Über die Entropie von Sprachen

Von Alexander Guthmann

Der Begriff Information

Die Bedeutung des Wortes Information weicht im Kontext der Informationstheorie von dem alltäglichen Gebrauch ab. Spricht man von Information, so meint man meist Bedeutung oder Sinn. Grob gesagt ist die Information ein Maß für die Freiheit beim Wählen einer Nachricht. Im Zusammenhang mit Sprache interessiert uns wie groß die Informationsdichte und Redundanz unserer Sprache ist. Wir möchten wissen wie viel Information ein Buchstabe liefert.

Entropie

Wir wollen an $H(p_1, \dots, p_n)$ folgende Eigenschaften[7] stellen:

1. $H(p_1, \dots, p_n)$ ist stetig für alle p_i
2. Wenn alle $p_i = \frac{1}{n}$ gleich sind soll $H\left(\frac{1}{n}, \dots \frac{1}{n}\right) < H(\frac{1}{n+1}, \dots \frac{1}{n+1})$ sein.
3. Für $d_i \in \mathbb{Z}^+$, $\sum d_i = n$

$$H\left(\frac{1}{n}, \dots \frac{1}{n}\right) = H\left(\frac{d_1}{n}, \cdots, \frac{d_k}{n}\right) + \sum_{i=1}^{k} \frac{d_i}{n} H\left(\frac{1}{d_i}, \dots, \frac{1}{d_i}\right)$$

Unterteilt man die p_n in unabhängige Klassen, so soll H unverändert bleiben.

Eine ausführliche Definition findet man C.E. Shannons Originalarbeit[3] und in der Literatur[7] zur Informationstheorie.

Diese Eigenschaften definieren eindeutig eine Funktion. Es ist die Entropie H.[7]

$$H = -\sum_i p_i \log p_i$$

Berechnung der Entropie von Sprachen

Analog zu oben definieren wir die Entropie H_n der n-Gramme:

$$H_n = -\sum_i p(b_i^n) \log_2 p(b_i^n)$$

Hierbei ist b_i^n ein Block aus n Buchstaben und $p(b_i^n)$ dessen relative Wahrscheinlichkeit. Der Informationsgehalt eines Buchstaben ergibt sich aus der Differenz F_n der Entropie der (n+1)-Gramme und der n-Gramme. Man nennt die F_n bedingte Entropie.[1] Man vergleiche dazu Anhang A.

$$F_n = H_{n+1} - H_n$$

Aus den Bedingungen die wir an die Entropiefunktion H gestellt haben ergibt sich, dass F_n einen positiven Grenzwert E besitzt.

$$E = \lim_{n \to \infty} F_n \qquad \text{Satz v. Shannon}^1$$

Die Wahl des $\log_2$ hat den Vorteil, dass wir die Zahl E als Anzahl der Bits oder Ja-Nein Entscheidung ansehen können, die nötig sind um einen Buchstaben zu bestimmen.

Um F_n zu berechnen müssen wir die relativen Wahrscheinlichkeiten $p(b_i^n)$ bestimmen. Dazu nimmt man eine genügend große Textquelle und zählt das Auftreten (Frequenz) der n-Gramme innerhalb dieses Texts.

Redundanz und Relative Entropie

Nun ist es nicht nur von Interesse die Information eines Buchstaben zu bestimmen, sondern auch den prozentualen Anteil unserer Sprache der keine Information enthält. Die Entropie wird maximiert wenn alle n-Gramme vorkommen und gleich wahrscheinlich sind. Im Fall des lateinischen Alphabets inklusive Leerzeichen gilt $p_n = \frac{1}{27^n}$ und für die maximale Entropie M_n verifiziert man:

$$M_n = -\log_2 p_n$$

Mit der Entropie H_n der n-Gramme können wir ein Maß G_n für den informationstragenden Anteil angeben.

$$G_n = \frac{H_n}{M_n}$$

Dieser ist äquivalent zur maximal möglichen Kompression im gegebenen Alphabet. Deswegen ist es Möglichkeit die Entropie der Sprache - oder besser eine Obergrenze - durch komprimieren von Text zu bestimmen.

Der Anteil R_n eines Textes ohne Information, die Redundanz, ist dann folglich

$$R_n = 1 - G_n$$

Durchführung

Nachdem nun klar ist was zu tun ist, n-Gramme zählen, stellt sich die Frage nach einer geeigneten Datenquelle. Wikipedia eignet sich dafür hervorragend.[4] Verfügbarkeit in vielen verschieden Sprachen und klare sowie liberale Lizenzierung machen Wikipedia zur idealen Datenquelle. Der Text ist von hoher Qualität und man hat keine Probleme mit fehlerhaften Scans wie es bei Büchern der Fall ist.

