

# Kann die Entropie von Sequenzen mittelst der Kompressibilität gemessen werden?

Werner Ebeling zum 60. Geburtstag gewidmet

Thorsten Pöschel

*Institut für Physik, Humboldt Universität zu Berlin, Invalidenstraße 110, D-10115 Berlin*

## 1 Einführung

Wenn die Werte einer binären Sequenz mit einer bestimmten Wahrscheinlichkeit angenommen werden, so kann man der Wahrscheinlichkeitsverteilung eine Entropie zuordnen. Die Entropie nimmt ihr Maximum dann an, wenn die möglichen Werte  $L$  und  $H$  mit gleicher Wahrscheinlichkeit und unkorreliert auftreten. Es ist plausibel, daß die Notation einer solchen Sequenz der Länge  $N \gg 1$  in nicht wesentlich weniger als  $N$  binären Zeichen notiert werden kann, ein Kompressionsalgorithmus würde bei einer solchen Sequenz versagen. Ebenso ist evident, daß eine periodische Sequenz mit der Periodenlänge  $P$  (die Periode sei in sich unkorreliert) mit ungefähr  $\text{ld}(\lfloor N/P \rfloor) + P$  binären Zeichen notiert, d.h. auf diese Größe komprimiert werden kann. Die Entropie einer periodischen Sequenz ist klein. Die Vermutung liegt nahe, daß die Entropie einer Sequenz und deren Kompressibilität verwandte Größen sind: kennt man die Entropie, kann man auf die Kompressibilität schließen, kennt man die Kompressibilität, so hat man damit ein Maß für die Entropie, kann also die Entropie quantitativ abschätzen.

Während die Bestimmung der Entropie einer Sequenz fast ausschließlich für Theoretiker von Interesse ist, ist die Kompression von Daten von höchstem wirtschaftlichen Interesse<sup>1</sup>, so daß sich viele Wissenschaftler und Ingenieure mit der Entwicklung und Optimierung von leistungsfähigen Kompressionsalgorithmen befassen. Man kann nun fragen, ob man dieses Wissen dazu benutzen kann, um die Entropie einer Sequenz mittelst ihrer Kompressibilität zu bestimmen, und mit dieser Frage befaßt sich die vorliegende Arbeit.

Im folgenden Abschnitt wollen wir versuchen, den Zusammenhang zwischen Kompressibilität und Entropie quantitativ zu fassen. In Abschnitt 3 soll die Idee der Entropiebestimmung mit Hilfe der Kompressibilität an einigen Beispielen getestet werden.

---

<sup>1</sup>Zum Beispiel werden in jedem Faxgerät die zu übermittelnden Daten komprimiert. Gelänge es, die dazu verwandten Algorithmen um wenige Prozent zu vervollkommen, so könnten Telefonkosten in Millionenhöhe eingespart werden.

## 2 Kompressibilität und Entropie

Shannons Kodierungstheorem (z.B. [1, 2]) sagt aus, daß  $N$  unabhängige und nach derselben Wahrscheinlichkeitsverteilung mit der Entropie  $H$  erzeugte zufällige Variable immer in mehr als  $NH$  Bits komprimiert werden können, wobei der dabei entstehende Verlust an Information mit  $N \rightarrow \infty$  verschwindet.

Diese Aussage von großem wissenschaftlichen Interesse (z.B. [2]) ist in der Praxis von geringerer Bedeutung, denn häufig besteht die Aufgabe nicht darin, eine verlustfreie Repräsentation zu finden, sondern ein Informationsverlust wird ausdrücklich zugelassen: Die Kunst besteht darin, herauszufinden, welcher Teil der Information für den Empfänger von Wichtigkeit ist, es wird also nach einer semantischen Komprimierung gesucht. Wichtige Erfolge wurden dabei bei der Bildverarbeitung erzielt. Es existieren eine ganze Reihe von Standardverfahren zur Bildkompression, die bewußt Informationsverlust in Kauf nehmen, z.B. MPEG- und JPEG-Verfahren (JPEG: Joint Photographic Experts Group, MPEG: Moving Pictures Experts Group). Diese sollten natürlich so beschaffen sein, daß diese Verluste nicht als störend empfunden werden – ein interdisziplinäres Problem, das nicht von Informatikern allein gelöst werden kann. Um diese Art der Kompression geht es hier nicht, vorläufig wollen wir uns mit einfacheren Dingen befassen: Das Shannonsche Kodierungstheorem sagt aus, daß es möglich ist, einen Satz zufälliger Variabler, z.B. eine Sequenz, zu komprimieren. Insbesondere wissen wir, daß zur Kompression von  $N$  unabhängigen Zufallsdaten  $x_1, x_2, \dots, x_N$  etwa  $NH(X) + \epsilon$  Bits benötigt werden, wobei  $H(X)$  die Entropie der Wahrscheinlichkeitsverteilung  $X$  ist, nach der die Daten erzeugt worden sind. Man kann vermuten, daß man normalerweise weniger Bits benötigen werden, falls die  $x_i$  korreliert sind.

