Lösungen zu erhalten,
gelten für die folgenden Aufgaben zwei Regeln:

1. Es wird immer die längste passende Zeichenkette gewählt.
2. Bei gleich langen Treffern wird der kleinste Abstand gewählt.

(In realen LZ77-Implementierungen können auch andere Auswahlregeln gelten.)



Verfahren der Datenkomprimierung

Wörterbuchkodierung am Beispiel von LZ77

Komprimieren Sie mit LZ77: ABABCABABCAB

AB(2,2)C(5,5)(3,2)

Stellen Sie den Originaltext wieder her: ABC(3,3)D(4,3)

ABCABCDABC

Komprimieren Sie mit LZ77: ABCDABCDABCDE

ABCD(4,4)(4,4)E

Stellen Sie den Originaltext wieder her: ABC(3,3)D(4,3)E(8,4)

ABCABCDABCEABCD



Verfahren der Datenkomprimierung

Wörterbuchkodierung am Beispiel von LZ77

Berechnen Sie den Speicherbedarf der folgenden unkomprimierten Information, für die genau fünf verschiedene Zeichen zur Auswahl standen.

ABCDABCDABCD E

5 Zeichen $\rightarrow$ 3 Bit je Zeichen notwendig ($2^3 = 8 > 5$)

13 · 3 Bit = 39 Bit bei unkomprimiert

Komprimieren Sie mit LZ77.

ABCD (4,4) (4,4) E

Berechnen Sie den Speicherbedarf der komprimierten Information. Rechnen Sie mit einer maximalen Länge und einem maximalen Abstand von 4 Zeichen.

Notwendige Bit je Angabe der Länge: 2 ($2^2 = 4$)

Notwendige Bit je Angabe des Abstands: 2 ($2^2 = 4$)

5 · 3 Bit + 2 · 2 Bit + 2 · 2 Bit = 23 Bit für komprimiert

Hinweis: Zur Vereinfachung werden zusätzliche Steuerinformationen realer LZ77-Verfahren nicht berücksichtigt.



Verfahren der Datenkomprimierung

Wörterbuchkodierung am Beispiel von LZ77

DEFGGDEFHH

Berechnen Sie

- den Speicherbedarf der unkomprimierten Information, für die genau fünf verschiedene Zeichen zur Auswahl standen.
- den Speicherbedarf der komprimierten Information mit Maximalwerten für Abstand und Länge von jeweils 8 Zeichen.

Bewerten Sie die Komprimierung der Information.

5 Zeichen $\rightarrow$ 3 Bit je Zeichen notwendig ($2^3 = 8 > 5$)

10 · 3 Bit = 30 Bit bei unkomprimiert

DEFG(1,1) (5,3)H(1,1)

Notwendige Bit je Angabe der Länge: 3 ($2^3 = 8$)

Notwendige Bit je Angabe des Abstands: 3 ($2^3 = 8$)

5 · 3 Bit + 3 · 3 Bit + 3 · 3 Bit = 33 Bit für komprimiert

Die Komprimierung lohnt sich nicht, weil die komprimierte Datei sogar größer wäre.

Verfahren der Datenkomprimierung

Wörterbuchkodierung am Beispiel von LZ77

LUSLUSCTLUSV

Berechnen Sie

- den Speicherbedarf der unkomprimierten Information, für die genau sechs verschiedene Zeichen zur Auswahl standen.
- den Speicherbedarf der komprimierten Information mit Maximalwerten für Abstand von 16 und Länge von 8 Zeichen.

Bewerten Sie die Komprimierung der Information.

6 Zeichen $\rightarrow$ 3 Bit je Zeichen notwendig ($2^3 = 8 > 6$)

$12 \cdot 3 \text{ Bit} = \underline{\underline{36 \text{ Bit}}}$ bei unkomprimiert

LUS(3,3)CT(5,3)V

Notwendige Bit je Angabe der Länge: 3 ($2^3 = 8$)

Notwendige Bit je Angabe des Abstands: 4 ($2^4 = 16$)

$6 \cdot 3 \text{ Bit} + 2 \cdot 3 \text{ Bit} + 2 \cdot 4 \text{ Bit} = \underline{\underline{32 \text{ Bit}}}$ für komprimiert

Die Komprimierung lohnt sich, weil die komprimierte Datei vier Bit kleiner ist.

FLEXAB

Passend zu diesem Skript findest
du das Arbeitsblatt
"Game Log Compression"
auf

www.flexAB.de