Die Software wurde in C++ geschrieben um eine maximale Speichereffizienz und Geschwindigkeit zu erreichen. Das Softwarepaket besteht aus drei verschiedenen Programmen. Das Programm *ngram* zählt die n-Gramme in einer Textdatei sowie Wörter und Wort-n-Gramme. Die n-Gramm-Frequenzen werden als CSV-Datei gespeichert. Ein weiteres Programm *entropy* berechnet aus dieser CSV-Datei die Entropie H_n. Zu guter Letzt, erstellt das Programm *histogramm* ein Histogramm der n-Gramm-Frequenzen.

Ergebnisse

Es wurde die Entropie von neun verschiedenen europäischen Sprachen untersucht. Aus Zeitmangel musste die Untersuchung auf Sprachen mit lateinischem Alphabet beschränkt bleiben. Dies zwingt den Ergebnissen eine gewisse zu erwartende Homogenität auf. In Tabelle 1 sind die verwendeten Textkorpora der Größen nach sortiert.

	EN	DE	FR	ES	IT	SV	NL	PL	PT
Größe [GB]	13	6,1	5,5	3,8	3,2	2,4	1,8	1,8	1,7
Artikel $* 10^6$	5,8	2,3	2	1,5	1,5	3,7	2	1,3	1

Tabelle 1

Die berechneten Entropien F_n sind in Abb. 1 dargestellt.

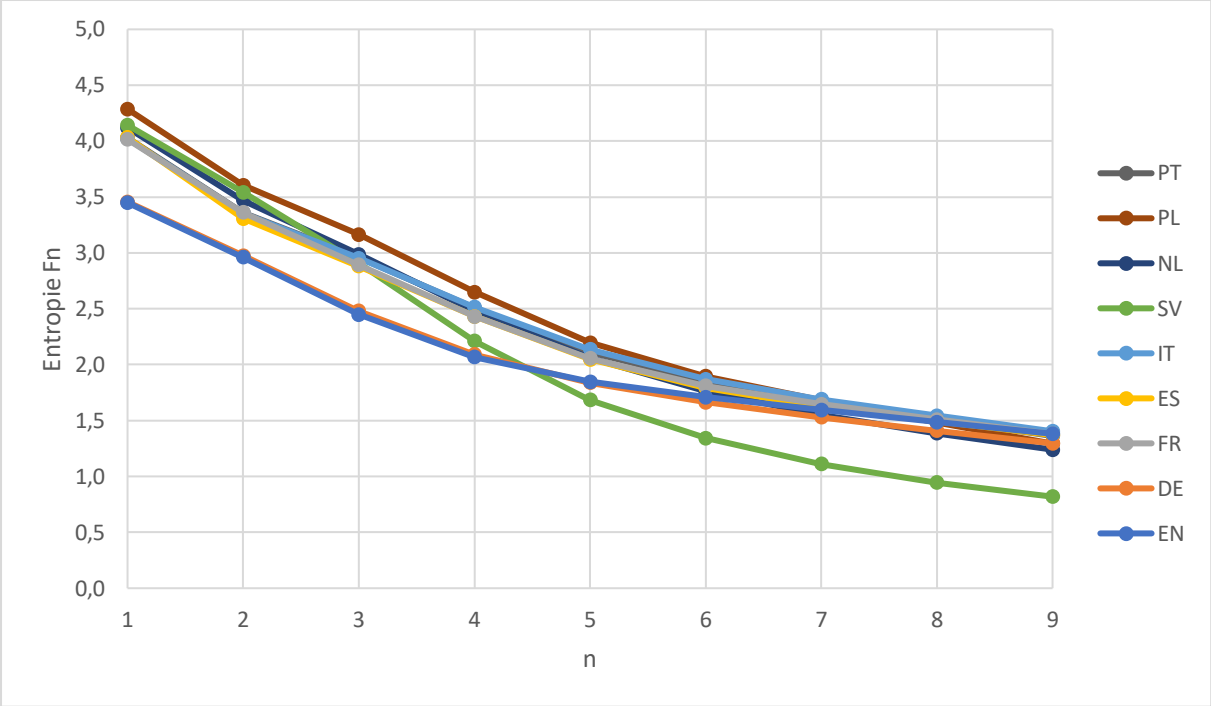

Abbildung 1

Es ist zu erahnen, dass alle untersuchten Sprachen etwa einem Grenzwert von 1 Bit pro Buchstabe zustreben.

Die einzige Ausnahme scheint das Schwedische mit einer deutlich geringeren Entropie zu sein. Ob dies eine tatsächliche Eigenschaft der schwedischen Sprache ist, oder andere Ursachen hat ist nicht klar. Eine mögliche Ursache ist, dass die schwedische Wikipedia nach Anzahl der Artikel die zweit größte ist, jedoch die absolute Textgröße recht gering ist. Es ist vorstellbar, dass durch schematisch gleichen Aufbau von Einträgen das Ergebnis verfälscht wird. Man denke nur an den immer ähnlichen Artikelanfang von Personeneinträgen. Für diese Erklärung spricht auch, dass sich der Effekt erst bei höheren n-Gramm Ordnung bemerkbar machen würde, wie es tatsächlich der Fall ist.