Wir wollen nun den theoretischen Zusammenhang zwischen Entropie und Kompressibilität herausarbeiten und folgen dabei [3, 4, 5]. Wir werden uns um der Kürze und der Übersichtlichkeit Willen auf binäre Kodierungen beschränken. Die Argumentation bleibt gültig für jedes Alphabet, auch die Verallgemeinerung auf kontinuierliche Variable ist möglich, allein die mathematische Formulierung wird dann aufwendig. Es soll darauf hingewiesen werden, daß wir an dieser Stelle nicht immer alle mathematischen Spitzfindigkeiten diskutieren wollen. Für eine sehr präzise und mathematisch einwandfreie Darstellung sei auf [1, 4, 6] verwiesen.

Definieren wir eine Sprache  $A^+$ , geschrieben in einem Alphabet  $A$ .  $A$  ist eine Menge von „Buchstaben“, z.B. das binäre Alphabet  $\{L, H\}$ . Mit  $A^N$  bezeichnen wir dann einen Satz von geordneten  $N$ -Tupeln, also Wörtern, aus der Menge  $A$ . In unserem Beispiel wäre also  $\{L, H\}^3$  die Menge  $\{LLL, HLL, LHL, HHL, LLH, HLH, LHH, HHH\}$ . Eine Sprache  $A^+$  ist dann die Menge aller (endlicher) Wörter, die aus den Buchstaben des Alphabets  $A$  zusammengesetzt werden können. Im Falle des binären Alphabets wäre die Sprache also  $\{L, H\}^+ = \{L, H, LL, LH, HL, HH, LLL, HLL, \dots\}$ . Ein Text ist also ein Element der Menge  $A^+$ .

Nun müssen wir noch Definitionen von Codes hinzufügen: Ein Symbolkode ist eine Abbildung der Elemente des Alphabets  $A = \{a_1, a_2, \dots, a_I\}$  auf die Menge  $\{L, H\}^+$ .  $c(x)$  ist das Kodewort eines Buchstaben  $x$  und  $l(x)$  ist die Länge des Kodeworts, bzw. kürzer  $l_i = l(a_i)$ . Ein Sprachenkode  $C^+$  (kurz Kode) ist eine Abbildung aus

$A^+$  in  $\{L, H\}^+$ . Bei der Kodierung werden die Symbolkodes einfach hintereinandergelagert. Wenn das Multiplikationszeichen Konkatenation bedeutet, schreiben wir dafür  $c^+(x_1 \cdot x_2 \cdot \dots \cdot x_N) = c(x_1) \cdot c(x_2) \cdot \dots \cdot c(x_N)$ .

Ein brauchbarere Kode  $C^+$  soll natürlich nicht nur eine Abbildung von  $A^+$  nach  $\{L, H\}^+$  leisten, sondern er soll auch *dekodierbar* sein. Da wir *a priori* keinen Wortseparator kennen, hat nicht jeder Kode diese Eigenschaft. Die Abbildung  $A \rightarrow L, B \rightarrow H, C \rightarrow LH$  hat diese Eigenschaft nicht: Die Kodierung von  $ABCAB$  lautet  $LHLLHLH$ , dekodiert man diese Sequenz, so kann man z.B.  $ABCAB$  oder  $CCC$  oder  $CABC$  erhalten. Ein Kode  $C(X)$  ist dann eindeutig dekodierbar, wenn jedem Element aus  $A^+$  ein anderes Bitwort zugeordnet wird. Diese Eigenschaft ist offenbar gegeben, wenn kein Kodewort Präfix eines anderen Kodewortes ist. Hat ein Kode diese Eigenschaft, so nennt man ihn Präfixkode. Obwohl nicht nur Präfixkodes die geforderte Eigenschaft der Dekodierbarkeit besitzen, liegt die Bedeutung solcher Kodes auf der Hand: Will man einen kodierten Text dekodieren, so liest man einfach von links nach rechts und es kann dabei nie passieren, daß man ein Teilstück übersetzt, und es sich erst später im Text herausstellt, daß eine andere Übersetzung gemeint ist. Oder einfacher: wenn ein Wort beendet ist, so wird man das merken, der folgende Text hat keinen Einfluß mehr auf die Übersetzung dieses Wortes. Präfixkodes sind also eindeutig dekodierbar.

Wir befassen uns nun mit der Länge eines Kodes und definieren dazu den Erwartungswert der Länge  $L(C)$  einer kodierten Sequenz  $C(X)$ :  $L(C) = \sum_{x \in A} P(x)l(x)$ .  $P(x)$  ist die Wahrscheinlichkeit, mit der der Buchstabe  $x$  im zu kodierenden Text vorkommt. Wir betrachten das Beispiel eines zu kodierenden Textes aus den Buchstaben  $A = \{A, B, C, D\}$ , dessen Buchstaben mit den Wahrscheinlichkeiten  $P_X = \{\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\}$  vorkommen. Kodieren wir  $c(A) = L, c(B) = HL, c(C) = HHL, c(D) = HHH$ , so können wir uns zunächst davon überzeugen, daß dieser Kode ein Präfixkode ist: Beginnt ein Text mit  $L$ , so übersetzt man  $A$  und weiß, daß man das Wortende gefunden hat. Findet man stattdessen  $H$ , so muß man weiterlesen. Folgt  $L$ , so übersetzt man  $B$  und hat das Wortende erreicht, usw. Jeder kodierte Text kann somit eindeutig dekodiert werden, z.B. findet  $c^+(\vec{x}) = LHLLHHHLLHHL$  nur eine einzige Möglichkeit der Übersetzung:  $ACDBAC$ . Setzen wir die Zahlenwerte ein, so finden wir für die Entropie von  $X$ :  $H(X) = 1.75$  Bits, und die erwartete Länge ist  $L(C) = 1.75$ . Wir werden sehen, daß ein Kode mit dieser Eigenschaft  $H(X) = L(C)$  ausgezeichnet ist. Ein anderer Präfixkode ist  $c(A) = LL, c(B) = LH, c(C) = HL, c(D) = HH$ . Auch Texte die damit kodiert worden sind, lassen sich eindeutig dekodieren. Für diesen Kode berechnen wir  $L(C) = 2$  Bits.

Man kann erwarten, daß die Kodewortlängen  $l_i$  den Schlüssel zur Abschätzung der Kompressibilität liefern. Stellen wir also die vorläufig etwas seltsam erscheinende Frage: Wenn man eine Menge natürlicher Zahlen  $\{l_i\}$  gegeben hat, gibt es dann einen eindeutig dekodierbaren Kode mit diesen Zahlen als Kodewortlängen? Die Antwort auf diese Frage gibt der Satz von Kraft und McMillan[7, 8]: Für jeden eindeutig dekodierbaren Kode  $C$  über dem binären Alphabet  $\{L, H\}$  müssen die Kodewortlängen der Gleichung  $\sum_i \frac{1}{2^{l_i}} \leq 1$  genügen. Auch die Umkehrung dieses Satzes gilt. Die Aussage scheint trivial - aber nur für den, der schon vorher weiß, was rauskommt! Wir werden ihn benützen, um eine wichtige Aussage zu beweisen, dazu verwenden wir die Gibbs-Ungleichung

$\sum_i p_i \log 1/q_i \geq \sum_i p_i \log 1/p_i$  ( $p_i, q_i$  sind Wahrscheinlichkeitsverteilungen):

Die untere Grenze für die Länge eines eindeutig dekodierbaren Codes ist die Entropie der Verteilung  $H(X)$  (nach [5]):

$$L(C) = \sum_i p_i l_i = \sum_i p_i \operatorname{ld} \frac{1}{\sum_{i'} 2^{-l_{i'}}} - \operatorname{ld} \sum_i 2^{-l_i} \geq \sum_i p_i \operatorname{ld} \frac{1}{p_i} - \operatorname{ld} \sum_i 2^{-l_i} \geq H(X)$$

Für den Fall, daß im Kraft-McMillan-Satz das Gleichheitszeichen gilt, verschwindet  $\operatorname{ld} \sum_i 2^{-l_i}$  und wir erhalten nun exakt das, was wir schon vorher wußten, oder wenigstens ahnten: Wenn man geschickt (binär) kodiert, so entspricht die Länge der kürzesten möglichen Kodierung gerade der Entropie. Darüber hinaus wissen wir: Die erwartete Länge wird minimal, wenn das Gleichheitszeichen im Kraft-McMillan-Satz gilt, d.h.  $l_i = \operatorname{ld}(p_i^{-1})$ . Weitere interessante Aussagen lassen sich über den Zusammenhang zwischen Kodierungen und allgemeineren Rényi-Entropien treffen [9].

Man kann nun einen solchen einfachen optimalen Code, den Huffman-Code [10], angeben (unser erstes Beispiel einer Kodierung ist gerade ein Huffman-Code, ähnliche Verfahren wurden schon früher entwickelt [11, 12]). Huffman-Kodes sind nur auf den ersten Blick optimal, insbesondere dann, wenn die Elementarbausteine des Codes nicht bekannt sind, und das ist die Standardsituation. Es existieren andere, darunter wesentlich bessere, aber auch sehr einfache Verfahren, z.B. die arithmetische Kodierung [1, 8, 13, 14], die Lauflängenkodierung (e.g. [15]), Lempel-Ziv-Algorithmen [16, 17]. Eine gute Übersicht findet sich in [18, 19, 20].

Es lassen sich viele Parallelen zu verwandten Problemen ziehen (die sich beim genaueren hinsehen sogar als identisch herausstellen). Wir wollen nur zwei Dinge erwähnen. Entropie und Vorhersagbarkeit: Besitzt ein Text eine hohe Redundanz, so ist er offensichtlich gut vorhersagbar, d.h. auf Grundlage der Kenntnis eines Textes aus  $N$  Buchstaben kann man den  $N + 1$ -ten Buchstaben umso besser raten, je redundanter der Text ist. Redundanz ist aber nur ein anderes Wort für kleine Entropie [21]. Der  $N + 1$ -te Buchstabe eines Textes kleiner Entropie läßt sich also zuverlässiger vorhersagen als für einen Text großer Entropie (s. z.B. [22, 23]). Shannon hat den Zusammenhang zwischen Vorhersagbarkeit und Entropie benützt, um den Informationsgehalt pro Buchstabe in englischen Texten abzuschätzen, indem er Testpersonen befragt hat, wie ein Textstück zu vervollständigen sei [24]. Das Problem der Vorhersagbarkeit wird ausführlich in [25, 26] diskutiert.