Auch ist es interessant, dass Deutsch und Englisch einen nahezu identischen Verlauf aufweisen. Die inhomogene Häufigkeitsverteilung der Buchstaben (1-Gramme) dieser beiden Sprachen äußert sich durch eine geringe Entropie bei $n = 1$. Die maximale Entropie für das 27 Zeichen Alphabet liegt bei $M_1 = \log_2(27) \approx 4{,}75$. Das Polnische kommt diesem Wert am nächsten.

In Abb. 2 ist die Redundanz für die neun Sprachen aufgetragen. Die Redundanz liegt bei allen Sprachen, bis auf dem Schwedischen, etwas über 50%. Konkret bedeutet das, dass etwa die Hälfte der Buchstaben in einem Text keine Information enthalten.

Wie Shannon in seiner Originalarbeit begründet ist dies der Grund dafür, dass es möglich ist Kreuzworträtsel zu spielen. So wäre es mit einer Sprache die etwa 30% Redundanz besitzt möglich dreidimensionale Kreuzworträtsel zu erstellen.

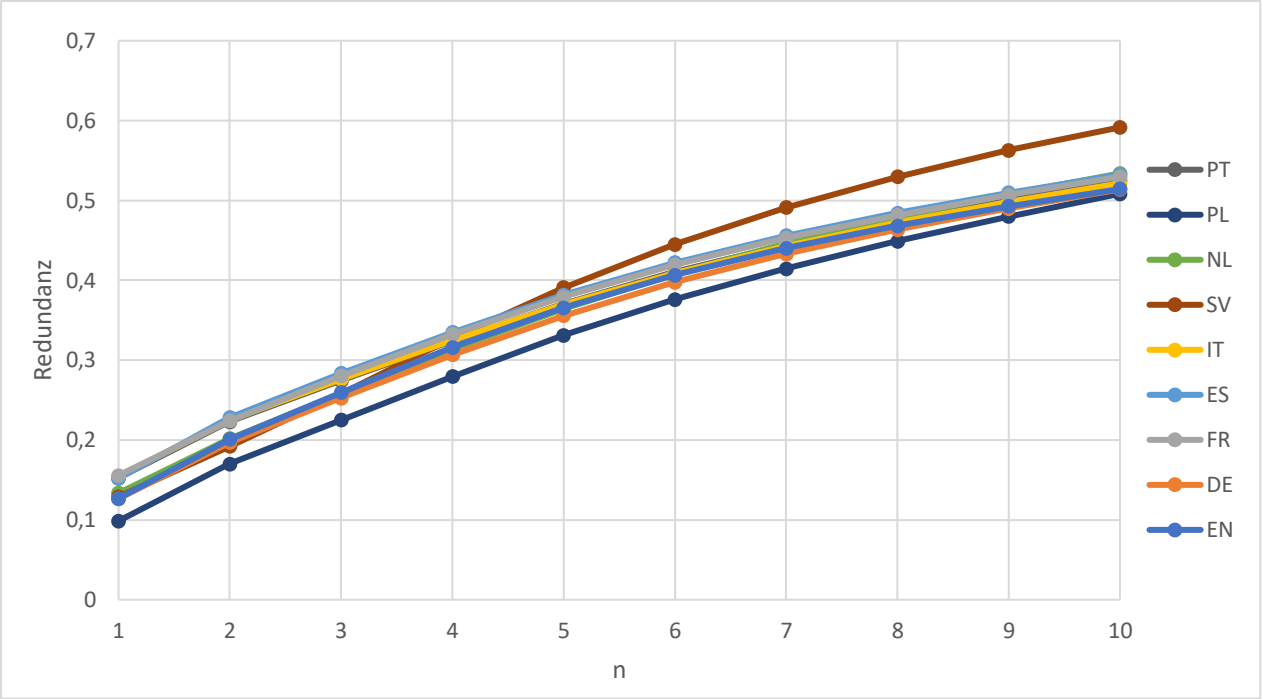

Abbildung 2

Zipfsches Gesetz und 1/f Rauschen

Das Zipfsche Gesetz wurde erstmals von G.K. Zipf[5] in den 1930er entdeckt und besagt folgendes: Ordnet man die Wörter nach ihrer Häufigkeit und trägt darüber die relative Häufigkeit auf, so ist die relative Häufigkeit $f(r)$ eines Wortes umgekehrt proportional zu seinem Rang r.

$$f(r) = \frac{a}{r^{\gamma}}$$

Es gilt etwa $a \approx 0{,}1$ und $\gamma \approx 1$. B. B. Mandelbrot hat diesen Zusammenhang erstmals aus der Informationstheorie hergeleitet.