Ein anderer Aspekt derselben Sache ist das Problem der algorithmischen Komplexität (Kolmogorov-Solomonoff-Komplexität) [27, 28, 29]. Wir betrachten eine (binäre) Sequenz der Länge  $N$  und interessieren uns für Programme für eine universelle Turing-Maschine (z.B. [6]), die genau diese Sequenz erzeugen. Die Längen dieser Programme kann man in Bits messen. Die Länge des kürzesten dieser Programme gibt die algorithmische Komplexität an. Offensichtlich können komplexe Sequenzen (= Sequenzen großer Entropie = inkompressible Sequenzen) nicht durch ein Programm der Länge kleiner als  $N$  Bit erzeugt werden. Die Kolmogorov-Solomonoff-Komplexität ist ein sinnvolles Maß, jedoch gibt es leider keinen Algorithmus, sie exakt zu berechnen (z.B. [30, 6]), wohl aber Abschätzungen (z.B. [31]).

Information, und damit Kompressibilität und Entropie, sind natürlich auch auf nichttriviale Weise abhängig vom Vorwissen des Empfängers, ein Aspekt, der hier nur erwähnt werden soll. Ein bekanntes Beispiel ist das Berry-Paradoxon (z.B. [32, 33]): Man kann eine Zahl mit der Vorschrift bestimmen: *die kleinste positive ganze Zahl, die in Worten nicht mit weniger als einer Million Buchstaben geschrieben werden kann.*

### 3 „Messungen“

#### 3.1 Idee

Nachdem im vorangegangenen Abschnitt gezeigt worden ist, daß es *prinzipiell* möglich ist, aus der Kompressibilität einer Sequenz deren Entropie zu bestimmen, wollen wir nun untersuchen, ob wir mit diesem Wissen etwas über die Eigenschaften von konkreten Sequenzen aussagen können. Die Idee ist sehr einfach: Wenn wir eine Sequenz  $S_A = [s_1 s_2 \dots s_k s_{k+1} \dots s_n]$  an einer zufällig gewählten Stelle  $k$  auftrennen und die Stücke vertauscht aneinanderfügen  $S_E = [s_{k+1} \dots s_n s_1 s_2 \dots s_k]$ <sup>2</sup> zerstören wir Korrelationen und erzeugen neue Korrelationen. Falls  $S_A$  eine Ordnung hatte, so sollte beim zufälligen Umordnen diese Ordnung meistens verkleinert werden, d.h. die erzeugten Korrelationen sind im Mittel geringer als die zerstörten, und folglich sollte sich  $S_A$  mit einem leistungsfähigen Kompressionsalgorithmus auf ein kleineres Maß komprimieren lassen als  $S_E$ . Da wir die Sequenz nur ein einziges Mal Umordnen, können wir nach Mittelung über viele solcher Operationen der Ordnung in  $S_A$  einen Wert zuweisen.

#### 3.2 Englischer Text und Zufallssequenz

Die statistische Untersuchung natürlichsprachlicher Texte hat eine lange Tradition, die bis zu Shannon [24] zurückreicht. Wir wollen an dieser Stelle nicht die umfangreichen Resultate referieren, sondern verweisen nur auf die Literatur, z.B. [4],[34]-[44].

Zur Untersuchung der Kompressibilität haben wir aus dem Text von „Moby Dick“ (Gesamtlänge ca. 1,2 MByte) 1000 Sequenzen mit zufällig gewähltem Anfangspunkt der Länge 30000 herausgeschnitten und mit einem Alphabet der Größe 32 kodiert [42]. Anschließend ordnen wir die Sequenzen wie beschrieben um und erhalten die Sequenzen  $S_A^1 \dots S_A^{1000}, S_E^1 \dots S_E^{1000}$ . Die Indizes bezeichnen den Original-File (A) und den umgeordneten File (E). Wir speichern die Sequenzen als ASCII Zeichen und erhalten 2000 Files von je 30000 Bytes. Diese Files komprimieren wir mit dem Programm *gzip* [18], das wohl zur Zeit einer der leistungsfähigsten schnellen Lempel-Ziv-Komprimierer ist und erhalten die komprimierten Files  $f_A^i$  und  $f_E^i$ , ( $i = 1 \dots 1000$ ).

Abb. 1 (links) zeigt die Verteilung der Längen der komprimierten Files<sup>3</sup>  $\#(f_A^i)$  und  $\#(f_E^i)$ . Die rechte Seite zeigt dieselbe Verteilung, wobei die Originalsequenz eine Zufallssequenz der Länge 1,2 MByte ist, geschrieben im selben 32-er Alphabet wie die

<sup>2</sup>Der Anfang  $s_1$  und das Ende  $s_k$  der Sequenzen sollten nicht ausgezeichnet sein. Wir haben deshalb in den Beispielen ein Stück aus der Mitte einer Originalsequenz herausgeschnitten und damit operiert.

<sup>3</sup>In den Abbildungen 1-6 beziehen sich die gestrichelten Kurven auf die Daten der umgeordneten Files und die ausgezogenen Kurven auf die Daten der Original-Files.

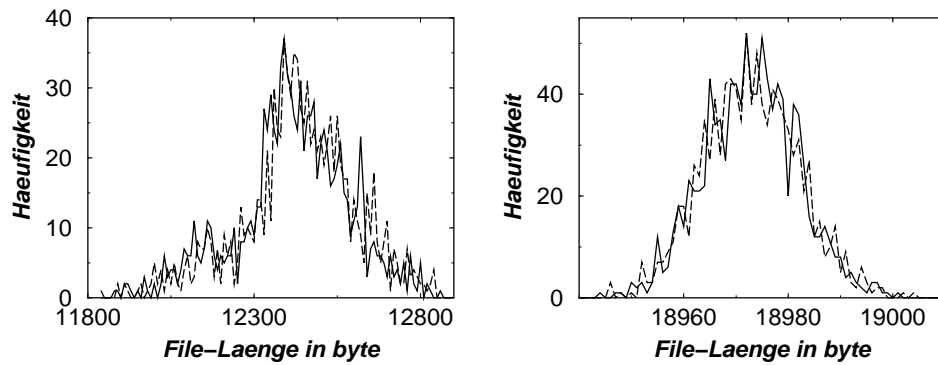


Abbildung 1: Histogramm der Längen der komprimierten Moby-Dick-Sequenzen (links) und der Zufallssequenzen (rechts).

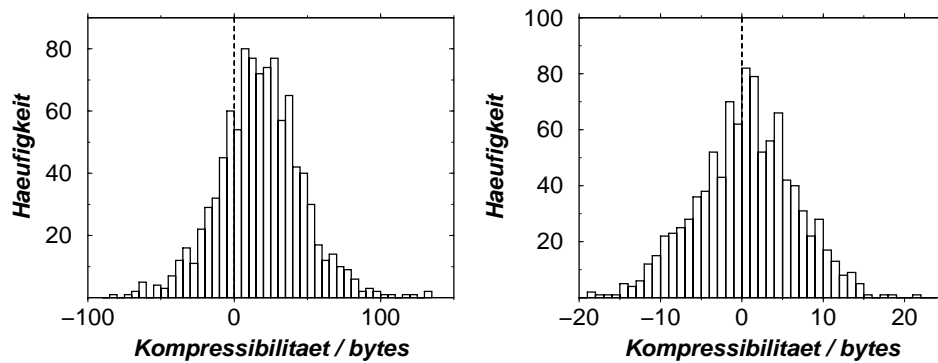


Abbildung 2: Histogramm der Längendifferenz zwischen dem komprimierten Original-File und dem komprimierten gebrochenen File (li.: Moby Dick, re.: Zufallssequenz).

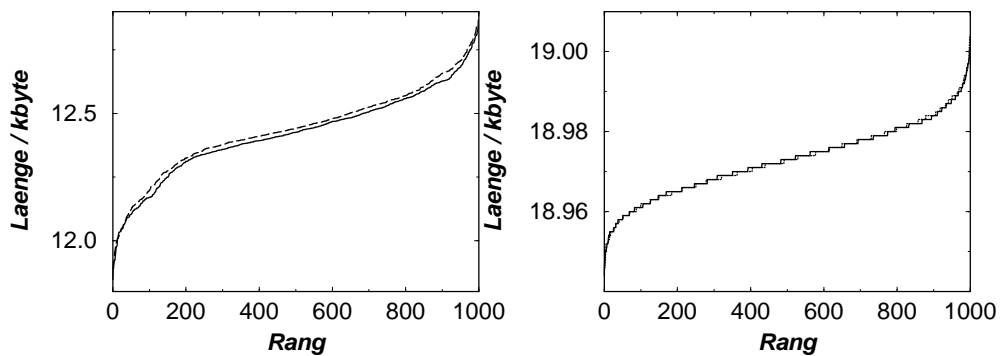


Abbildung 3: Ranggeordnete Längen der komprimierten Original-Files und der komprimierten gebrochenen Files (links: Moby Dick, rechts: Zufallssequenz).

kodierte Moby-Dick-Sequenz. Wir können links keine Regelmäßigkeiten beobachten und schließen, daß diese Darstellung ungeeignet ist.

Tragen wir in Abb. 2 die Verteilung der Größenunterschiede  $\#(f_E^i) - \#(f_A^i)$  auf, so sehen wir, daß die komprimierten umgeordneten Moby-Dick-Files  $f_E^i$  häufig größer sind als die komprimierten Originalsequenzen  $f_A^i$  (links). Dasselbe Bild für die Zufallssequenz ist wie erwartet symmetrisch. Ordnet man also Moby-Dick an einer zufällig gewählten Stelle um, so läßt sich der entstandene File im Mittel schlechter komprimieren als der Original-File. Deutlicher wird der Unterschied zwischen Moby Dick und der Zufallssequenz in Abb. 3, wo die ranggeordneten Größen der komprimierten umsortierten Files  $\#(f_E^i)$  und der komprimierten Original-Files  $\#(f_A^i)$  gezeichnet sind. Im Falle der Zufallssequenz (rechts) finden wir nur eine sehr kleine Abweichung der Kurven, für den englischen Text weichen die Kurven systematisch voneinander ab. Die mittlere Differenz der Kompressibilität beträgt  $\Delta = \left(\sum_{i=1}^{1000} \#(f_E^i) - \#(f_A^i)\right) / 1000 = 12437.6 - 12420.4 = 17.2$  Bytes. Für die Zufallssequenz erhält man  $\Delta = 18972.882 - 18972.880 = 0.002$  Bytes. Diese Werte können als Maß für die bei einer Umsortierung im Mittel zerstörten Korrelationen gelten.