Das $1/f$ Verhalten zeigt sich eindeutig, wenn man die Anzahl verschiedener n-Gramme gegen die n-Gramm-Frequenz (Anzahl der Fundstellen in einem Text) doppelt logarithmisch aufträgt. In Abb. 3 ist dies für die Tri-Gramme der deutschen Sprache zu sehen. Das $1/f$ Rauschen kann bei erstaunlich vielen Prozessen in der Natur beobachtet werden.[6] Das es in unserer Sprache auftritt ist nicht verwunderlich wenn man bedenkt, dass der Mensch vor etwa 60.000 Jahren anfing zu Sprechen.

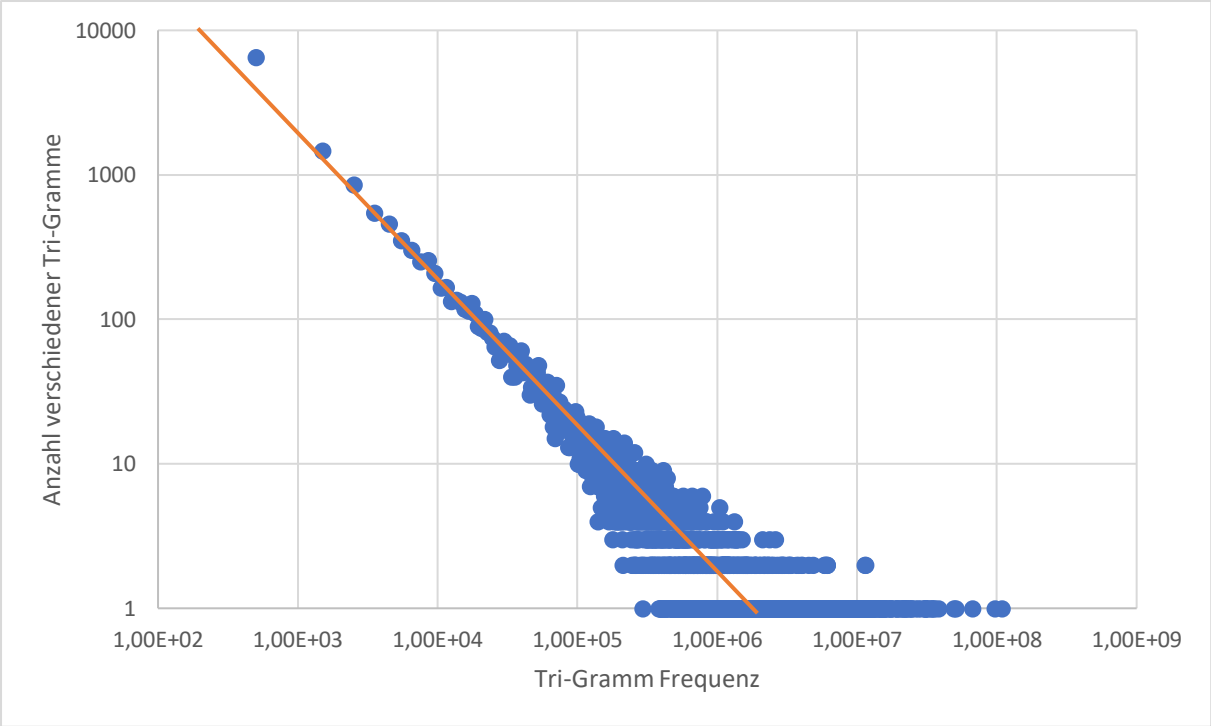

Abbildung 3

Das Diagramm ist wie folgt zu verstehen: Der Punkt ganz links besagt, dass es etwa 9000 verschiedene Tri-Gramme gibt, die weniger als 1000 im Text vorkommen. Um das Histogramm zu erstellen wurde die Tri-Gramm Frequenz in Klassen mit einer Breite von 1000 unterteilt. Die dargestellte Tri-Gramm Frequenz entspricht dem Mittelpunkt einer Klasse.

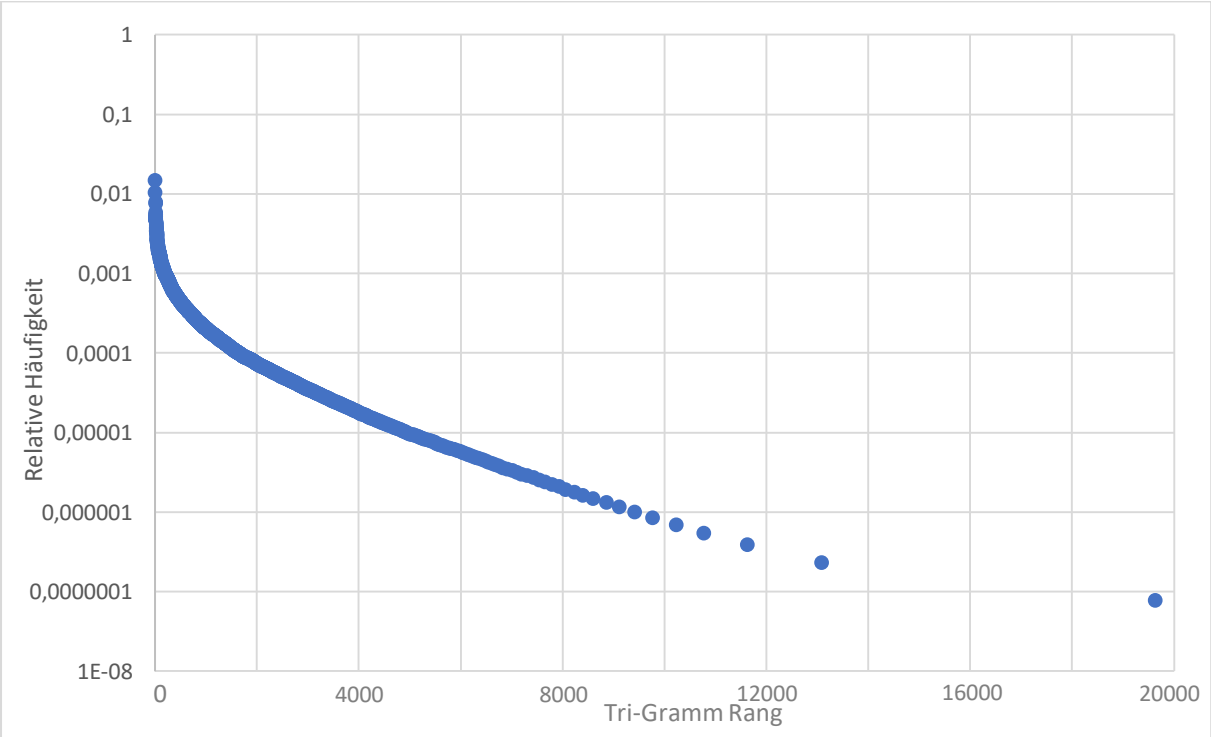

Abbildung 4

Aus diesen Daten kann dann das Zipfsche Gesetz gewonnen werden indem man die Tri-Gramme entsprechend ihrer Häufigkeit einen Rang zuweist und darüber die relative Häufigkeit derselben aufträgt. Wie in Abb. 4 zu sehen ist.

Das Zipfsche Gesetz bezieht sich ursprünglich auf einzelne Wörter. Doch wie man sieht unterliegen auch die Tri-Gramme diesem Verhalten. Die wahre Natur des $1/f$ Verhaltens wird jedoch in der Darstellung der Tri-Gramm-Frequenzen besser wiedergegeben.

Speicherplatz

Um die relativen Wahrscheinlichkeiten der n-Gramme zu bestimmen muss die Anzahl der Fundstellen in einem Text gespeichert werden. Es muss also jedem einzelnen n-Gramm eine Speicheradresse zugewiesen werden dessen Inhalt entsprechend hochgezählt wird. Wie sich zeigt, ist diese Zuweisung nicht trivial. Glücklicherweise gibt es für dieses Problem bereits eine Lösung, die Hashtabelle. Aus der n-Gramm-Zeichenkette wird ein Hash berechnet und dieser einer Speicheradresse zugewiesen. Nachteilig hierbei ist jedoch, dass die Hash-Funktion nicht eindeutig ist und somit das n-Gramm selbst gespeichert werden muss.

Die Anzahl der möglichen n-Gramme beträgt 27^n, ein Buchstabe belegt 8 Bit = 1 Byte Speicher und die n-Gramm-Frequenz wird mit einer Integer-Variablen der Größe l gespeichert. Für den Speicherplatz g_n gilt dann folgender Zusammenhang:

$$g_n \propto 27^n (l + n)$$

Die benötigte Speichergröße wächst also exponentiell, die Rechenzeit hingegen linear. Natürlich werden nicht alle n-Gramme realisiert, allein schon, weil die Quelle eine endliche Größe hat. Trotzdem ist klar, dass der Speicherplatz die limitierende Größe ist. Um dennoch ein möglichst großes n zu erreichen, wurden die Berechnungen auf einem Server mit 60GB RAM ausgeführt.

Aufbereitung der Daten

Wie fast alle neuen Internetseiten, ist Wikipedia im UTF-8 Zeichensatz kodiert. Ein Buchstabe ist unter Umständen mehrere Byte groß. Das Programmpaket unterstützt jedoch aus praktischen Gründen nur 1 Byte Zeichensätze. Deswegen wurden der Text zunächst in den ISO88592 Zeichensatz konvertiert, der die wichtigsten westeuropäischen Sprachen unterstützt. Danach wurde der Text in Kleinbuchstaben umgewandelt und alle Umlaute bzw. Sonderzeichen in ihre zugrundeliegenden Buchstaben umgewandelt wie z.B. Ü → ü → u. Zuletzt wurden alle Zeichen, die nicht dem lateinischen Alphabet inklusive Leerzeichen angehören entfernt. Im Abschnitt Normalisierung wird der Effekt dieses Vorgehens untersucht. Näheres zum Ablauf findet sich in Anhang B.