### 3.3 Feigenbaum Abbildung, Katja-Sequenz und Zufallszahlen

Wir wollen nun noch zwei Sequenzen mit bekannten Korrelationen untersuchen, die Feigenbaum-Sequenz [45] und die Katja-Sequenz [46]. Die Abbildungen 4, 5 und 6 zeigen die zu 1-3 analogen Kurven. Wir können hier nicht auf Details eingehen, sondern wollen nur darauf hinweisen, daß unser Maß  $\Delta$  auch bei diesen Sequenzen Korrelationen anzeigt: Für die Katja-Sequenz erhält man  $\Delta = 820.717 - 809.519 = 11.2$  Bytes, die Feigenbaum-Sequenz liefert  $\Delta = 2.62$  Bytes.

Schließlich haben wir versucht, die Korrelationen der Daten eines schlechten Zufallszahlengenerators zu messen. Wir haben die Sequenz  $x_{i+1} = x_i \cdot 16807 \pmod{(2^{31} - 1)}$  (s. [19, 47]) auf ein 26-er Alphabet abgebildet und mit dem beschriebenen Algorithmus bearbeitet. Das Resultat war  $\Delta = (18973005 - 18972999) / 1000$  Bytes = 0.006 Bytes. (Ein guter Zufallszahlengenerator (s. 3.2) lieferte  $\Delta = 0.002$  Bytes.) Obwohl das Vorzeichen des Resultats anzeigt, daß die ursprünglichen Sequenzen stärker korreliert waren als die umgeordneten, muß die Signifikanz dieser kleinen Zahl bezweifelt werden.

Die ranggeordneten Differenzen der Kompressibilitäten, d.h. die geordneten Differenzen der Kurven 3 und 6 und die Daten des zuletzt untersuchten schlechten Zufallszahlengenerators, sind in Abb. 7 zusammengestellt.

## 4 Zusammenfassung

Der Zusammenhang der Kompressibilität einer Sequenz mit ihrer Komplexität (Entropie) wurde theoretisch erläutert. Anhand einiger Sequenzen haben wir eine Meßvorschrift angegeben, mit der Sequenzen hinsichtlich ihrer Korrelationen untersucht werden können. In drei Beispielen korrelierter Sequenzen hat die Methode qualitativ korrekte Resultate geliefert, eine Zufallssequenz wurde mit großer Zuverlässigkeit als

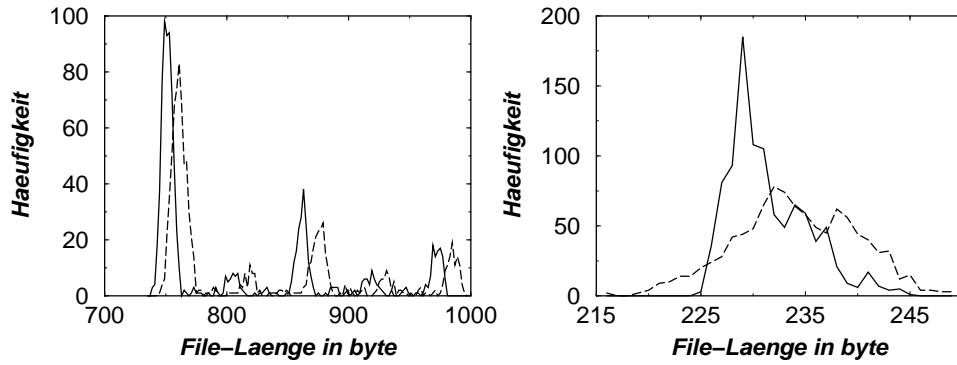


Abbildung 4: Histogramm der Längen der komprimierten Sequenzen.

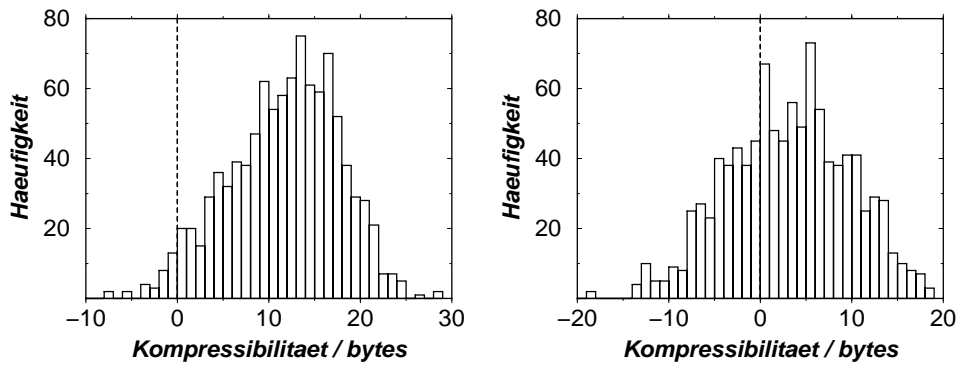


Abbildung 5: Histogramm der absoluten Längendifferenz zwischen dem komprimierten Original-File und dem komprimierten gebrochenen File.