Einfluss der Normalisierung

Es stellt sich die Frage nach dem Einfluss der Textaufbereitung auf die Ergebnisse. Um das zu untersuchen wurde die deutsche Wikipedia auf drei unterschiedliche Weisen normalisiert. Die Ergebnisse sind in Tabelle 2 dargestellt.

In der Spalte „Normal“ sind die F_n zu sehen, wobei Umlaute wie bei den anderen Sprachen in ihre Basisbuchstaben umgewandelt wurden. $\{\text{ä}, \text{ö}, \text{ü}, \text{ß}\} \rightarrow \{\text{a}, \text{o}, \text{u}, \text{s}\}$
Dann wurde die Entropie mit Umlauten untersucht. Das Alphabet wurde um die vier Umlaute erweitert (Spalte „Mit“).
Eine weitere Möglichkeit mit den Umlauten zu verfahren ist sie einfach zu löschen (Spalte „Ohne“).

n	Normal	Mit	Ohne
1	3,4557	3,4857	3,4847
2	2,9760	3,0246	3,0239
3	2,4801	2,5244	2,5242
4	2,0915	2,1311	2,1313
5	1,8376	1,8679	1,8684
6	1,6641	1,6842	1,6848
7	1,5288	1,5402	1,5407
8	1,4086	1,4136	1,4140
9	1,2958	1,2954	1,2956

Tabelle 2

Wie man sieht, ist bei $n = 9$ der Unterschied zwischen den drei Methoden verschwindet klein. Erwartungsgemäß ist F_1 mit Umlauten am Größten, schließlich ist das Alphabet größer.

Einfluss der Textgröße

Wie sich gezeigt hat, ist neben dem Speicherplatz die Größe der verwendeten Textquelle eine weitere limitierende Größe. Wählt man n zu groß, so sind die Frequenzen (Anzahl Fundstellen) der einzelnen n-Gramme so gering, dass keine statistisch relevanten Aussagen getroffen werden können. Offensichtlich ist hierbei die Größe der Textquelle entscheidend. Denn die Größe eines Textes entspricht ja gerade der totalen Anzahl n-Gramme in selbigem.

So kommt in der englischen Wikipedia bei $n = 10$ jedes 10-Gramm im Mittel 18,2 Mal vor. Die Mittleren Häufigkeiten der n-Gramme in den analysierten Texten sind in Tabelle 7 im Anhang C zu finden.

In Tabelle 6 findet man die Anzahl der verschiedenen gefunden n-Gramme. Interessanterweise sind nur im Englischen alle Tri-Gramme realisiert. Dabei sind es nur $27^3 - 3 = 19680$ Stück, abzüglich 3 da mehrfache Leerzeichen gelöscht wurden.

Ausblick

Im Rahmen dieser Arbeit wurden nur Sprachen mit lateinischem Alphabet untersucht. Es wäre sehr interessant Sprachen mit anderem Alphabet zu untersuchen wie z.B. kyrillische Sprachen. Des Weiteren könnte es sich lohnen die Statistik der Plansprachen Esperanto und Interlingua zu untersuchen.

Der Vorgang der Sprachbildung ist eine Markoff-Kette äußerst großer Ordnung. Die Wahrscheinlichkeit für den Übergang von einem n-Gramm Zustand in einen anderen hängt stark von den vergangenen Zuständen ab. Die Übergangswahrscheinlichkeiten der n-Gramme sind der Frequenztabelle der 2n-Gramme enthalten. Ohne größeren Aufwand könnte man ein Programm schreiben welches die Übergangsmatrix aus den Frequenztabellen berechnet. Anhand der Übergangsmatrizen für große n könnte man den statistischen Vorgang der Sprachbildung auf Ergodizität untersuchen.

Zwar wurden das Programm zum Zählen der n-Gramm Frequenzen auf einem Server mit 60GB Arbeitsspeicher ausgeführt, trotzdem war es damit nur möglich bis $n = 10$ zu gehen. Durch den Einsatz eines Hochleistungsrechners mit mehr Arbeitsspeicher wäre es möglich n-Gramme höherer Ordnung zu untersuchen. Voraussetzung ist, dass sich eine Textquelle findet die ausreichend groß ist um ein aussagekräftiges Ergebnis zu erhalten.

Anhang A: Die bedingte Entropie

$$\begin{aligned} F_n &= H_{n+1} - H_n \\ &= -\sum_i p\left(b_i^{n+1}\right) \log_2 p\left(b_i^{n+1}\right) + \sum_i p(b_i^n) \log_2 p(b_i^n) \\ &= -\sum_{i,j} p(b_i^n, j) \log_2 p(b_i^n, j) + \sum_i p(b_i^n) \log_2 p(b_i^n) \end{aligned}$$

Die zweite Summe wird mit dem Faktor $1 = \sum_j p(j)$ erweitert. Also ist wegen $p(b_i^n)p(j) = p(b_i^n, j)$

$$\begin{aligned} F_n &= -\sum_{i,j} p(b_i^n, j) \log_2 p(b_i^n, j) + \sum_i p(b_i^n) \log_2 p(b_i^n) \sum_j p(j) \\ &= -\sum_{i,j} p(b_i^n, j) \log_2 p(b_i^n, j) + \sum_{i,j} p(b_i^n, j) \log_2 p(b_i^n) \\ &= -\sum_{i,j} p(b_i^n, j) \left[\log_2 p(b_i^n, j) - \log_2 p(b_i^n)\right] \\ &= -\sum_{i,j} p(b_i^n, j) \log_2 \frac{p(b_i^n, j)}{p\left(b_i^n\right)} \\ &= -\sum_{i,j} p(b_i^n, j) \log_2 p_{b_i^n}(j) \end{aligned}$$

Damit ist klar, warum F_n bedingte Entropie genannt wird.

Anhang B: Textaufbereitung und Arbeitsablauf

Die Spiegelung der Wikipedia Datenbank[4] ist zunächst nach dem Entpacken eine riesige XML-Datei welche die gesamte Wikipedia Datenbank enthält. Glücklicherweise gibt es einen offenen Parser[8] der den reinen Text extrahiert.

Die anschließende Konvertierung vom UTF-8 Zeichensatz in ISO88592 erfolgt mit dem unix-Befehl:

iconv -c -f UTF8 -t ISO88591 < input.txt > output.txt

Das Normalisieren des Textes erfolgt mit dem Skript *convert.sh* welches das unix-Programm *tr* benutzt.

Den Vorgang der Normalisierung erkennt man gut an einem Beispiel:

Ausgangstext:

> *Die barometrische Höhenformel beschreibt die vertikale Verteilung der (Gas-)Teilchen in der Atmosphäre der Erde, also die Abhängigkeit des Luftdruckes von der Höhe. Man spricht daher auch von einem vertikalen Druck-Gradienten, der jedoch aufgrund der hohen Wetterdynamik innerhalb der unteren Atmosphäre nur mit Näherungen auf mathematischem Wege beschrieben werden kann.*

Normalisiert:

> *die barometrische hohenformel beschreibt die vertikale verteilung der gas teilchen in der atmosphare der erde also die abhangigkeit des luftdruckes von der hohe man spricht daher auch von einem vertikalen druck gradienten der jedoch aufgrund der hohen wetterdynamik innerhalb der unteren atmosphare nur mit naherungen auf mathematischem wege beschrieben werden kann*

Anhang C: Rohdaten

Die n-Gramm Entropie H_n:

n	EN	DE	FR	IT	ES	NL	PL	SV	PT
1	4,1519	4,1533	4,0148	4,0220	4,0320	4,1201	4,2861	4,1423	4,0294
2	7,6008	7,6380	7,3765	7,3749	7,3399	7,5912	7,8916	7,6842	7,3893
3	10,5623	10,6620	10,2678	10,3265	10,2195	10,5790	11,0572	10,5899	10,3511
4	13,0108	13,1862	12,6995	12,8402	12,6504	13,0592	13,7053	12,8001	12,8567
5	15,0787	15,3175	14,7551	14,9769	14,6973	15,1184	15,9013	14,4846	14,9687
6	16,9266	17,1859	16,5656	16,8462	16,4886	16,8813	17,7977	15,8282	16,8153
7	18,6355	18,8707	18,2110	18,5367	18,1149	18,4337	19,4792	16,9402	18,4797
8	20,2297	20,4114	19,7195	20,0801	19,6091	19,8189	20,9631	17,8857	19,9900
9	21,7152	21,8254	21,1000	21,4841	20,9795	21,0587	22,2568	18,7048	21,3526
10	23,0967	23,1210	22,3588	22,7547	22,2310	22,1669	23,3774	19,4224	22,5735

Tabelle 3

Die bedingten Entropien F_n:

n	EN	DE	FR	ES	IT	SV	NL	PL	PT
1	3,4489	3,4558	4,0148	4,0320	4,0220	4,1423	4,1201	4,2861	4,0294
2	2,9615	2,9761	3,3616	3,3080	3,3530	3,5419	3,4711	3,6055	3,3599
3	2,4485	2,4801	2,8913	2,8796	2,9516	2,9057	2,9878	3,1656	2,9619
4	2,0679	2,0915	2,4317	2,4309	2,5137	2,2102	2,4802	2,6481	2,5056
5	1,8479	1,8376	2,0556	2,0469	2,1367	1,6845	2,0592	2,1960	2,1120
6	1,7089	1,6641	1,8105	1,7913	1,8693	1,3436	1,7629	1,8964	1,8466
7	1,5942	1,5288	1,6454	1,6263	1,6905	1,1120	1,5524	1,6815	1,6644
8	1,4855	1,4086	1,5085	1,4942	1,5434	0,9455	1,3852	1,4839	1,5103
9	1,3815	1,2958	1,3805	1,3704	1,4040	0,8191	1,2398	1,2937	1,3626