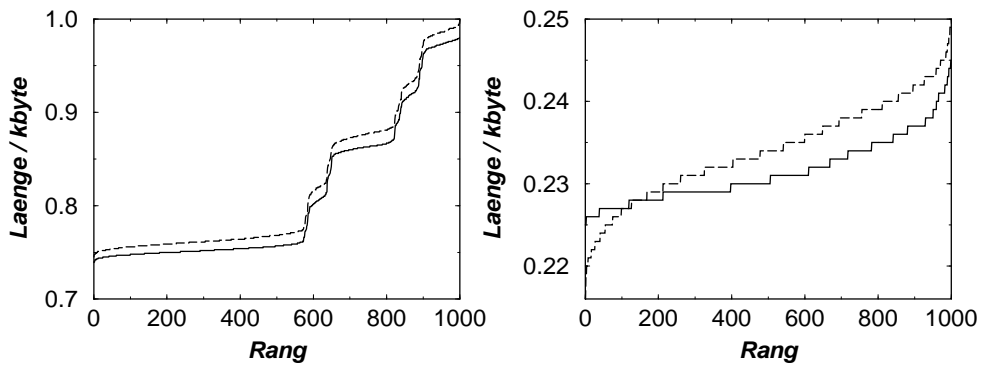


Abbildung 6: Ranggeordnete Längen der komprimierten Original-Files und der komprimierten gebrochenen Files.



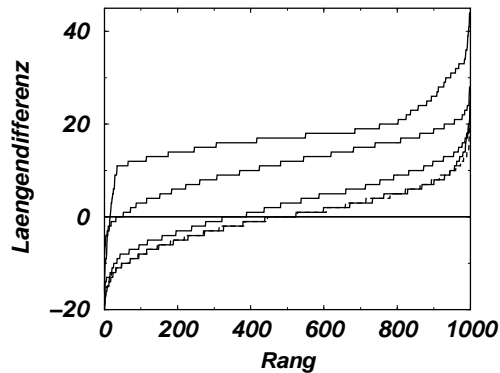


Abbildung 7: Die Differenzen der Kompressibilitäten für Moby Dick, die Katja-Sequenz, die Feigenbaum-Sequenz und den schlechten Zufallszahlengenerator (von oben nach unten). Die Kurve der Zufallssequenz (Abschnitt 3.2) fällt fast mit dem schlechten Zufallszahlengenerator zusammen. Sie ist mit Strichlinie gezeichnet.

solche erkannt. Inwiefern die mit dem Programm *gzip* gemessenen Werte der Kompressibilität als *quantitatives* Maß der Entropie einer Sequenz dienen können, ist unklar und könnte untersucht werden.

Der Autor bedankt sich bei Dieter Beule, Jan Freund und Katja Rateitschak für die im Abschnitt 3.3 untersuchten Sequenzen.

## Literatur

- [1] N. Abramson, *Information Theory and Coding*, McGraw-Hill (New York, 1963).
- [2] P. Nath und D. P. Mittal, „A Generalization of Shannon’s Inequality and Its Application in Coding Theory“, *Inform. and Control*, **23**, 439-445 (1973).
- [3] G. J. Chaitin, *Information, Randomness and Incompleteness*, World Scientific (Singapore, 1990).
- [4] E. T. Jaynes, *Probability Theory: The Logic of Science* (im Druck).
- [5] D. MacKay, „A Short Course on Information Theory“, unveröff. Manuskript.
- [6] W. J. Paul, *Komplexitätstheorie*, Teubner (Stuttgart, 1978).
- [7] I. Csiszár und J. Körner, *Information Theory*, Academic Press (New York, 1981).
- [8] J. J. Rissanen, „Generalized Kraft Inequality and Arithmetic Coding“ *IBM J. Res. Dev.* **20**, 198-203 (1976).
- [9] L. L. Campbell, „A Coding Theorem and Rényi’s Entropy“, *Inform. and Control*, **8**, 423-429 (1965).