Tabelle 4

Die Redundanz R_n:

n	EN	DE	FR	ES	IT	SV	NL	PL	PT
1	0,1268	0,1265	0,1556	0,1541	0,1520	0,1335	0,0986	0,1288	0,1526
2	0,2007	0,1968	0,2243	0,2245	0,2282	0,2018	0,1702	0,1920	0,2230
3	0,2595	0,2526	0,2802	0,2761	0,2836	0,2584	0,2249	0,2576	0,2744
4	0,3159	0,3067	0,3323	0,3249	0,3349	0,3134	0,2794	0,3270	0,3240
5	0,3658	0,3557	0,3794	0,3700	0,3818	0,3641	0,3312	0,3907	0,3704
6	0,4067	0,3976	0,4193	0,4095	0,4220	0,4083	0,3762	0,4452	0,4106
7	0,4401	0,4330	0,4529	0,4431	0,4558	0,4462	0,4148	0,4910	0,4448
8	0,4682	0,4634	0,4816	0,4721	0,4845	0,4790	0,4489	0,5298	0,4745
9	0,4926	0,4900	0,5069	0,4980	0,5098	0,5079	0,4799	0,5629	0,5010
10	0,5143	0,5137	0,5298	0,5214	0,5325	0,5338	0,5084	0,5915	0,5253

Tabelle 5

Anzahl verschiedener n-Gramme in den Textkorpora:

n	EN	DE	FR	ES	IT	SV	NL	PL	PT
1	27	27	27	27	27	27	27	27	27
2	728	728	728	728	728	728	728	728	728
3	19630	19622	19628	19625	19577	19204	19071	19346	19404
4	427379	363452	358903	354261	314557	299463	281916	301622	293692
5	3,9E+06	3,1E+06	2,8E+06	2,7E+06	2,4E+06	2,4E+06	2,2E+06	2,4E+06	2,1E+06
6	2,1E+07	1,6E+07	1,4E+07	1,2E+07	1,1E+07	1,1E+07	1,0E+07	1,2E+07	9,1E+06
7	7,8E+07	5,3E+07	4,5E+07	3,9E+07	3,5E+07	3,4E+07	3,1E+07	3,9E+07	2,7E+07
8	2,0E+08	1,3E+08	1,1E+08	9,0E+07	8,3E+07	7,5E+07	6,8E+07	9,2E+07	6,2E+07
9	4,1E+08	2,5E+08	2,0E+08	1,7E+08	1,6E+08	1,3E+08	1,2E+08	1,7E+08	1,1E+08
10	7,1E+08	4,2E+08	3,4E+08	2,8E+08	2,6E+08	2,0E+08	1,9E+08	2,6E+08	1,8E+08

Tabelle 6

Mittlere Häufigkeit der n-Gramme:

n	EN	DE	FR	ES	IT	SV	NL	PL	PT
1	4,8E+08	2,3E+08	2,0E+08	1,4E+08	1,2E+08	8,9E+07	6,7E+07	6,7E+07	6,3E+07
2	1,8E+07	8,4E+06	7,6E+06	5,2E+06	4,4E+06	3,3E+06	2,5E+06	2,5E+06	2,3E+06
3	6,6E+05	3,1E+05	2,8E+05	1,9E+05	1,6E+05	1,2E+05	9,4E+04	9,3E+04	8,8E+04
4	3,0E+04	1,7E+04	1,5E+04	1,1E+04	1,0E+04	8,0E+03	6,4E+03	6,0E+03	5,8E+03
5	3,3E+03	2,0E+03	1,9E+03	1,4E+03	1,3E+03	994,9	817,9	749,9	802,9
6	607,1	386,7	404,6	312,0	293,3	215,3	179,8	155,4	186,1
7	166,5	114,8	123,3	98,6	91,6	70,6	58,9	46,5	62,1
8	63,5	47,3	51,3	42,1	38,4	32,2	26,5	19,6	27,5
9	31,4	24,5	26,8	22,4	20,1	18,4	14,8	10,7	15,0
10	18,2	14,6	16,3	13,7	12,1	12,1	9,5	6,9	9,3

Tabelle 7

Quellennachweise

1. C.E. Shannon, Prediction and Entropy of Printed English, *Bell System Technical Journal* ***30*** *379-423 (1951)*
2. L.B. Levitin, Z. Reingold, Entropy of Natural Languages: Theory and Experiment, *Chaos Solitons & Fractals* ***4****, 709-743 (1994)*
3. C.E. Shannon, A Mathematical Theory of Communication, *Bell System Technical Journal* ***27*** *379-423, 623-656 (1948)*
4. https://dumps.wikimedia.org/
5. G. K. Zipf, The Psycho-biology of Language: An Introduction to Dynamic Philology. *Houghton Mifflin Company (1935)*
6. W. H. Press, Flicker Noises in Astronomy and Elsewhere, *Comments Astrophys.* ***7*** *103-119 (1978)*
7. S. Roman, Coding and Information Theory, *Springer (1992)*
8. https://dizzylogic.com/wiki-parser/