- 
- [10] D. A. Huffman, „A Method for the Construction of Minimum-Redundancy Codes“, *Proc. IRE*, **40**, 1098-1101 (1952).
- [11] C. E. Shannon und W. Weaver, „The Mathematical Theory of Communication“, University of Illinois Press (Urbana, 1949).
- [12] R. M. Fano, *Transmission of Information*, M. I. T. Press, (Cambridge, 1949).
- [13] R. Pasco, „Source Coding Algorithms for Fast Data Compression“, Dissertation, Dept. of Electrical Engineering, Stanford Univ., Stanford (1976).
- [14] H. Printz, „Tutorial on Arithmetic Coding“, *unveröff. Manuskript* (1994).
- [15] R.C. Gonzalez, P. Wintz, *Digital Image Processing*, Addison Wesley, (Reading, 1977).
- [16] J. Ziv und A. Lempel, „A Universal Algorithm for Sequential Data Compression“, *IEEE Trans. Inform. Theory*, **23**, 337-343 (1977).
- [17] J. Ziv und A. Lempel, „Compression of Individual Sequences via Variable Rate Coding“, *IEEE Trans. Information Theory*, **IT-24**, 530-536 (1978).
- [18] M. Nelson und J. Gailly, „The Data Compression Book“ M&T (New York, 1995).
- [19] D. Knuth, *The Art of Computer Programming, Vol. 2 Seminumerical Algorithms 2nd Edition*, Addison-Wesley (Reading, 1981).
- [20] J.A. Storer, *Data Compression: Methods and Theory*, Computer Science Press (1988).
- [21] A. J. Stam, „Some Inequalities Satisfied by the Quantities of Information of Fisher and Shannon“, *Inform. and Control*, **2** 101-112 (1959).
- [22] P. Lupton, *Simplicity and Misrepresentation*, unveröffentlichtes Manuskript.
- [23] J. Hartmanis, „The Applications of Some Basic Inequalities for Entropy“, *Inform. and Control*, **2**, 199-213 (1959).
- [24] C. E. Shannon, „Prediction and Entropy of Printed English,“ *Bell Sys. Tech. Jour.*, **30**, 51-64 (1951); *Claude Elwood Shannon: Collected Papers*, IEEE Press (Piscataway, 1993).
- [25] M. Feder, „Gambling Using a Finite State Machine,“ *IEEE Trans. Information Theory*, **IT-37**, 1459-1465 (1991).
- [26] M. Feder, N. Merhav und M. Gutman, „Universal Prediction of Individual Sequences“, *IEEE Trans. Information Theory*, **IT-38**, 1258-1270 (1992).
- [27] A. Kolmogorov, „Logical Basis for Information Theory and Probability Theory“, *IEEE Transactions on Information Theory*, **5** 662-664 (1968).
- [28] R. Solomonoff, „A Formal Theory of Inductive Inference“, *Inform. and Control*, **7**, 1-22 und 224-254 (1964).

- 
- [29] G. Chaitin, *Algorithmic Information Theory*, (Cambridge University Press, 1987).
- [30] M. Li und P. M. B. Vitányi, *Introduction to Kolmogorov Complexity and its Application*, Springer (New York, 1993).
- [31] W. Ebeling und M.A. Jimenez Montano, „On Grammars, Complexity, and Information Measures of Biological Macromolecules“ *Math. Biosc.* **52**, 53-71 (1980).
- [32] G. J. Chaitin, „The Berry Paradox“, *Complexity*, **1**, 26-30 (1995).
- [33] J. v. Heijenoort, *From Frege to Gödel: A Source Book in Mathematical Logic, 1879-1931*, Harvard Univ. Press (Cambridge, 1967).
- [34] E. B. Newman und N. C. Waugh, „The Redundancy of Texts in Three Languages“, *Inform. and Control*, **3** 141-153 (1960).
- [35] G. Siromoney, „Entropy of Tamil Prose“ *Inform. and Control*, **6**, 297-300 (1963).
- [36] T. Pöschel, W. Ebeling und H. Rosé, „Guessing Probability Distributions from Finite Samples“ , *J. Stat. Phys.*, **80** 1443-1452 (1995).
- [37] K. R. Rajagopalan, „A Note on Entropy of Kannada Prose“ , *Inform. and Control*, **8**, 640-644 (1965).
- [38] A. D. Booth, „A ‚Law‘ of Occurrences for Words of Low Frequency“ , *Inform. and Control*, **10**, 386-393 (1967).
- [39] P. Balasubrahmanyam und G. Siromoney, „A Note on Entropy of Telugu Prose“ , *Inform. and Control*, **13**, 281-285 (1968).
- [40] W. Ebeling und G. Nicolis „Entropy of Symbolic Sequences: The Role of Correlations“ *Europhys. Lett.*, **14**, 191-196 (1991).
- [41] W. Ebeling und G. Nicolis, „Word Frequency and Entropy of Symbolic Dynamics: A Dynamical Perspective“ *Chaos, Solitons & Fractals* **2**, 635-650 (1992).
- [42] W. Ebeling und T. Pöschel, „Entropy and Long-Range Correlations in Literary English“ *Europhys. Lett.*, **26** (1994) 241-246.
- [43] M. C. Grignetti, „A Note on the Entropy of Words in Printed English“ , *IEEE Transactions on Information Theory*, **7** (1964).
- [44] D. Jamison und K. Jamison, „A Note on the Entropy of Partially-Known Languages“, *Inform. and Control*, **12**, 164-167 (1968).
- [45] P. Grassberger, „Toward a Quantitative Theory of Self-Generated Complexity“, *Int. J. Theor. Phys.*, **25**, 907-938 (1986).
- [46] K. Rateitschak, „Entropie und Grammatik im Periodenverdopplungsszenario der logistischen Abbildung“ , in diesem Buch.
- [47] D. Beule und I. Große, „Wie zufällig sind Zufallszahlen?“ , in diesem Buch.